공학박사 학위논문

Improvement of Object Detection in Autonomous Driving using Communication Systems

통신 시스템을 이용한 자율주행 물체감지 성능 향상 기법

2023년 8월

서울대학교 대학원

전기정보공학부

황 선 욱

공학박사 학위논문

Improvement of Object Detection in Autonomous Driving using Communication Systems

통신 시스템을 이용한 자율주행 물체감지 성능 향상 기법

2023년 8월

서울대학교 대학원

전기정보공학부

황 선 욱

Improvement of Object Detection in Autonomous Driving using Communication Systems

지도 교수 박 세 웅

이 논문을 공학박사 학위논문으로 제출함

2023년 8월

서울대학교 대학원

전기정보공학부

황 선 욱

황선욱의 공학박사 학위 논문을 인준함

2023년 6월

위 원 장: 최 완 (인)

부위원장: 박 세 웅 (인)

위 원: 이 경 한 (인)

위 원: 김 형 신 (인)

위 원: 주 창 희 (인)

Abstract

Object detection for autonomous driving has mainly relied on two kinds of methods. One is a cooperative autonomous driving method in which vehicles share their information through V2X communication and cooperate to understand road conditions. The other is a stand-alone autonomous driving method that detects the type of object and the distance between the vehicle and the object by processing the information obtained from vision sensors with a deep learning model for 3D object detection.

In this regard, the above two methods have the following advantages and disadvantages, respectively. The cooperative autonomous driving method using V2X communication has the advantage of detecting vehicles in areas invisible to vision sensors. However, there is a limiting condition that all vehicles must cooperate in information sharing through a communication infrastructure, and reliability problems arise depending on the status information sent by each vehicle. On the other hand, in the stand-alone object detection method via vision sensors, the detection reliability is high, but the area obscured by obstacles cannot be detected. Therefore, research is needed to enhance the advantages of the above two object detection methods to ensure the safety of users driving autonomous vehicles.

In this dissertation, we propose methods to improve the object detection performance for cooperative autonomous driving and stand-alone autonomous driving, respectively: (i) improvement message reception rate (MRR) using Cellular Vehicle-to-Vehicle (C-V2V) on-demand relaying system, and (ii) semi-supervised 3D object detection without sharing raw-level unlabeled scene.

First, we propose a novel C-V2V on-demand relaying system that effectively contributes to finding hidden V-UEs without any subsidiary feedback process and relaying Cooperative Awareness Messages (CAMs) of hidden V-UEs. To achieve this goal, we introduce a novel CAM configuration to contain additional information of nearby

V-UEs without overhead by utilizing previously unused bytes in the conventional CAM. Then, we verify that this novel relay system helps to improve MRR in C-V2V communications.

Second, we propose UpCycling, a novel semi-supervised learning framework for 3D object detection models that utilizes only de-identified intermediate features. Moreover, UpCycling is a unique framework that addresses labeling costs, privacy leakage, and the computational burden on the Autonomous Vehicle (AV) simultaneously. In addition, while preserving privacy, UpCycling performs better or comparably to the state-of-the-art (SOTA) methods that utilize raw-level unlabeled data in both domain adaptation and partial-label scenarios. With the robust performance, UpCycling demonstrates the value of unlabeled feature-based learning in the context of 3D object detection, in terms of both privacy and accuracy.

In summary, from Chapter 2 to Chapter 3, the two pieces mentioned above of the research work, improvement MRR using C-V2V on-demand relaying system and semi-supervised 3D object detection without sharing raw-level unlabeled scenes, respectively. Through this research, we take a step forward to make commercialization of autonomous vehicles by further ensuring user safety.

keywords: 3D object detection, Autonomous driving, Semi-supervised learning, Vehicular communication

student number: 2016-20989

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Main Contributions

1.1.1 On-Demand Relaying in Vehicular Communications

Cellular Vehicle-to-Vehicle (C-V2V) communications take autonomous driving technology to the next level by allowing a Vehicular User Equipment (V-UE) to receive Cooperative Awareness Messages (CAMs) from other V-UEs, and enable the V-UE to see beyond what is detectable by vision-based sensors, thereby preventing accidents and ensuring user safety. However, there remains a fundamental limitation in the conventional CAM broadcasting since a transmitter (TX) V-UE cannot confirm whether its CAM is successfully received at other V-UEs. Without a feedback process, a significant uncertainty arises in CAM reception, posing a critical threat to user safety. To address this threat, we propose Beyond-Vision, an effective C-V2V on-demand relay system that allows CAMs that are not well received at nearby V-UEs to be better received. Through simulation that reflects realistic vehicle mobility and road environments in urban scenarios, we verify the superiority of Beyond-Vision over the conventional C-V2V, which improves performance by up to 215% in terms of message reception ratio (MRR) within a communication range under Non-Line-Of-Sight (NLOS) channels.

In summary, we claim the following contributions in this work.

- We propose a novel C-V2V relay system that improves MRR with no overhead by utilizing previously unused bytes in the conventional CAM.
- We evaluate Beyond-Vision performance via simulation which reflects realistic vehicle mobility and road situations based on Simulation of Urban MObility (SUMO) [6].
- We verify the superiority of Beyond-Vision with the latest C-V2V protocol defined in 3GPP and other relay systems.

1.1.2 Semi-Supervised 3D Object Detection with De-identified Unlabeled Scenes

Semi-supervised Learning (SSL) has received increasing attention in autonomous driving to reduce the enormous burden of 3D annotation. In this paper, we propose UpCycling, a novel SSL framework for 3D object detection with zero additional raw-level point cloud: learning from unlabeled de-identified intermediate features (*i.e.*,, "smashed" data) to preserve privacy. Since these intermediate features are naturally produced by the inference pipeline, no additional computation is required on autonomous vehicles. However, generating effective consistency loss for unlabeled feature-level scene turns out to be a critical challenge. The latest SSL frameworks for 3D object detection that enforce consistency regularization between different augmentations of an unlabeled raw-point scene become detrimental when applied to intermediate features. To solve the problem, we introduce a novel combination of *hybrid pseudo labels* and *feature-level Ground Truth sampling* (*F-GT*), which safely augments unlabeled multi-type 3D scene features and provides high-quality supervision. We implement UpCycling on two representative 3D object detection models: SECOND-IoU and PV-RCNN. Experiments on widely-used datasets (Waymo, KITTI, and Lyft) verify that UpCycling outperforms other augmentation methods applied at the feature level. In addition, while preserving privacy, UpCycling performs better or comparably to the

state-of-the-art methods that utilize raw-level unlabeled data in both domain adaptation and partial-label scenarios.

In summary, we claim the following contributions in this work.

- UpCycling is the first framework that tackles labeling cost, privacy leakage, and AV-side computation cost altogether to train a 3D object detection model, which deeply investigates how to learn from unlabeled intermediate features.
- UpCycling provides a fresh eye on GT sampling in the context of SSL since it safely improves data diversity of unlabeled feature-level 3D scenes and significantly improves pseudo-label quality by providing zero-noise labels.
- UpCycling not only protects privacy but also performs better or comparably to the state-of-the-art methods in both domain adaptation and partial-label scenarios, on representative models and datasets for 3D object detection.

1.2 Organization of the Dissertation

The rest of the dissertation is organized as follows.

Chapter 2 presents that a fundamental problem of conventional CAM broadcasting due to the absence of a feedback process and propose Beyond-Vision, an effective C-V2V on-demand relay system. The design of Beyond-Vision is presented, and the performance evaluation is explained.

Chapter 3 demonstrates that feature-based semi-supervised learning, which combines *hybrid pseudo labels* and *F-GT*, significantly enhances the performance of 3D object detection models while also preserving data privacy. The design of UpCycling is presented, and the performance enhancement is verified via representative datasets and models under various scenarios. Also, we show that UpCycling preserves privacy with intuitive image-level analysis.

Finally, Chapter 4 concludes the dissertation with a summary of contributions and a discussion of future research directions.

Chapter 2

Beyond Vision: Hidden Car Detector with On-demand Relaying in Vehicular Communications

2.1 Introduction

Autonomous driving technology has evolved over time from lab-based "future technology" to "real-world technology" visible on the roads. However, commercialization of the technology requires autonomous vehicles to understand their surroundings to prevent accidents and ensure user safety. Peripheral object recognition is well known as one of the essential functions required for safety in autonomous driving. Until now, autonomous vehicles have mainly relied on sensors such as Light Detection And Ranging (LiDAR), radar, and cameras to detect objects on the roads [7]. However, vehicles face a serious challenge if they solely rely on these vision-based sensors because peripheral sensing is not possible in a Non-Line-Of-Sight (NLOS) environment and external factors such as weather may degrade sensing accuracy.

In an effort to overcome these limitations, studies have been conducted on vehicular communications that can be effective even in NLOS situations and are less vulnerable to external factors. In addition, Cellular Vehicle-to-Vehicle (C-V2V) communications have been standardized based on Long Term Evolution (LTE) [8, 9] since Release 14 of

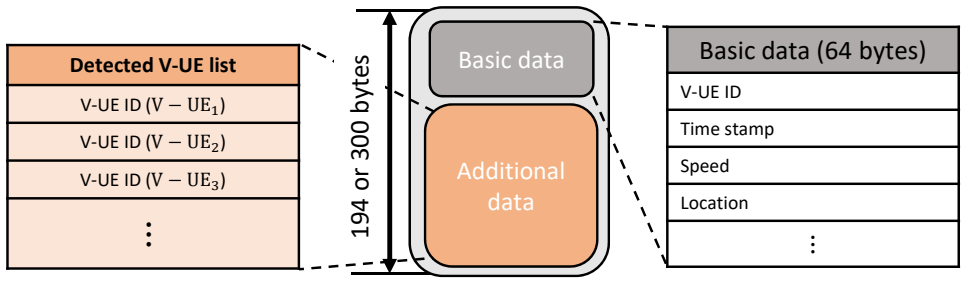


Figure 2.1: Proposed CAM configuration.

3GPP organization. Also, interest in C-V2V has grown recently as it is one of the core services in 5G concerning the safety of autonomous vehicles.

In C-V2V communications, a Vehicular User Equipment (V-UE) periodically broadcasts Cooperative Awareness Messages (CAMs) including its status information for nearby V-UEs.[1] Upon receiving CAMs, V-UEs can detect the existence of other V-UEs transmitting CAMs. The reception of CAMs helps a V-UE to detect other V-UEs beyond the detectable range of vision-based sensors or those invisible due to NLOS positions. In addition, the V-UE can use received CAM information for various driving assistance applications such as collision avoidance, accident warning, and intelligent navigation [10].

However, the conventional CAM broadcasting has a fundamental problem because it has no feedback process to confirm whether a CAM is received or not. In other words, there is no way for a transmitter (TX) V-UE to know whether receiver (RX) V-UEs have received a CAM since the CAM does not contain any feedback information and no feedback message is defined in C-V2V. In particular, in a NLOS situation where vision-based sensors are unable to detect an object, the uncertainty of CAM reception becomes a fatal threat to users.

In this paper, we propose an effective C-V2V on-demand relay system, termed

[1]The CAM contains the V-UE's status information including CAM generation time, V-UE's location obtained from GPS, V-UE's speed, and V-UE's ID, etc.

Beyond-Vision, that enables V-UEs to identify which nearby V-UEs fail to receive which CAMs. To achieve this goal, we focus on the specific information that a CAM should contain. According to the ETSI standard [11], a CAM contains the TX V-UE's status information which occupies approximately 64 bytes of data. However, given the fact that the size of data used for actual CAM transmission is 194 or 300 bytes [10], the size of basic data, including the TX V-UE's status information, is even smaller. As shown in Fig. 2.1, the novel CAM configuration we propose contains additional information of nearby V-UEs detected during the CAM generation period.

Then the V-UE exploits received information of detected V-UE lists to identify which V-UEs have received which CAMs and which V-UEs are hidden to which V-UEs. This novel relay system effectively contributes to finding hidden V-UEs without any subsidiary feedback process, and relaying CAMs of hidden V-UEs, which helps to improve Message Reception Ratio (MRR) in C-V2V communications.

The merits of Beyond-Vision and the contributions of this paper are as follows:

- We propose a novel C-V2V relay system that improves MRR with no overhead by utilizing previously unused bytes in the conventional CAM.
- We evaluate Beyond-Vision performance via simulation which reflects realistic vehicle mobility and road situations based on Simulation of Urban MObility (SUMO) [6].
- We verify the superiority of Beyond-Vision with the latest C-V2V protocol defined in 3GPP and other relay systems.

The rest of this paper is organized as follows. We first present the related work and motivation of the work in Section 2.2. Section 2.3 introduces the basic operation of the conventional C-V2V protocol. Then, we present our proposed relaying scheme, Beyond-Vision, in Section 2.4, and evaluate Beyond-Vision through system-level simulation under various scenarios in Section 2.5.

2.2 Related Work

In this section, we summarize previously studied relaying schemes in V2V communications and present the motivation of our proposed scheme.

2.2.1 Relaying schemes for V2V

Previous relaying studies on V2V communications have been performed primarily under the IEEE 802.11p-based system called Dedicated Short Range Communications (DSRC) [12, 13].

The farthest-first dissemination is the most commonly used strategy to disseminate safety data in V2V communications. This strategy allows the vehicle farthest from the sender to be selected as a relay node for disseminating safety data. For example, Street Broadcast Reduction (SBR) scheme, proposed by Martinez *et al.* [14], utilizes the farthest-first dissemination scheme to reduce the warning message notification time in urban setting scenarios with multiple intersections and obstacles. Urban Multihop Broadcast (UMB) scheme, proposed by Korkmaz *et al.* [15], maximizes its one-hop dissemination performance by selecting a vehicle in the road segment farthest from the sender. Li *et al.* [16] came up with OppCast, a safety data dissemination scheme with enhanced scalability. OppCast operates in two phases. First, farthest-first dissemination takes place to disseminate data as far as possible. Second, make-up dissemination completes the process while ensuring high reliability.

Another method is probability-based broadcasting. In this method, stochastic relaying limits the number of relaying events [17, 18], thereby preventing redundant re-transmissions in V2V communications. Specifically, vehicles are prioritized by their assigned relaying probabilities. Slotted p-Persistence Broadcasting proposed in [17], assigns a relay probability to each relaying according to its distance from the original TX; The farther the vehicle is from the original TX, the larger relaying probability it is assigned. Considering the density level of nearby vehicles, AutoCast proposed in [18]

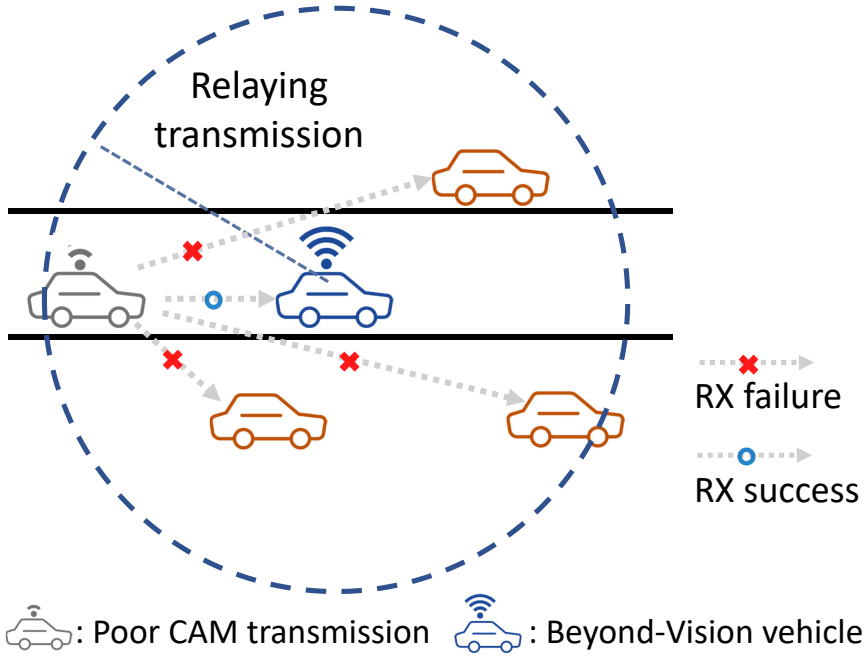


Figure 2.2: Motivation: Finding and relaying CAMs that are not well received at nearby V-UEs.

determines the relaying probability.

There are more studies that propose different relaying schemes. For example, Packet-value-based dissemination scheme (PVCast) [19] presents a novel way to determine relay priority considering both spatial and temporal preferences of each received CAM. In [20], the authors propose a cooperative transmission scheme employing a signal superposition technique. Under this scheme V-UEs superpose other V-UEs' signals that they have received onto their own transmission signals. In [21], the authors propose Reliable Broadcasting of Life Safety Messages (RBLSM) where vehicles nearer to the sender suffer shorter wait time and packets delivered to nearby vehicles experience smaller latency. In [22], the authors compare DSRC and C-V2V communication performance in several aspects. Furthermore, in [23], the authors propose a relay system focusing on hybrid V-UEs, i.e., V-UEs equipped with both DSRC and C-V2V modules.

In [24], the authors propose a relaying scheme with Road Side Units (RSUs) in vehicular communications. Unlike the studies on the above relaying protocols, J. Heo *et al.* explore the utility and trade-off of using buses as mobile RSUs through mathematical analysis, simulation, and real-world experiments [25]. Also, B. Kang *et al.* study the traffic steering scheme to extend the operation of D2D communications to both licensed and unlicensed bands as well as propose a transmission power adaptation algorithm for C-V2X Mode 4 [26, 27].

2.2.2 Motivation of Proposed Beyond-Vision

The previous studies on the relaying scheme described above are as follows. To determine CAM selection priority for relaying, these relaying methods take into account: 1) The distance between TX V-UEs and RX V-UEs, 2) the number of V-UEs that can receive CAMs, and 3) temporal and spatial preferences of each CAM. However, these methods are limited in improving MRR performance because they do not take a sophisticated approach to examine how successfully nearby vehicles receive CAMs when selecting a CAM for relaying.

Our proposed Beyond-Vision ensures effective CAM relaying by addressing the limitation. Specifically, Beyond-Vision finds 'hidden V-UEs' that are not detected because CAMs transmitted by those V-UEs cannot be received within the communication range. As shown in Fig. 2.2, Beyond-Vision enables V-UEs to selectively choose and relay hidden V-UEs' CAMs. Beyond-Vision enables this process simply by adding some information to each CAM, without additional transmission for confirming the reception of CAMs.

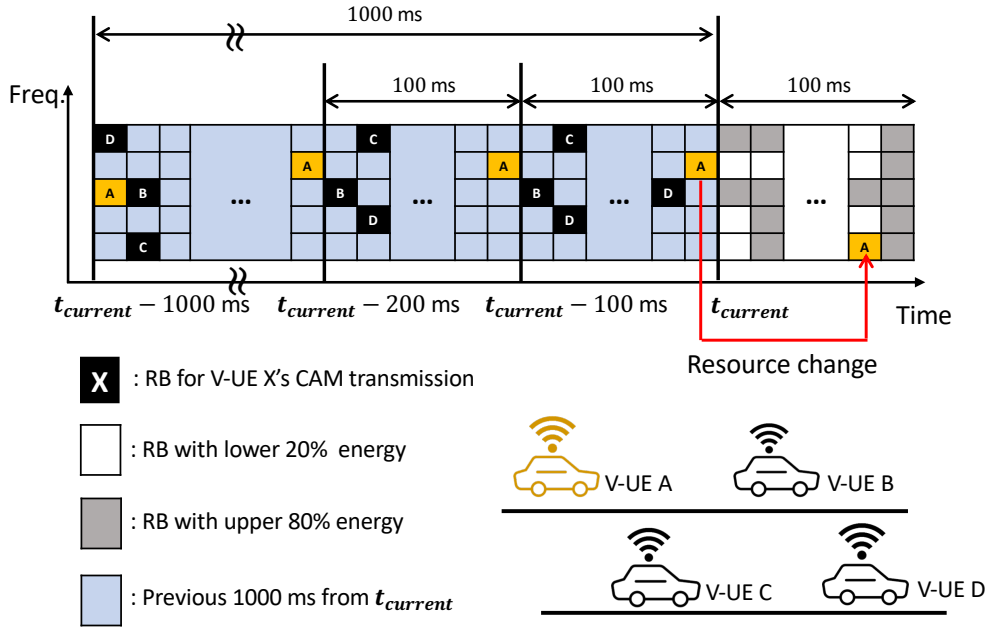


Figure 2.3: C-V2V resource allocation.

2.3 Preliminaries

2.3.1 C-V2V

In this section, we describe C-V2V communications defined in the 3GPP standard Release 14 [8, 9], for which our proposed scheme applies. In C-V2V communications, each V-UE exchanges accurate information such as its ID, location, velocity, and acceleration [11, 28], which contributes to improving traffic safety.

C-V2V was originated from LTE sidelink, called LTE Device-to-Device (LTE-D2D) communications which 3GPP first introduced in Release 12 for public safety. As LTE-D2D was designed to lower battery consumption rather than latency, it is not suitable for C-V2V which requires low latency and high reliability [29, 30]. A significant difference between C-V2V and LTE-D2D is in how to allocate dedicated resources. While LTE-D2D systems rely on specific LTE uplink resources, C-V2V

systems utilize separate resources.

C-V2V communications using a single-carrier frequency division multiple access support one or two channels of 10 MHz in the 5.9 GHz spectrum which many countries already dedicate to vehicular communications [31]. The minimum resource unit that C-V2V utilizes in the 5.9 GHz spectrum is Resource Block (RB).[2] It has a frequency width of 180 kHz (12 subcarriers of 15 kHz) and consists of one subframe (= 1 ms). In the 10 MHz channel, there are 50 RBs available on the frequency axis for C-V2V communications. Also, C-V2V defines subchannel as a group of multiple RBs. Multiple V-UEs can transmit simultaneously by using subchannels in the same subframe.

C-V2V selects a subchannel, which is a resource for transmission in two ways: Sidelink Modes 3 and 4 [9]. Under sidelink Mode 3, Evolved Node B (eNodeB) allocates resources for V-UEs in a centralized manner. Under sidelink Mode 4, in contrast, V-UEs select resources independently. This means that a V-UE under Mode 4 allocates resources regardless of the cellular coverage of the eNodeB. In this paper, we assume that the Beyond-Vision operating environment is controlled in a distributed manner, i.e., sidelink Mode 4.

When selecting a resource for transmission in Mode 4, a V-UE uses the sensing-based Semi-Persistent Scheduling (SPS) scheme, which is defined in 3GPP Release 14. As shown in Fig. 2.3, the V-UE in Mode 4 analyzes energy levels detected during the previous 1000 ms. Based on average sensed Received Signal Strength Indicator (RSSI) analysis, the V-UE extracts a pool of candidate resources from the current time to 100 ms later and selects new resources. In doing so, the V-UE randomly chooses one of the subchannels as a resource with the lowest 20% energy level to avoid possible collisions with adjacent V-UEs that select the same subchannel [9]. With a period of 100 ms, the V-UE repeatedly occupies the resource as many times as a randomly selected counter between 5 and 15. When the counter expires, the V-UE selects a new

[2]Since C-V2V is defined based on LTE, users in LTE system utilize RBs for the minimum resource unit as well.

resource and counter with the same procedure.

2.3.2 Challenge of Relaying Protocols in C-V2V

The operation of relaying protocols in C-V2V should consider the following characteristics of CAMs. First, conventional CAM broadcasting has no feedback process to confirm whether a CAM is received. For this reason, relay transmission may cause unnecessary transmissions by repeatedly relaying already received CAMs. To reduce unnecessary transmissions, the relaying V-UE should find a proper CAM that needs to be relayed under this constraint. Second, a V-UE generates a CAM periodically, and updates its CAM information every 100 ms of the typical option in C-V2V [9]. If the CAM information becomes invalid 100 ms after its generation, the CAM is no longer eligible for relaying. Thus, the V-UE should seek to relay valid CAMs before new ones are created. Third, there is a communication range for CAM transmission, defined differently according to the average speed of the V-UE in the road environment. Specifically, the communication range is defined as 150 m for urban environments [32]. For effective transmission, relaying protocols should be designed to ensure a high reception rate of CAMs within the communication range.

2.4 Beyond-Vision: Proposed C-V2V Relay System

2.4.1 Overview

We propose Beyond-Vision to overcome the defects of the CAM relaying schemes previously studied. As described, the primary goal of Beyond-Vision, which uses a newly proposed CAM configuration, is to select CAMs that are not successfully transmitted to V-UEs within the communication range and to relay them efficiently.[3] Beyond-Vision achieves this goal with the following two features:

[3]We interchangeably use the terms 'the communication range of a CAM' and 'the communication range of a V-UE' to represent the communication range of a V-UE at the moment of the CAM generation.

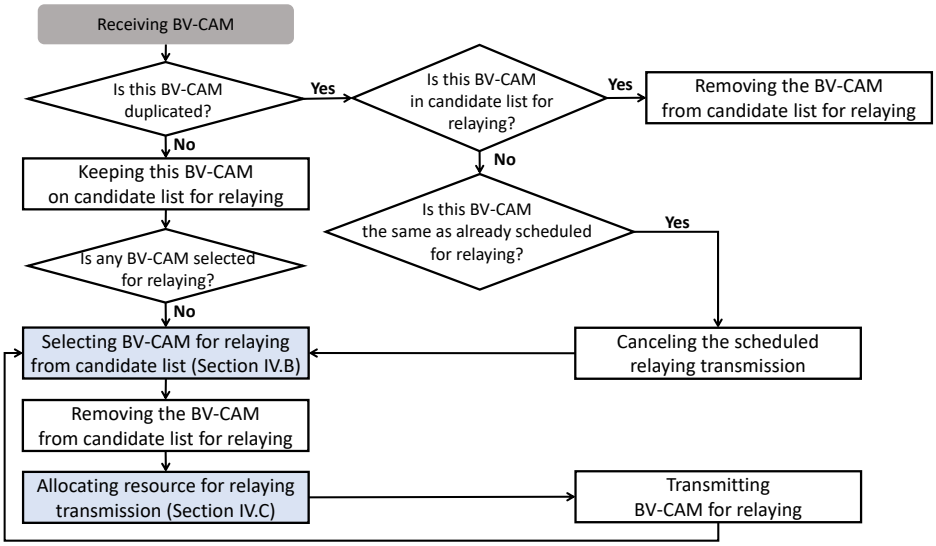


Figure 2.4: Overall Beyond-Vision operation.

- CAM selection algorithm that utilizes novel CAM configuration for relaying
- Standard-compliant relaying that minimizes redundant re-transmissions

Before explaining the details, we present the overall operation of Beyond-Vision described in Fig. 2.4. We define a novel CAM used for the Beyond-Vision as BV-CAM in this paper. A V-UE periodically broadcasts a BV-CAM. Upon receiving a BV-CAM, a V-UE checks whether it is an original BV-CAM or a relayed duplicate BV-CAM.

If the BV-CAM is original, the V-UE forwards the BV-CAM information to the V-UEs in the candidate list for relaying and keeps this information until the BV-CAM becomes invalid. In other words, the candidate list only has information of the received original BV-CAM within 100 ms of its creation. If there is no BV-CAM selected for relaying, then the V-UE selects one from the candidate list for relaying according to the selection algorithm, which will be specified in Section 2.4.2. After the selection, the V-UE removes the BV-CAM from the candidate list, duplicates the BV-CAM, and marks the duplicate on the BV-CAM using one flag bit. Finally, the V-UE allocates resources for relaying transmission. Once a duplicate BV-CAM is transmitted, BV-CAM

selection algorithm is invoked to select a new BV-CAM for next relaying.

If the BV-CAM is a duplicate, on the other hand, the V-UE does not need to relay the BV-CAM. Such duplicate BV-CAMs are removed from the candidate list for relaying and excluded in the selection for BV-CAM relaying. When a BV-CAM equal to the received duplicate has been already scheduled for relaying, the V-UE cancels the scheduled BV-CAM relaying to prevent redundant re-transmissions and selects a new BV-CAM for relaying.

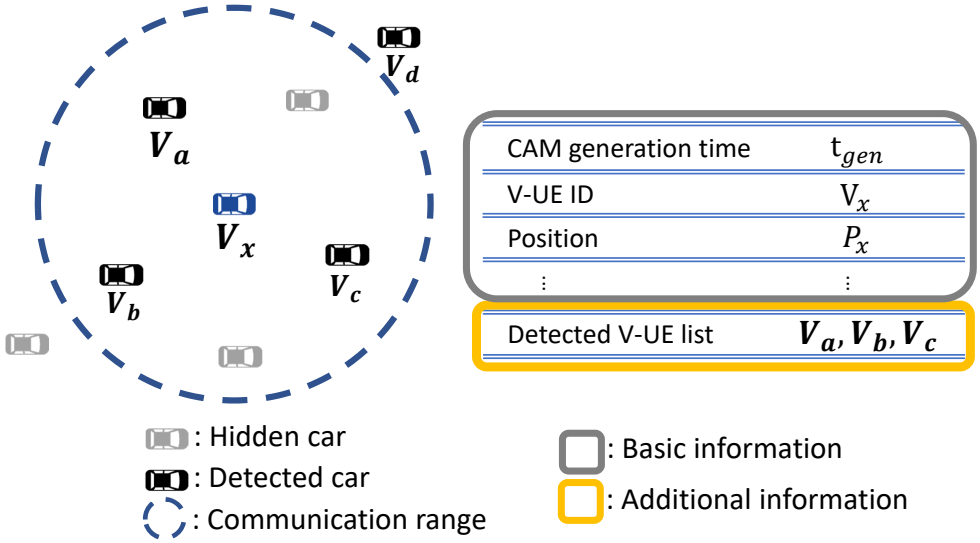
2.4.2 BV-CAM Selection for Relaying in Beyond-Vision

As we mentioned, the conventional CAM carries only the information of the TX V-UE itself and CAM generation time. In this paper, we define a novel CAM configuration for Beyond-Vision. A conventional CAM carries 194 or 300 bytes of data that contains vehicle information, consisting of 64 bytes of basic information. In Beyond-Vision, a V-UE utilizes the vacant space in the conventional CAM to contain Detected V-UE List (DVL), a newly defined list of detected V-UE IDs within the TX V-UE's communication range.

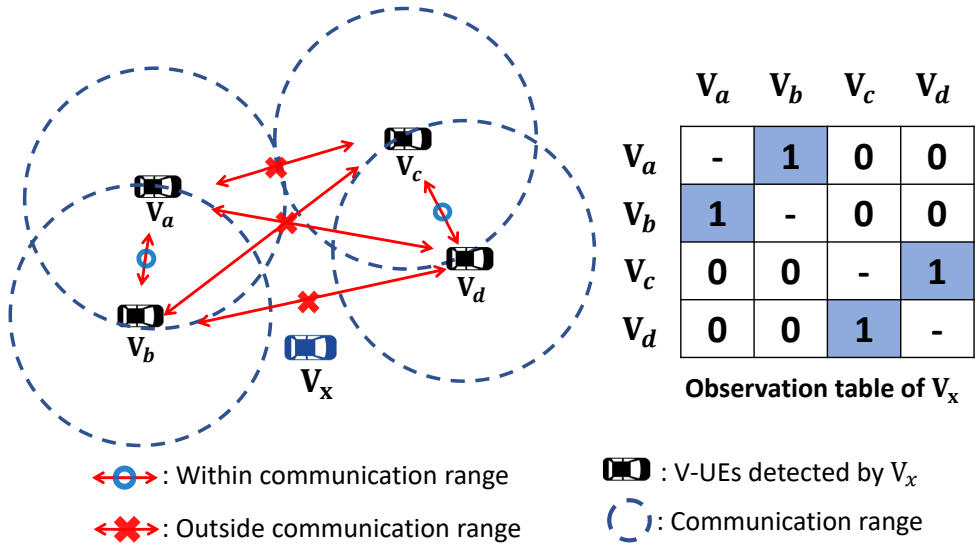
As described in Fig. 2.5a, BV-CAM of V_x contains DVL as well as its basic information. In the DVL, V_x includes the detected V-UE IDs: V_a, V_b, V_c, except for V_d which is detected but exists out of the communication range of V_x.[4] In the process of DVL creation, a V-UE uses only valid BV-CAMs since they present the current state of their TX V-UEs. By doing so, the V-UE not only sends its own status information via BV-CAM, but also notifies the successful reception of valid BV-CAMs transmitted by V-UEs within its communication range.

Each V-UE uses received valid BV-CAMs and their DVLs as the basis of its BV-CAM selection for relaying. The selection process consists of the following compo-

[4] According to the ETSI standard [11], the data size of a V-UE ID is 4 bytes. A 300-byte CAM contains approximately 60 V-UE IDs. If a V-UE utilizes data compression techniques such as hash, its CAM can contain more V-UE IDs.



(a)



	V_a	V_b	V_c	V_d
V_a	-	1	0	0
V_b	1	-	0	0
V_c	0	0	-	1
V_d	0	0	1	-

Observation table of V_x

(b)

Figure 2.5: Proposed configuration: (a) BV-CAM configuration and (b) observation table.

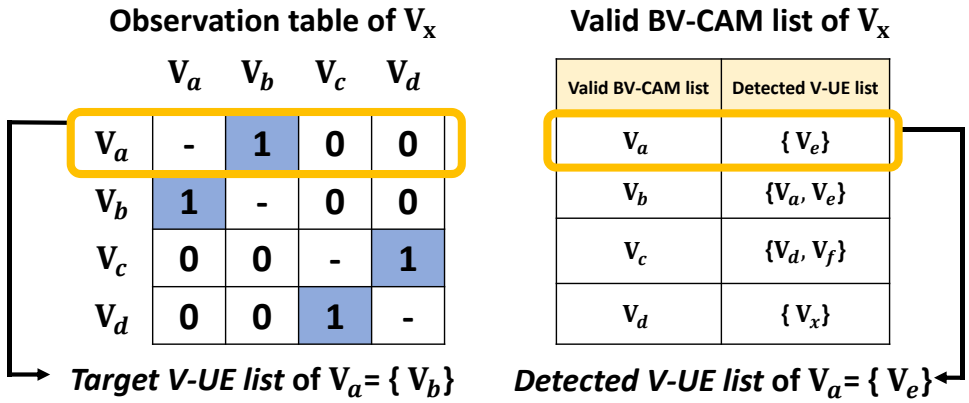


Figure 2.6: Beyond-Vision: BV-CAM selection for relaying.

nents.

Development of Observation Table

A V-UE creates its own Observation Table (OT) with the received valid BV-CAMs. The OT shows the relationship between V-UEs that are detected through valid BV-CAMs. As shown in Fig. 2.5b, assume that there are four valid BV-CAMs received at V_x, and each V-UE's ID is V_a, V_b, V_c, and V_d, respectively. Since each valid BV-CAM contains location information about its TX V-UE, V_x calculates the distance between each V-UE. If the distance between two V-UEs, say $d(V_a, V_b)$, is shorter than the communication range (D_{range}), the relationship between the two is denoted as '1' on the OT. If the distance between the two is longer than D_{range}, on the other hand, their relation is denoted as '0'.

In short, with the information of detected V-UEs, we create the OT by using

$$OT(V_a, V_b) = \begin{cases} 1, & d(V_a, V_b) \leq D_{\text{range}}, \\ 0, & d(V_a, V_b) > D_{\text{range}}. \end{cases} \tag{2.1}$$

Table 2.1: Acronyms and terms

V_x	ID of V-UE
DVL_x	Detected V-UE List of V_x
TVL_x	Target V-UE List of V_x
HVL_x	Hidden V-UE List of V_x
FC_x	Failure Counter of V_x
SC_x	Success Counter of V_x

Calculation of Estimated Message Reception Rate

Estimated Message Reception Rate (eMRR) is a metric that indicates the ratio of the number of V-UEs that received a specific BV-CAM to the number of all V-UEs within the BV-CAM's communication range. V-UEs calculate eMRR for each received valid BV-CAM, which is calculated using two components: Failure Counter (FC) and Success Counter (SC).

Failure Counter is defined as the number of V-UEs not receiving a BV-CAM within the communication range of the BV-CAM. A V-UE calculates the FC for each received valid BV-CAM using its OT and the BV-CAMs' DVLs. For example, Fig. 2.6 shows that V_x recognizes which V-UEs are within V_a's communication range from an observer's perspective based on its OT. We define the list of such V-UEs as Target V-UE List (TVL) of V_a, denoted by TVL_a. At the same time, when receiving a BV-CAM from V_a, V_x becomes aware of V_a's DVL (DVL_a). By comparing TVL_a in the OT with DVL_a, V_x identifies a list of V-UE(s) that V_a did not detect, which is the Hidden V-UE List (HVL) of V_a, denoted by HVL_a. In this case, according to V_x's OT, TVL_a contains V_b, but DVL_a does not contain V_b. This means that even though V_a is within V_b's communication range, it failed to receive a valid BV-CAM of V_b. Thus, V_b's FC, denoted as FC_b, is increased by 1. By comparing the OT and DVL of the received valid BV-CAMs' TX, V_x yields its FC for all TXs of the received valid BV-CAMs.

Success Counter is defined as the number of V-UEs receiving a BV-CAM within the communication range of the BV-CAM. When a V-UE generates its own BV-CAM,

Algorithm 1 Calculation of eMRR in Beyond-Vision

Require: Observation of BV-CAM information

V_i, V_j, V_k: Presenting V-UE's ID

l_{ID}: The V-UE's ID list of valid BV-CAMs

Initialize:

Initializing Failure Counter (FC) and Success Counter (SC) for all ID to 0

Counting FC and SC

for V_i **in** l_{ID} **do**
    Create TVL_i based on OT
    Extrcat DVL_i from BV-CAM of V_i
    $HVL_i \leftarrow TVL_i \cap DVL_i^C$
    for V_j **in** HVL_i **do**
        $FC_j \leftarrow FC_j + 1$
    end for
    for V_k **in** DVL_i **do**
        $SC_k \leftarrow SC_k + 1$
    end for
end for

Calculating eMRR

for V_i **in** l_{ID} **do**
    $eMRR_i \leftarrow \frac{SC_i}{FC_i+SC_i}$
end for

it records the V-UEs' IDs in its DVL that are contained in the received valid BV-CAMs in its communication range. Therefore, we obtain the SC of a V-UE by counting the number of valid BV-CAMs containing a DVL that records the V-UE's ID.

According to the values of FC_i and SC_i, where i is the BV-CAM's ID of V_i, we calculate the eMRR as

$$eMRR_i = \frac{SC_i}{FC_i + SC_i}. \tag{2.2}$$

To select a BV-CAM for relaying in Beyond-Vision, a V-UE needs to find out the eMRR

of each received valid BV-CAM and identify which BV-CAM has a low eMRR.

Algorithm 1 shows the pseudo code to calculate eMRR from OT and DVL.

Weighted Random Selection

A V-UE selects a BV-CAM for relaying in Beyond-Vision based on the eMRR of each received valid BV-CAM. However, to prevent the same BV-CAM from being selected by multiple adjacent V-UEs at the same time, the V-UE does not simply select the BV-CAM with the lowest eMRR. Instead, the V-UE selects a BV-CAM according to the selection probability using its eMRR as a weight parameter. The probability of selecting a BV-CAM is calculated as

$$P_i = \frac{1 - eMRR_i}{\sum_{k \in c_{\mathrm{id}}} (1 - eMRR_k)} \tag{2.3}$$

where P_i is the probability of selecting V_i's BV-CAM for Beyond-Vision relaying and c_{id} is the set of V-UE IDs in the candidate list for relaying.

2.4.3 Resource Selection for Beyond-Vision Relaying

When a V-UE selects a BV-CAM for Beyond-Vision relaying, it selects RBs to send the selected BV-CAM. To comply with the standard C-V2V defined in 3GPP, RBs for Beyond-Vision relaying are allocated according to the sense-based SPS operation. The V-UE analyzes energy levels for the duration of 1000 ms before the BV-CAM is selected for relaying. Through the process, the V-UE extracts candidate RBs from resources with the lowest 20% received energy levels. However, for the relayed BV-CAM to be valid, it must be sent before it expires with the generation of a new BV-CAM. Therefore, the V-UE randomly chooses RBs within the BV-CAM's generation time plus 100 ms ($t_{\mathrm{gen}} + 100$ ms) for BV-CAM relaying within a valid period.

We design Beyond-Vision to take this aspect into account as it is necessary to inhibit redundant re-transmission of BV-CAMs through relay operation. When transmitting in Beyond-Vision, a V-UE duplicates the BV-CAM selected for relay and records its flag

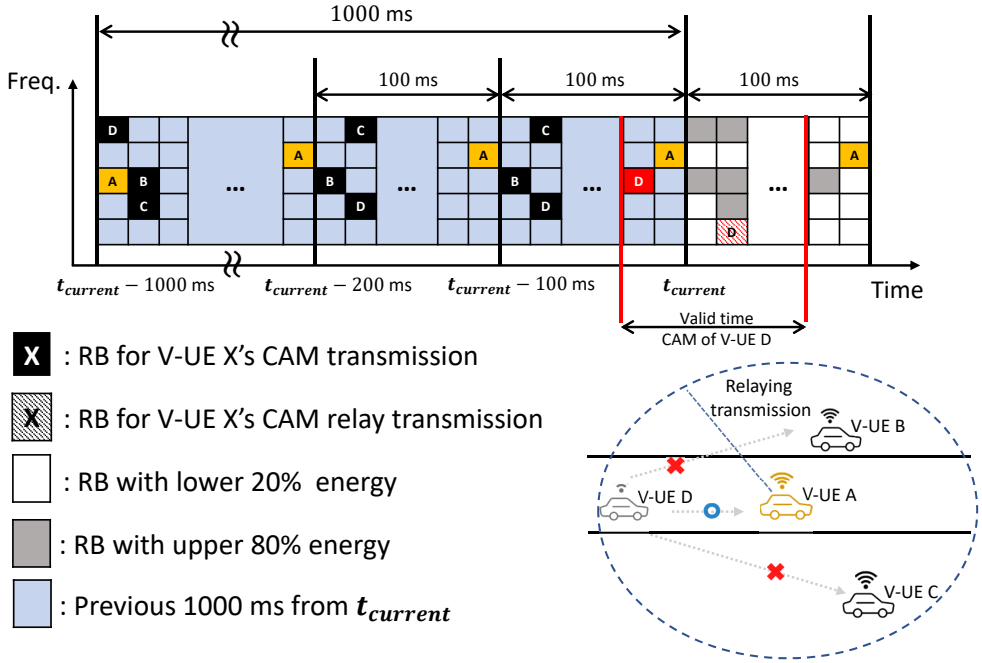


Figure 2.7: Relaying resource allocation in Beyond-Vision.

bit to indicate that the BV-CAM is duplicated. Through this flag bit, the other V-UEs receiving the BV-CAM can find out whether it is the original or a duplicate. To prevent unnecessary re-transmissions, a V-UE removes a duplicate BV-CAM from the candidate list where the BV-CAM is chosen for relaying transmission. Furthermore, if the V-UE has already scheduled the BV-CAM for relaying transmission before receiving its duplicate, it cancels its transmission schedule and selects a new BV-CAM for relaying again.

2.5 Performance Evaluation

In this section, we evaluate the performance of Beyond-Vision with the comparison schemes, through the simulation that reflects realistic vehicle mobility and the road environment in urban scenarios.

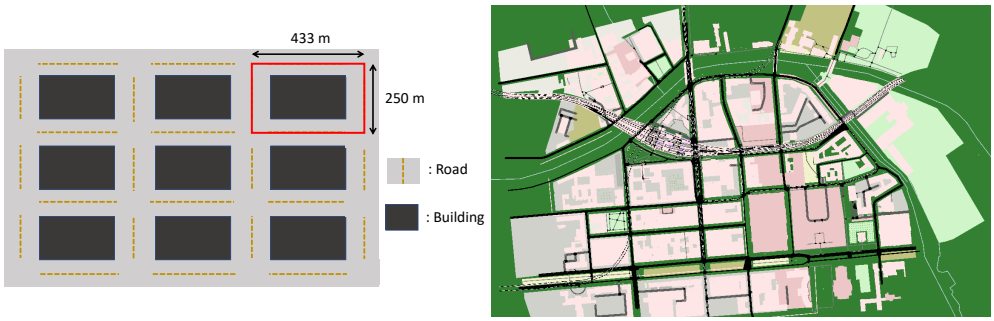
Table 2.2: Simulation environments

Carrier frequency	5.9 GHz
System bandwidth	10 MHz (50 RBs)
Topology	Manhattan grid [32] and Berlin
Target communication range	150 m
No. of total V-UEs	500, 200 (Manhattan, Berlin)
Vehicle mobility model	SUMO [6]
Link performance model	LTE error model [33]
Channel model	Fast fading + shadowing + pathloss + in-band emission [32] + out-of-band emission [34]
Modulation	QPSK
Code rate	0.529
TX power of V-UE	23 dBm
Noise figure	9 dB
Noise power	-174 dBm/Hz
CAM size	300 bytes
CAM generation period	100 ms
Simulation time	50,000 subframes (50 s)

2.5.1 Simulation Environments

Table 2.2 shows the parameters for simulation environments.

Topology and vehicle mobility model As shown in Fig. 2.8, we consider Manhattan grid and Berlin topologies for simulation in this paper. Manhattan grid topology, which is typically used for urban scenarios [32], includes a total of nine 433 m $\times$ 250 m-sized grids. We adopt Berlin topology to reflect the actual mobility of vehicles. SUMO provides OpenStreetMap (OSM) [35], which applies realistic map information to our simulator. Manhattan grid and Berlin topologies have traffic lights installed at each intersection, and use SUMO-generated mobility models [6]. SUMO helps to create real road environments, including vehicles' movement considering traffic lights linked to the actual map information provided by OSM. The number of V-UEs determined as the medium traffic case in [32] is 500 in the Manhattan grid scenario while it is 200 in



(a) (b)

Figure 2.8: Simulation topology: (a) Manhattan grid and (b) Berlin.

the Berlin scenario to achieve the equal density level of V-UEs.

Channel model The simulator adopts WINNER+ B1 model as the pathloss model [36] and the shadowing model in [32], which follows a log-normal distribution with 3 dB and 4 dB standard deviations for LOS and NLOS, respectively. ITU-R IMT UMi model in [37] is used for fast fading. For in-band emission, undesired emission to subchannels under the same channel and time slot, we adopt the model in [38].

Link performance model We choose a proven error model of LTE data transmission from [33], which is also used by an established open-source simulator in the network and communications field, ns-3 [39]. The conversion of Signal-to-Interference-plus-Noise Ratio (SINR) based on the channel model to Transmission BLock Error Rate (TBLER) enables the simulator to determine whether the message reception is successful.

Configuration CAM resources in C-V2V In DSRC, Quadrature Phase-Shift Keying (QPSK) and code rate of 0.5 are the optimal option [40] for CAM transmission. Since we use QPSK and code rate of 0.529 (i.e., closest to the optimal rate in the LTE environment), one RB can contain 177 bits. Therefore, to transmit a CAM size of 300 bytes, 15 RB pairs form one subchannel. Assuming that there are 50 RBs in the 10 MHz bandwidth, 3 $(= \lfloor 50/15 \rfloor)$ subchannels are available.

2.5.2 Comparison Schemes

This paper adopts various comparison schemes to prove the excellence of Beyond-Vision. They are 802.11p-based DSRC protocols which are modified to operate in the C-V2V standard for fair comparison. The comparison schemes

First, Farthest-First Relaying (FAR) is the most representative relaying scheme, studied in several papers [14–16]. It allocates wait time for relaying transmission to be inversely proportional to the distance between the TX V-UE and the RX V-UE. As a result, a V-UE relays the CAM received from the farthest first. To prevent unnecessary re-transmission, the V-UE waits until the wait time ends and transmits the CAM unless it receives a relayed CAM during this period.

Second, another scheme is Probability-based Relaying (PR). This method, proposed in [18], considers the density of nearby V-UEs in determining the relaying probability. The relay probability is calculated in the number of V-UEs around a TX V-UE and as the number of the V-UEs increases, the relay probability decreases. The V-UE does not cancel the scheduled relaying when it receives an already-relayed CAM, but it prevents redundant transmission by stochastic relay transmission.

Finally, no relaying scheme (NR) is the baseline protocol of C-V2V in the 3GPP standard [8, 9]. In NR, V-UEs or any other objects such as Road Side Units (RSUs) do not relay CAMs.

2.5.3 Performance Metrics

Message reception ratio The MRR is a basic metric for performance evaluation which indicates CAM reception ratio of V-UEs within the communication range of the TX V-UE. In Fig. 2.9, for example, the MRR is $5/7$ because five V-UEs succeeded while two V-UEs failed in receiving the CAM. To reflect various MRR indexes, we consider not only overall MRR performance, but also MRR performance in NLOS. Since we evaluate performance in urban environments, we set the communication range at 150 m.

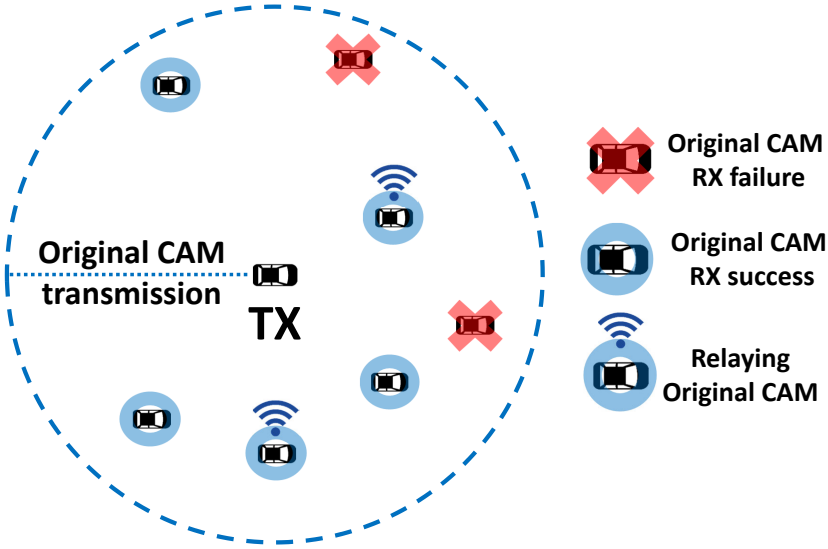


Figure 2.9: Message reception rate and relaying ratio.

Average value of lower MRR We obtain the MRR of each CAM and calculate the average the lowest 10% and 20% MRRs. In this way, we can see whether the MRR of each CAM that was not successfully transmitted via relaying protocols improves. This paper reveals the average of lower MRR in each comparison scheme.

Relaying ratio Relaying ratio is defined as the ratio of the number of V-UEs that relayed the original CAM to the total number of V-UEs that received it. To relay a CAM, a V-UE first should receive the original CAM. The V-UE that succeeded in receiving the original CAM becomes a relaying seed. In the case of Fig. 2.9, five V-UEs become relaying seeds since they received the original CAM. On the other hand, the number of V-UEs relaying the original CAM is 2. Thus, the relaying ratio is $2/5$. In this paper, we verify the relaying ratio relative to the MRR of original CAM transmission under each scheme.

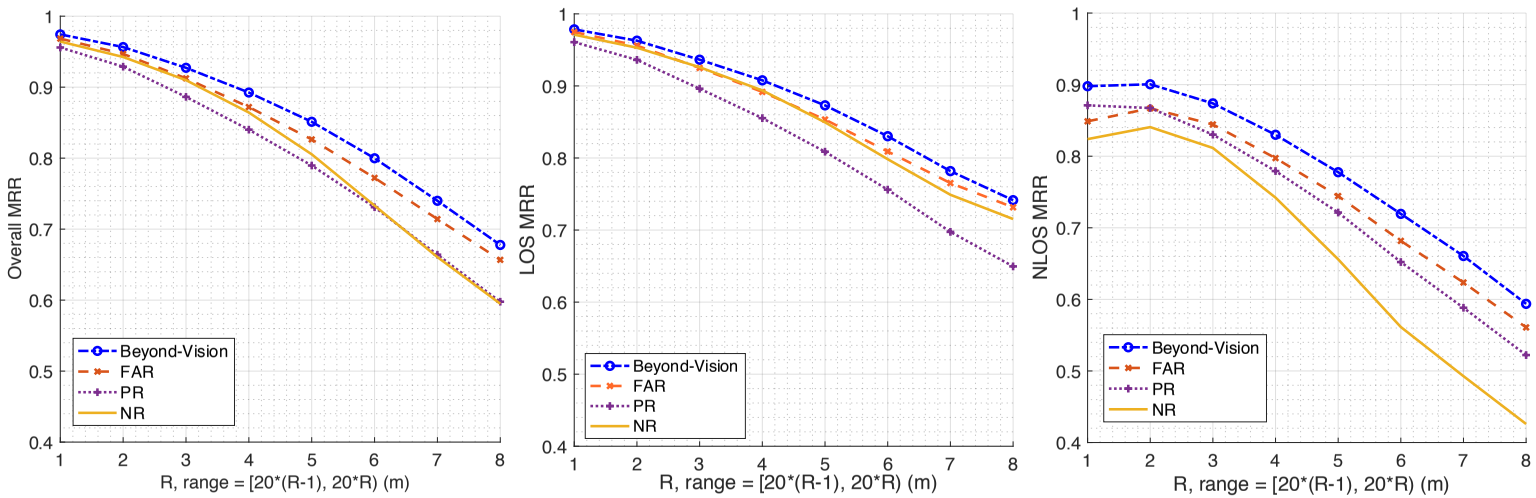


Figure 2.10: Ranged MRR performance in Manhattan grid topology: (a) Overall MRR, (b) LOS MRR, and (c) NLOS MRR.

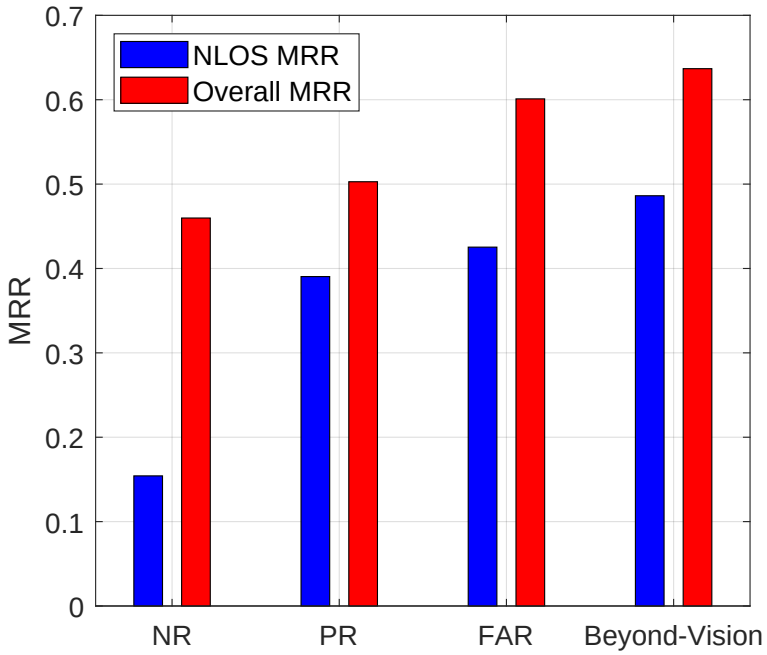


Figure 2.11: MRR performance in a range close to the communication range in Berlin topology.

2.5.4 Simulation Results

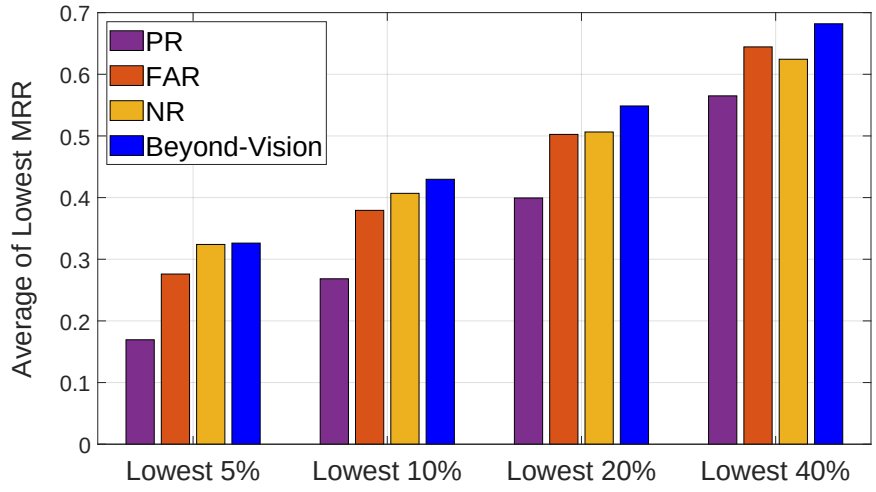
Fig. 2.10 shows MRR performance in the Manhattan grid topology. The graphs in the figure show how MRR performance varies with the distance between the TX V-UE and the RX V-UE, denoted as R. Fig. 2.10a represents overall MRR which incorporates all MRR values, when the TX V-UE and RX V-UE are in the LOS or NLOS position. Figs. 2.10b and 2.10c show LOS MRR and NLOS MRR, respectively. These graphs show that MRR performance degrades with the distance. Beyond-Vision outperforms the other schemes in terms of MRR performance.

In Fig. 2.10b, LOS MRR shows a similar pattern to overall MRR. In particular, the MRR performance of PR is lower than that of NR for the following reason. Although the V-UE under PR does not cancel the scheduled relaying when receiving the same duplicate CAM, it relays duplicate CAMs by the probabilistic manner as to prevent

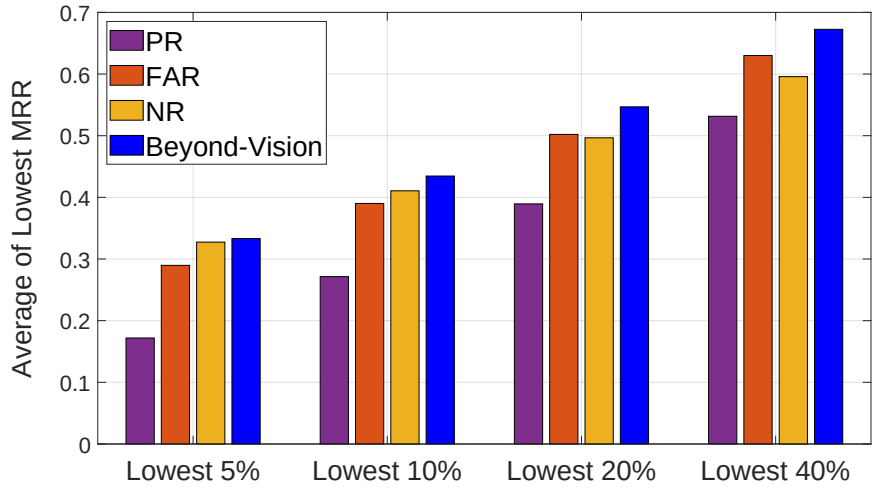
redundant transmission. Therefore, in high MRR environments, as in the case of LOS, redundant relaying of original CAMs is more likely to occur. Such unnecessary relaying causes resource collision, degrading MRR performance.

On the other hand, we can see in Fig. 2.10c that NR shows the worst performance and its performance significantly deteriorates with R. In the NLOS case, the relaying schemes improve MRR, and Beyond-Vision is the most effective of all. FAR shows better performance than PR since it does not relay the same CAM it has received before.

Fig. 2.11 shows the MRR performance in the Berlin topology. As confirmed previously, the MRR performance decreases with the distance between the TX V-UE and RX V-UE. Given that 3GPP sets the communication range in the urban environment as 150 m, we verify the MRR performance in the range [140 m, 160 m). As in the Manhattan grid topology, we can confirm that NR shows severe MRR performance degradation in the NLOS case while the relaying schemes improve MRR performance. Again, Beyond-Vision outperforms its competitive schemes in MRR improvement.



(a)

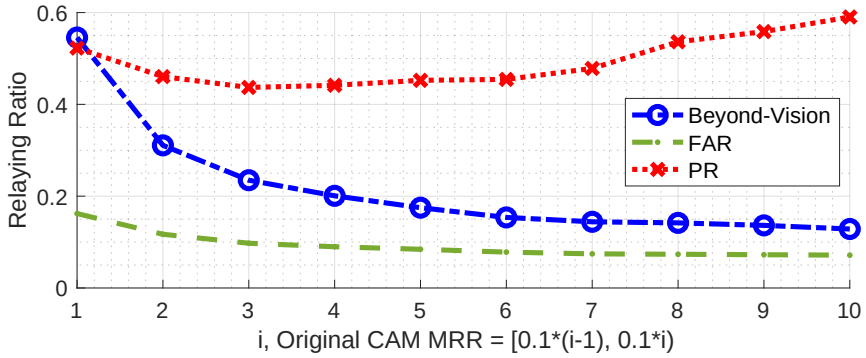


(b)

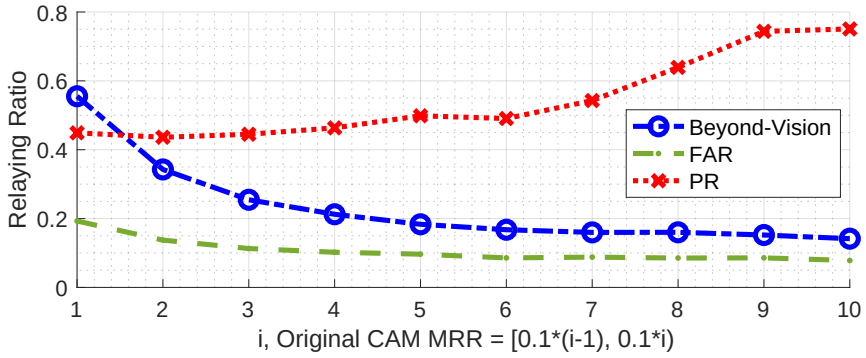
Figure 2.12: Lowest MRR Average: (a) Manhattan grid and (b) Berlin topology.

Unlike the previous evaluation, Fig. 2.12 compares MRR performance regardless of the distance between the TX V-UE and RX V-UE. It shows the average MRR of CAMs for the lowest 5%, 10%, 20%, and 40% MRR levels regardless of the distance. From the results, we see how much improvement Beyond-Vision makes for CAMs

with low MRR. Fig. 2.12 summarizes the simulation outcomes for all cases under the two topologies, where we observe the same patterns. Note that the average MRRs in the lowest 5%, 10%, and 20%-MRR-CAM group are lower under the schemes of PR and FAR than under NR. Only after exceeding the lowest 20%-MRR-CAM group, FAR shows performance similar to or greater than NR. This indicates that the other comparison schemes do not efficiently improve MRR performance for low-MRR-CAM groups. We confirm that Beyond-Vision is the only relaying scheme that improves MRR performance for the low-MRR-CAM groups by selectively relaying CAMS with low MRR.



(a)



(b)

Figure 2.13: Relaying ratio: (a) Manhattan grid and (b) Berlin topology.

Fig. 2.13 represents the relaying ratio relative to original CAM MRR. The original CAM MRR indicates the MRR for original CAM transmission before any relaying occurs. As described before, V-UEs that received original CAMs become relaying seed V-UEs enabled to relay CAMs. Thus, the relaying ratio represents the ratio of V-UEs that relayed CAMs to relaying seed V-UEs. In Fig. 2.13, x-axis indicates that the range of MRR for the original CAM. In Beyond-Vision, the probability that a V-UE relays a CAM is higher when the CAM is less likely to be received at surrounding V-UEs. In addition, the V-UE does not relay duplicated CAMs. Therefore, even when there are many relaying seeds due to high original CAM MRR, the relaying operation in other relaying seeds is effectively suppressed.

On the other hand, the relaying ratio of PR is the highest in almost all sections of x-axis. In particular, the graphs in Figs. 2.13a and 2.13b show that PR is more likely to relay CAMs with a higher MRR of the original CAM. Redundant relaying can occur in PR that determine relay operation in a stochastic manner. As a result, the higher the original CAM MRR, the greater the proportion of V-UE available for relaying, resulting in more redundant relaying.

Finally, FAR yields the lowest relaying ratio in all ranges of CAM MRR for two reasons. First, in FAR, the V-UE does not relay duplicate CAMs if it has already received the same CAMs before. Second, in our adoption of FAR to C-V2V, the V-UE sometimes fails to schedule CAM relaying due to lack of available subchannels within a determined wait time. Compared with Beyond-Vision, FAR shows less significant inverse-proportional relation between relaying ratio and original CAM MRR, which demonstrates that Beyond-Vision relays CAM more effectively from the perspective of MRR improvement.

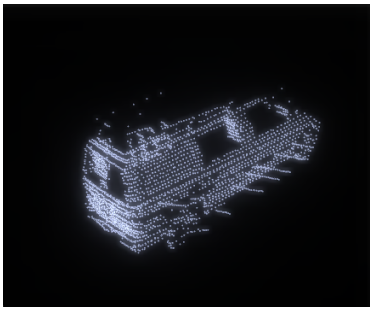
Chapter 3

UpCycling: Semi-supervised 3D Object Detection without Sharing Raw-level Unlabeled Scenes

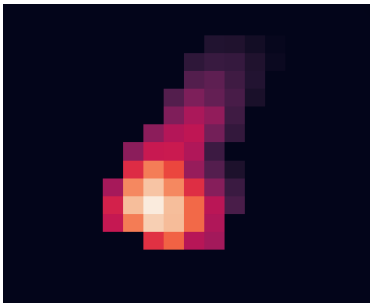
3.1 Introduction

Although the concept of Autonomous Vehicles (AVs) has been around for years, ensuring the safety of users driving AVs on real roads via 3D object detection models is still challenging. To this end, there have been continuous efforts to collect large datasets of 3D road scenes and annotate them carefully [1–3]. While rapid advances in sensor technology facilitate the collection of 3D scenes at scale, the severe *annotation burden* remains as a main challenge. To alleviate the problem, a couple of semi-supervised learning (SSL) methods for 3D object detection have been proposed recently, such as a combination of perturbation and consistency loss [41] and confidence-based filtering using IoU prediction results [42].

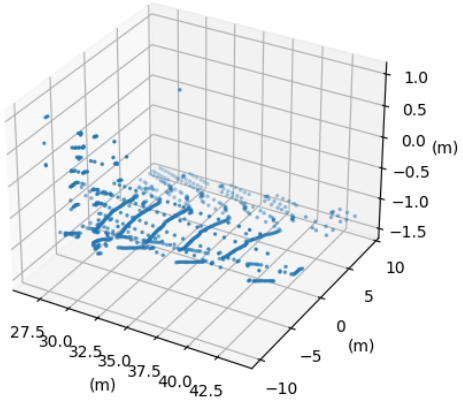
However, these methods learn from unlabeled raw 3D scenes. Collecting a vast amount of raw-level road scenes from AVs can potentially cause disclosure of sensitive private information on the roads [43–45]. Moreover, the demand for privacy-preserving domains is rapidly accelerating. The EU's General Data Protection Regulation requires firms to implement data protection measures, safeguarding consumers' privacy. This



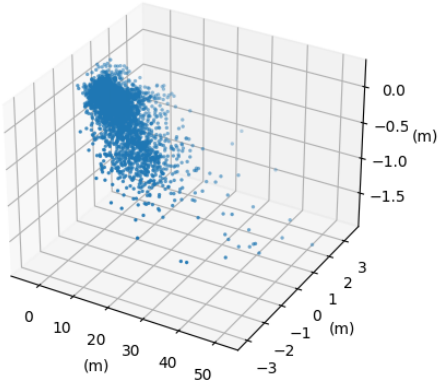
(a) Raw-point data



(b) Feature data produced from the 3D object detection network



(c) Original point cloud scene



(d) Restored point cloud scene using the inversion attack

Figure 3.1: Visualization of point cloud scenes. UpCycling improves level of privacy protection since an original point cloud scene cannot be restored from its intermediate feature.

applies even to companies collecting autonomous driving data [46]. In addition, as 2D images can be restored from limited 3D data [47], it's critical to fundamentally secure raw 3D point data.

Given that the problem of potential *privacy leakage* from raw data collection exists in various applications, a number of studies have tried to not deal with raw data directly. Going beyond encrypting raw data [43], federated learning [48, 49] makes each edge node consume its data locally to train the model and share the model weights (or gradients) instead of raw data. Split learning [49–51] designs edge nodes to not share raw data but its intermediate feature (*i.e.,*, smashed data) that comes from passing through early-stage layers of the model. However, these approaches require local training [52, 53], which makes resource-constrained AVs suffer more *computation overhead*. Given that AVs use significant computing resources to process inference pipelines for 3D detection during driving, such additional computation hinders continuous model updates in natural driving conditions.

In this paper, we aim to address all the three issues: labeling cost, privacy, and AV-side computation overhead. To ensure this end, we propose UpCycling, a novel SSL framework that does not utilize unlabeled raw 3D scenes (Figure 3.1(a)) but *de-identified, unlabeled* intermediate features (Figure 3.1(b)) to advance 3D object detection models. Since an unlabeled intermediate feature is naturally produced during a regular detection pipeline with the 3D scene, UpCycling requires neither additional AV-side computation (*e.g.,*, local training) nor server-side annotation burden. Further, sharing features instead of raw 3D scenes improves the level of privacy protection as the detection pipeline includes nonlinear layers and compression [54–58]. Because the process in the nonlinear layers [59] is irreversible, the original scene cannot be completely restored from its intermediate feature. As depicted in Figures 3.1(c) and (d), the inversion attack [60] attempted on the server side to restore the raw-point data does not result in a successful restoration.[1]

[1]For further details, please refer to Section 3.5.3 and Supplementary material where more comprehen-

To realize the advantages, UpCycling should provide an effective feature-based SSL method for 3D object detection, which involves two challenges: (1) augmenting unlabeled intermediate features reliably to increase data diversity [61, 62] and (2) providing high-quality pseudo labels to supervise these augmented features. The state-of-the-art (SOTA) semi-supervised 3D object detection frameworks [41, 42] generate consistency loss between weak and strong augmentations of a 3D point scene. However, the augmentation methods targeting raw-level point clouds become detrimental when applied at a feature level. This is because an intermediate feature is a smashed form of its original 3D scene and has multiple types depending on the 3D object detection models, such as grid- and set-types. Therefore, naïve application of the point augmentation methods at a feature level damages the important information in the 3D scene, which causes the pseudo labels to suffer from significant noise.

To address the challenges, we propose high-quality *hybrid pseudo labels* and feature-level ground-truth sampling (*F-GT*). Combining these methods not only achieves significant data diversity but also improves quality of pseudo labels by adding zero-noise labels. We implement UpCycling on two representative 3D detection models, PV-RCNN [58] and SECOND-IoU [63],[2] and perform various experiments on three major datasets for AV applications, KITTI [2], Lyft [3], and Waymo [1]. The results demonstrate the effectiveness of UpCycling in both partial-label and domain adaptation scenarios.

The contributions of this work are summarized as follows:

- UpCycling is the first framework that tackles labeling cost, privacy leakage, and AV-side computation cost altogether to train a 3D object detection model, which deeply investigates how to learn from unlabeled intermediate features.
- UpCycling provides a fresh eye on GT sampling in the context of SSL since it safely improves data diversity of unlabeled feature-level 3D scenes and significantly

sive information is provided.

[2]SECOND-IoU adds an IoU module to the original SECOND model [55].

improves pseudo-label quality by providing zero-noise labels.

- UpCycling not only protects privacy but also performs better or comparably to the SOTA methods in both domain adaptation and partial-label scenarios, on representative models and datasets for 3D object detection.

3.2 Related Work

Semi-supervised learning. SSL has been actively studied in the context of image classification [62, 64–66]. Most of the recent SSL methods [61, 62, 64, 66] leverage consistency regularization which trains the model to obtain consistent prediction results across label-preserving data augmentation. In the SSL frameworks, proper data augmentation is essential, which should significantly increase diversity effect without losing consistency with the original data [67, 68]. Accurate pseudo-labeling is another crucial element for SSL to provide high-quality supervision for unlabeled data [65, 69]. While there have been only a couple of studies on SSL for 3D object detection [41, 42], data augmentation and pseudo-labeling are still important. SESS [41] targets indoor 3D object detection, leveraging a teacher-student architecture that takes differently augmented 3D scenes as inputs and utilizes three kinds of consistency losses between outputs. 3DIoUMatch [42] improves quality of pseudo labels with confidence-based filtering in the IoU-guided NMS stage. However, the SSL methods require direct access to a vast amount of raw data, which causes potential privacy leakage.

Feature-level data augmentation. Data diversity can be limited when augmenting only raw data. To further increase diversity, feature-level data augmentation has been investigated [70–74]. In image classification tasks, adding Gaussian noise to feature-level data gains more data diversity for training and domain generalization [70]. The work in [71–73] resolves lack of data for specific classes by using feature augmentation. Feature augmentation is also applied to few-shot learning in NLP tasks [75]. To our knowledge, however, feature-level augmentation has not been studied in the context of

semi-supervised 3D object detection.

Private representation learning. Private representation learning [48, 49] aims to learn from various clients without sharing their raw data, which heavily relies on local training at resource-constrained clients. Federated learning designs clients to not share any data but model weights or gradients with the server. Due to the local computation burden for training the whole model, federated learning methods [76–78] face significant hurdles in training large neural nets. Split learning [49–51] is more similar to UpCycling in that clients share intermediate features of local data with the server. However, it still requires local training of early layers of the model. Continuous communication burden during training is another problem of these approaches.

3D object detection models. Main challenges in 3D object detection come from the irregular and sparse positions of 3D point clouds. To address the issues, some researches [79, 80] opt for point-based methods that extract set-type features by processing raw point clouds directly [81]. Other approaches [54–56, 58] suggest voxel-based methods, which first voxelize a point cloud and extract grid-type features with 3D convolution networks. Therefore, UpCycling should be able to handle both grid- and set-type unlabeled features. Specifically, we adopt two representative 3D object detectors: voxel-based SECOND-IoU [55, 63] and PV-RCNN [58] that mixes point- and voxel-based methods.

3.3 Method

3.3.1 Problem Definition

Given a 3D point cloud scene $\mathbf{x}$, we aim to detect a set of 3D bounding boxes and class labels for all objects in $\mathbf{x}$, denoted as $\{\mathbf{y}\}$. We perform this task under a new challenging SSL scenario with unlabeled de-identified data: in contrast to the regular SSL setting, unlabeled raw-level point clouds are not available. Specifically, we have access to N training samples, including N^l labeled point clouds $\{\mathbf{x}_i^l, \{\mathbf{y}_i^l\}\}_{i=1}^{N^l}$ and N^u

unlabeled scenes in the form of *intermediate feature* $\{\mathbf{f}_i^u\}_{i=1}^{N^u}$. Here $\mathbf{f}^u$ is the output of the backbone network for an unlabeled point cloud $\mathbf{x}^u$.

3.3.2 UpCycling Framework

Figure 3.2 depicts the overall UpCycling framework incorporating server- and AV-side operations. For initialization, the server trains a 3D object detection model on its labeled data $\{\mathbf{x}_i^l, \{\mathbf{y}_i^l\}\}_{i=1}^{N^l}$ and shares the pre-trained model with AVs. UpCycling targets the latest 3D detection models with an IoU module that returns *confidence scores* for bounding box localization. In this paper, we apply UpCycling in PV-RCNN [58] and SECOND-IoU [63]. PV-RCNN is the representative IoU-aware model for 3D object detection and SECOND-IoU is a modified version of SECOND [55] with addition of IoU module.

For autonomous driving, AVs continuously perform the model's detection pipeline for newly observed 3D scenes. At the same time, to further update the model with more 3D scenes in diverse environments, each AV sends a new 3D scene $\mathbf{x}^u$'s intermediate feature $\mathbf{f}^u$ to the server, which serves as *de-identified unlabeled training data*. It is noteworthy that *zero additional computation* is needed for the de-identification since the feature naturally comes from processing the 3D backbone network in the detection pipeline. Each AV also sends the detection results $\{\tilde{\mathbf{y}}^u\}$ to the server.

With the received features and detection results $\{\mathbf{f}_i^u, \{\tilde{\mathbf{y}}_i^u\}\}_{i=1}^{N^u}$, the server generates consistency loss in a different way of the SOTA SSL methods on 3D object detection that utilize unlabeled raw-point scenes $\{\mathbf{x}_i^u\}_{i=1}^{N^u}$ [41, 42]. Specifically, given that supervising $\mathbf{f}^u$ by using its detection result $\{\tilde{\mathbf{y}}^u\}$ again is meaningless, (1) proper augmentation of $\mathbf{f}^u$ and (2) high-quality pseudo labels are essential.

The SOTA methods on semi-supervised 3D object detection [41, 42] take a teacher-student architecture [64] by using random sampling (RS) for weak augmentation and both RS and Flip for strong augmentation of a point cloud. However, in our scenario where an input is an intermediate feature, the augmentation methods significantly

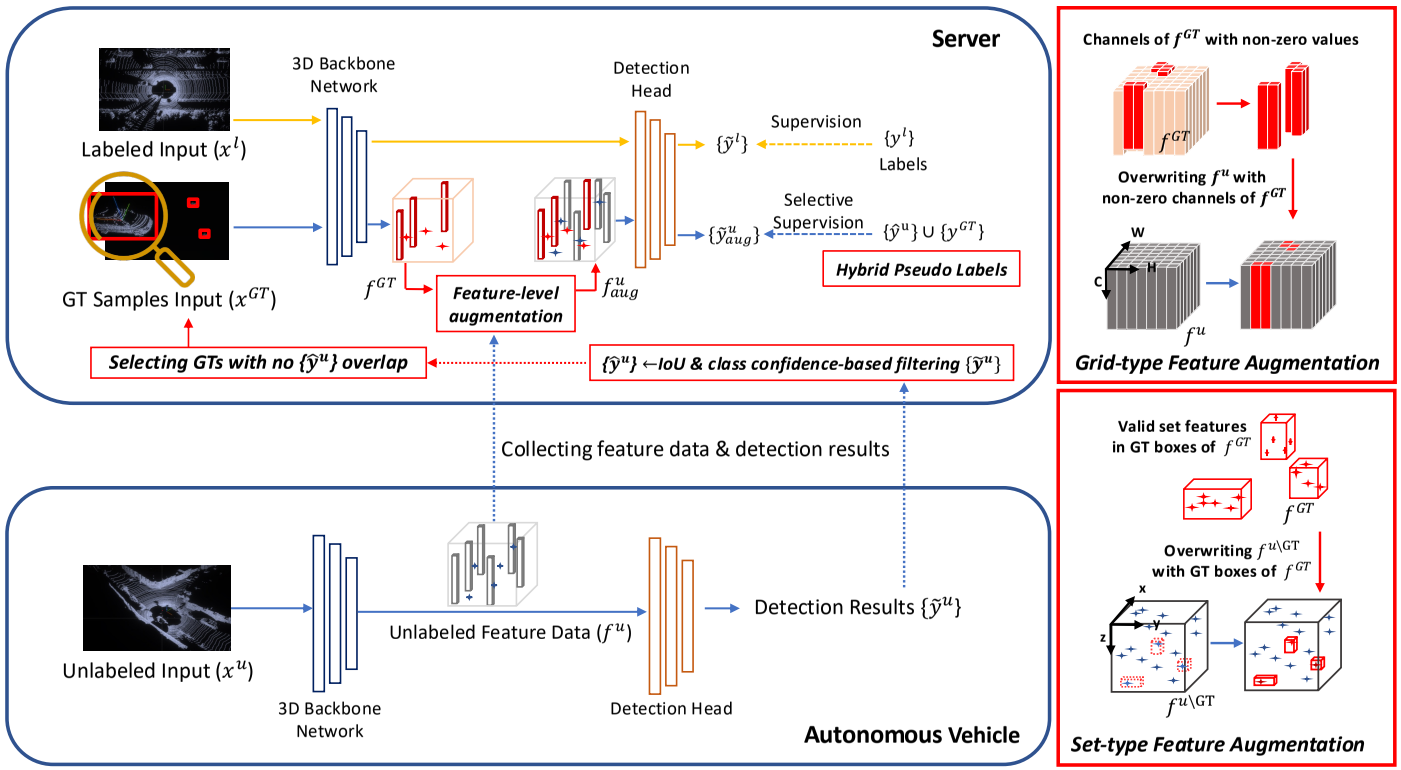


Figure 3.2: Overview of the UpCycling framework. $\mathbf{f}^u$ and $\{\tilde{\mathbf{y}}^u\}$ refer to unlabeled feature data and detection results from AVs, respectively. IoU and class confidence-based threshold filters detection results to obtain $\{\hat{\mathbf{y}}^u\}$. GTs that do not overlap with $\{\hat{\mathbf{y}}^u\}$ are sampled to form high-quality hybrid pseudo labels. To obtain data diversity, UpCycling augments the collected unlabeled feature-level data $\mathbf{f}^u$ with GT sampling (*F-GT*). The resulting augmented feature, $\mathbf{f}^u_{aug}$, is supervised by the high-quality hybrid pseudo labels.

damage the original scene. Instead, we propose feature-level ground-truth sampling (*F-GT*) for feature augmentation, as illustrated in Figure 3.2. Although ground-truth (GT) sampling has been used as a point cloud augmentation method for supervised 3D object detection [54–58] and is known to provide at most fair performance improvement [82], we claim that its impact can be more significant when it comes to *feature-level* augmentation of an *unlabeled* 3D scene. This is because *F-GT* tackles one of the most crucial issues for successful SSL: improving the quality of pseudo labels for unlabeled features by generating *hybrid pseudo labels*.

3.3.3 Hybrid Pseudo Labels

For effective SSL, we adopt *F-GT* to augment an unlabeled scene feature $\mathbf{f}^u$ and include the sampled GT labels (zero-noise labels) in the pseudo-label set for the unlabeled feature. By doing so, UpCycling constructs high quality *hybrid pseudo labels*.

Confidence-based pseudo-label filtering. First, inspired by 3DIoUMatch [42], UpCycling screens the received detection results $\{\tilde{\mathbf{y}}^u\}$ by using each $\tilde{\mathbf{y}}^u$'s confidence scores for both object classification and bounding box localization. Assume that τ_{IoU} and τ_{cls} are thresholds for box localization and object classification, respectively. UpCycling filters out a detection result if its class confidence or localization confidence is lower than the given threshold, leaving a set of high-quality pseudo labels, denoted as $\{\hat{\mathbf{y}}^u\}$. The confidence-based pseudo-label filtering is applied for more accurate supervision.

Pseudo-label-aware GT sampling. When GT sampling is applied for supervised learning, it first constructs a GT database that consists of labeled 3D bounding boxes and point clouds in the boxes, collected from the entire labeled training set $\{\mathbf{x}_i^l, \{\mathbf{y}_i^l\}\}_{i=1}^{N^l}$. To augment a labeled 3D scene $\mathbf{x}^l$, GTs are sampled from the database and randomly placed in the 3D scene. To avoid tampering with GT information, a GT sample that overlaps with a ground-truth bounding box in the original labeled scene is removed.

In contrast, our *F-GT* aims to augment an *unlabeled* 3D scene feature $\mathbf{f}^u$ without accurate box labels. Instead, given that a set of high-quality pseudo labels $\{\hat{\mathbf{y}}^u\}$ is

provided, *F-GT* samples GTs that do not overlap with the *pseudo labels*. Importantly, although the pseudo labels are filtered with the two thresholds τ_{IoU} and τ_{cls}, these thresholds are set moderately [42], enabling the pseudo labels to cover most objects in the original scene $\mathbf{x}^u$; GT samples are likely to be placed on the background of $\mathbf{x}^u$.

Hybrid pseudo-labels. To generate pseudo labels that supervise an augmented unlabeled feature $\mathbf{f}^u_{aug}$, UpCycling merges the high-quality pseudo-label set for the original feature $\mathbf{f}^u$, $\{\hat{\mathbf{y}}^u\}$, with the label set for the GT samples, $\{\mathbf{y}^{GT}\}$, resulting in a set of *hybrid pseudo labels* $\{\hat{\mathbf{y}}^u\} \cup \{\mathbf{y}^{GT}\}$. Given that $\{\mathbf{y}^{GT}\}$ are literally ground-truth labels with *zero noise*, adding these labels to the pseudo labels enables powerful supervision. Furthermore, generating the hybrid pseudo labels does not need to execute the inference pipeline at the server, since all GT labels are already given.

3.3.4 Feature-level 3D Scene Augmentation

Regarding *F-GT*, since the server does not have an original unlabeled scene $\mathbf{x}^u$ but only its intermediate feature $\mathbf{f}^u$, it is impossible to directly place GT samples on the point cloud scene. Instead, *F-GT* generates a separate point cloud input that comprises only GT samples. The GT-only point cloud passes through the model's 3D backbone network, resulting in a GT-only feature $\mathbf{f}^{GT}$. Note that while the 3D backbone of SECOND-IoU generates only grid-type features, that of PV-RCNN [58] generates both grid- and set-type features. To this end, *F-GT* augments $\mathbf{f}^u$, grid- or set-type feature, as follow:

Grid-type feature augmentation. As shown in Figure 3.2, when $\mathbf{f}^u$ and $\mathbf{f}^{GT}$ are grid-type features, *F-GT* generates an augmented feature by overwriting $\mathbf{f}^u$ with $\mathbf{f}^{GT}$; if a channel on $\mathbf{f}^{GT}$ has non-zero values, the $\mathbf{f}^{GT}$ channel replaces that in $\mathbf{f}^u$. Giving higher priority for $\mathbf{f}^{GT}$ removes some information included in $\mathbf{f}^u$. However, given that the GT samples take up a tiny portion of an entire scene (*i.e.,*, most values in $\mathbf{f}^{GT}$ are zero), only a small number of values in $\mathbf{f}^u$ are modified. In addition, the removed information in $\mathbf{f}^u$ is related to the background since the sampled GTs are not overlapped

with pseudo labels, which does not harm model training.

Set-type feature augmentation. When an unlabeled feature $\mathbf{f}^u$ and a GT sample feature $\mathbf{f}^{GT}$ are set types, each of them consists of n represented points, denoted as $\mathbf{f}^u = \{f_i^u\}_{i=1}^n$ and $\mathbf{f}^{GT} = \{f_i^{GT}\}_{i=1}^n$, respectively. In this case, as illustrated in Figure 3.2, *F-GT* generates an augmented feature as a point set, denoted as $\mathbf{f}_{aug}^u = \{f_{aug,i}^u\}_{i=1}^n$. To this end, we first exclude the scene feature points f_i^u that are in the GT boxes, generating $\mathbf{f}^{u\backslash GT}$. Then each feature point $f_{aug,i}^u$ is randomly sampled from either $\mathbf{f}^{u\backslash GT}$ or $\mathbf{f}^{GT}$.

In doing so, it is important that the scene feature contains much more information than the GT feature; for reasonable augmentation, $\mathbf{f}_{aug}^u$ should include scene feature points more than GT feature points. To determine proper sampling frequency, we utilize the information in the grid-type feature that is generated simultaneously with the set-type feature by the 3D backbone network: how many values in the grid-type feature for the scene and GT samples are non-zero. For example, if the number of grid with non-zero values in the scene and GT features (grid types) is 2000 and 50, respectively, points in the augmented feature set $\mathbf{f}_{aug}^u$ is sampled from $\mathbf{f}^{u\backslash GT}$ 400 times more than $\mathbf{f}^{GT}$.

3.3.5 Loss

The model's detection head is trained to predict the hybrid pseudo labels for the augmented feature $\mathbf{f}_{aug}^u$. Given that our target models have an IoU module as well as a Region Proposal Network (RPN), the unlabeled loss $\mathcal{L}(\mathbf{f}_{aug}^u)$ includes loss of each of the two modules as follows:

$$\begin{aligned}\mathcal{L}(\mathbf{f}_{aug}^u) =& \mathcal{L}_{loc}^{RPN}(\{\hat{\mathbf{y}}^u\} \cup \{\mathbf{y}^{GT}\}) + \mathcal{L}_{loc}^{IoU}(\{\hat{\mathbf{y}}^u\} \cup \{\mathbf{y}^{GT}\}) \\ &+ \mathcal{L}_{cls}^{RPN}(\{\hat{\mathbf{y}}^u\} \cup \{\mathbf{y}^{GT}\}).\end{aligned} \tag{3.1}$$

The exact calculation of the three terms depends on the model architecture, following the calculation of supervised loss. Assuming that a training batch consists of a set of

labeled scenes $\{\mathbf{x}^l\}$ and a set of augmented features for unlabeled scenes $\{\mathbf{f}^u_{aug}\}$, the total loss for the batch is calculated as below, where w is the unsupervised loss weight:

$$\mathcal{L}_{total} = \mathcal{L}(\{\mathbf{x}^l\}) + w\mathcal{L}(\{\mathbf{f}^u_{aug}\}). \tag{3.2}$$

3.4 Analysis on 3D Scene Feature Augmentation

In this section, we take a deeper look into subtle feature-level 3D scene augmentation. Specifically, we focus on why widely-used point cloud augmentation methods damage important information when applied at a feature level.

To this end, Figure 3.3 depicts activation heat maps of the Bird-eye View (BEV) compression module in SECOND-IoU when Flip, Rotation, and GT sampling are applied to an example 3D scene covering x, y, z axis range 70.4, 80, 4 meters. The figure shows that in the cases of Flip and Rotation, raw-level augmentation (*i.e.,*, flipping/rotating the whole point cloud) and feature-level augmentation (*i.e.,*, flipping/rotating the feature vector) result in significantly different activations. In both cases, although the two activation heat maps look similar at a glance, taking the difference between the two causes errors that are widely spread over the entire feature map. In contrast, when using GT sampling, raw- and feature-level augmentations provide similar activation heat maps. Although some errors exist, they are placed in restricted areas where GT samples are inserted.

Figure 3.4 provides a visual illustration of Flip and Rotation for feature augmentation. If a point cloud is voxelized with each voxel producing its feature value, flipping/rotating the feature vector is similar to flipping/rotating voxels. This means that point locations are shifted not individually but in groups, and the geometric relationship between intra-voxel points is maintained; they are neither flipped nor rotated. In the worst case, the group (voxel)-wise flipping causes a valid car object to break apart, making its label detrimental to training. Breaking the geometric relationship between points on the background can also cause severe misinterpretation. Similarly, the group-wise

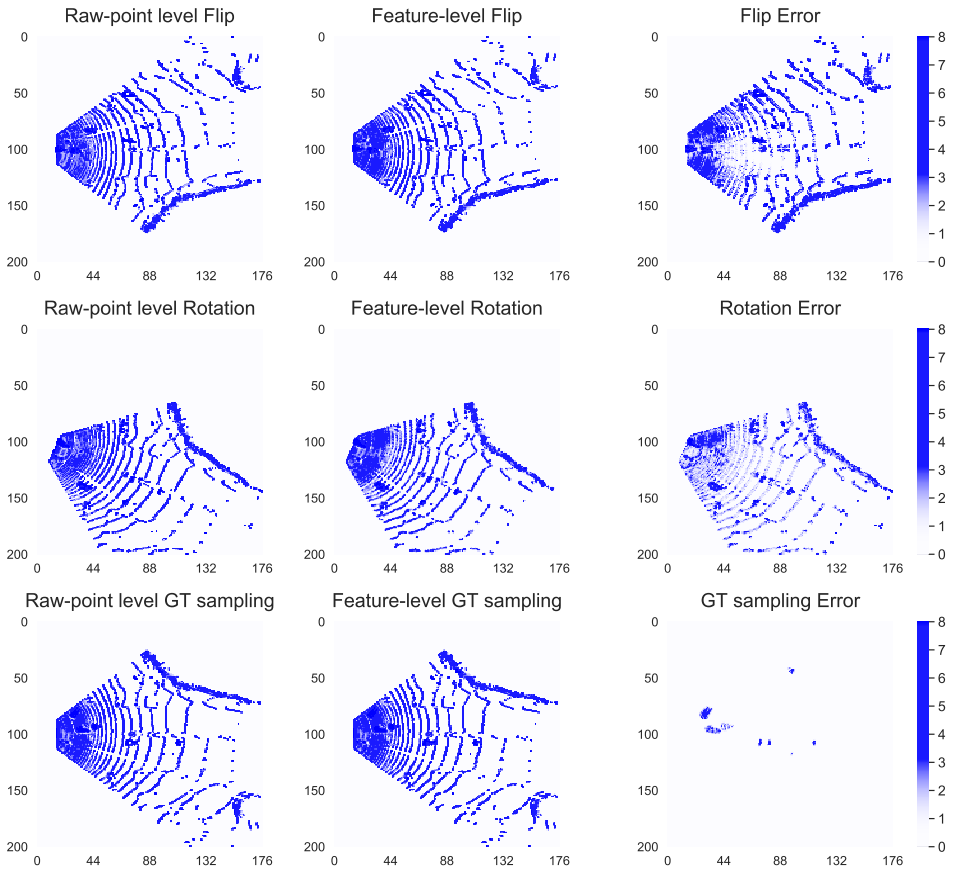


Figure 3.3: Feature-level scenes for three data augmentation methods: Flip (1st row), Rotation (2nd row), and GT sampling (3rd row). Feature-level scenes of raw-point level augmentation are on the left. Feature-level scenes of feature-level augmentation are in the middle. Heatmaps of RMSE based on comparison between raw-level and feature-level augmentation scenes are on the right.

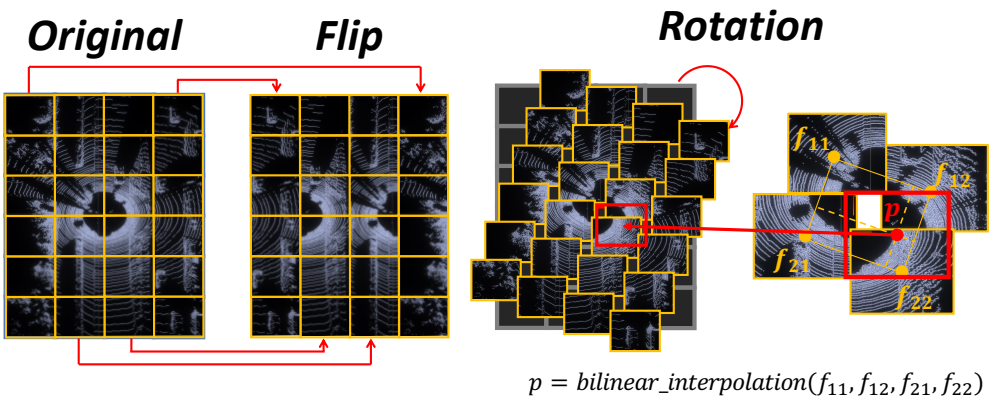


Figure 3.4: Conceptual images of feature-level augmentation with Flip and Rotation.

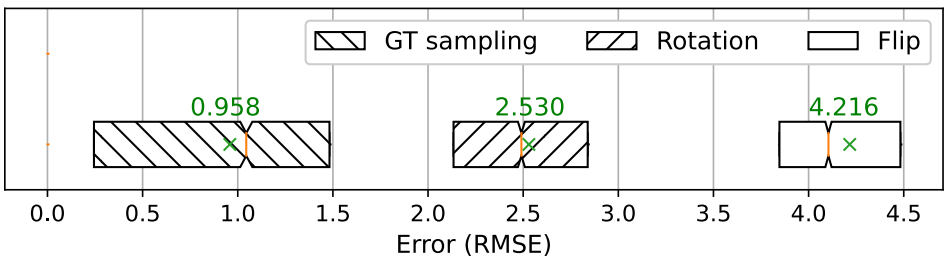


Figure 3.5: RMSE between raw- and feature-level augmentations of the entire KITTI training dataset. Box range covers the first quartile to the third quartile and the mark '×' indicates the mean value.

rotation breaks the geometric relationship mildly and its bilinear interpolation creates the errors, which is not proper for augmentation.

Figure 3.5 confirms our description by showing the average of root mean square error (RMSE) between raw- and feature-level augmentations in the KITTI dataset. This plot illustrates that feature-level Flip and Rotation severely damage the original scene, in contrast to GT sampling, which only produces minor errors.

3.5 Experiments

3.5.1 Experimental Setup

Scenarios. To demonstrate the effectiveness of UpCycling in various practical situations, we conduct experiments in both domain adaptation and partial-label scenarios. The domain adaptation task is to adapt the model, which is trained on abundant labeled data in the source domain, to an unseen target domain that provides only unlabeled data. In the partial-label scenario, the model is trained and tested in the same domain but most of the training data is unlabeled.

Datasets. We choose three datasets widely used for detection applications of AVs: Waymo [1], Lyft [3], and KITTI [2]. Among the three, the Waymo dataset is the most diverse and the largest in volume. The 3D scenes in the Waymo dataset are captured in Phoenix, Mountain View, and San Francisco, the US, under multiple weather and time settings. The Lyft dataset is collected around Palo Alto, the US, in clear weather in the daytime. The KITTI dataset is collected in Karlsruhe, Germany, in clear weather during the daytime. Due to regional characteristics, car sizes in KITTI are different from those in Waymo and Lyft [4]. We focus on car objects in this section and more details are in the supplementary material.

Implementation details. When training a model with UpCycling, we set the two filtering thresholds τ_{IoU} and τ_{cls} to 0.5 and 0.4, respectively, and the weight for the loss $\mathcal{L}(\{f^u_{aug}\})$ is set as $w = 1$. We set the ratio of labeled data to unlabeled data

Table 3.1: Effects of feature augmentation methods in a partial-label scenario where the 3D object detection model is SECOND-IoU and 10% training data is labeled in KITTI.

Policy #	*Flip*	*Noise*	*RS*	*Rot.*	*F-GT*	AP_{3D} Easy	Mod	Hard
Baseline						70.58	56.00	47.94
1	✓					−16.31	−20.09	−19.79
2		✓				+0.03	+0.13	−1.23
3			✓			+2.47	−0.96	+0.63
4∗	✓		✓			−11.69	−13.75	−13.32
5				✓		+4.80	+5.42	+7.96
UpCycling					✓	**+7.81**	**+7.87**	**+8.14**

in a mini-batch to 1:2 and 1:1 for domain adaptation and partial-label experiments, respectively. Importantly, *F-GT* samples GT boxes only from the labeled dataset: the source domain data in the domain adaptation scenario and a small portion of labeled data in the partial-label scenario. Lastly, UpCycling freezes the 3D backbone network after training it on the labeled data to prevent the divergence between an intermediate feature from the server's 3D backbone network and that collected from AVs. Therefore, UpCycling updates only the detection head using unlabeled feature-level data. More details are in the supplementary material.

3.5.2 Effect of Feature Augmentation Schemes

First, we investigate feature augmentation deeply by evaluating the superiority of *F-GT*, which is utilized for UpCycling, to other augmentation schemes in a partial-label scenario. To this end, we train SECOND-IoU on the KITTI dataset when only 10% of its training data is labeled. Importantly, given that the KITTI dataset is originally shuffled regardless of place and time sequence, we rearrange it in chronological order for each place to prevent the data leakage between the labeled and unlabeled sets [83].

Comparison schemes. In this scenario, **Baseline** trains the model using only the limited

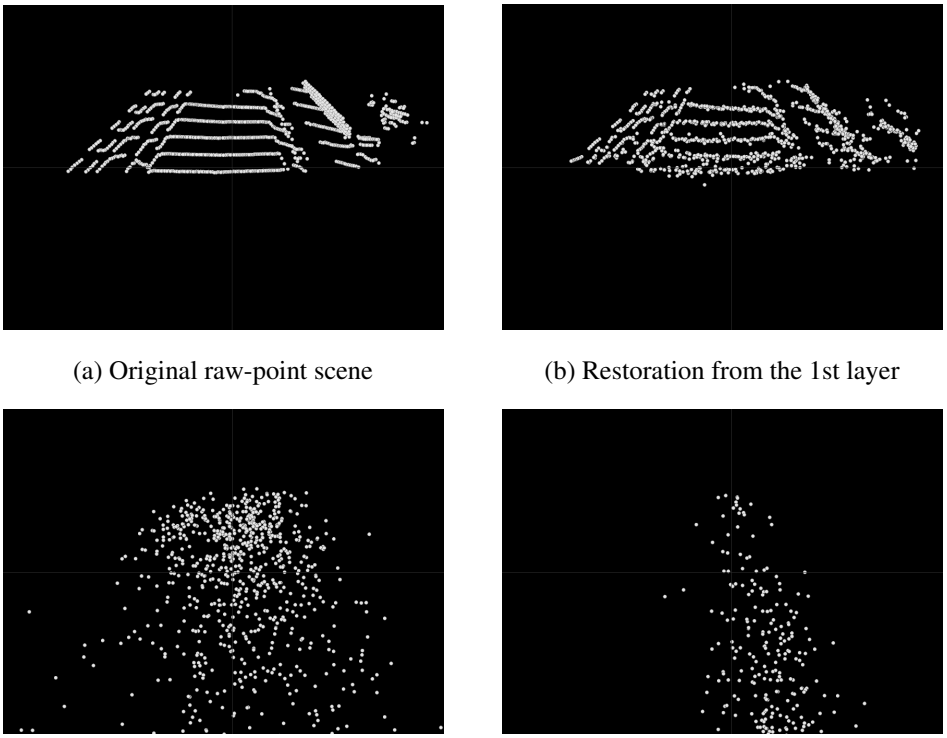
amount of labeled data. **Flip** and **RS** are used in the SOTA SSL methods on 3D object detection to augment raw-level 3D scenes [41, 42]. For feature-level Flip, we place feature information to its symmetric position on the feature map. For feature-level RS, we nullify randomly selected 5% of feature data. Combination of feature-level Flip and RS is actually a feature-level variant of the SOTA 3DIoUMatch [42], named **F-3DIoUMatch**.[3] **Noise** is an existing feature augmentation method that adds Gaussian noise, which is used for domain generalization of image classification [70]. Lastly, **Rotation** rotates the feature with a degree randomly selected from [-45°, 45°] and performs bilinear interpolation.

Result analysis. Table 3.1 shows each augmentation scheme's performance margin compared to Baseline in the partial-label scenario. Flip significantly underperforms Baseline despite the use of much more (unlabeled) training data, verifying that feature-level Flip damages important information in 3D scenes. Both Noise and RS have marginal impact on performance, showing that these perturbation strategies do not result in meaningful data diversity. Combining Flip and RS (*i.e.,*, F-3DIoUMatch) still performs worse than Baseline due to the negative effect of Flip, which confirms that naïve application of SOTA SSL methods at a feature level does not work. Although Rotation improves performance, our *F-GT* provides the lowest augmentation errors (Figure 3.5) and thus *the best performance* in all cases.

3.5.3 Privacy Protection of Feature Sharing

As neural network activations could be inverted to reconstruct input data [84–86], there could be concerns on potential privacy leaks when sharing features. We investigate whether an inversion attack can recover the grid-type feature data generated from both the SECOND-IoU and PV-RCNN backbone networks to the original point cloud. To this end, we implement the inversion attack model using the decoder method [60] that is widely used to evaluate whether a model consisting of convolutional layers can be

[3] Policy 4$*$ indicates F-3DIoUMatch.



(a) Original raw-point scene

(b) Restoration from the 1st layer

(c) Restoration from a middle (3rd) layer

(d) Restoration from the last (5th) layer, same as UpCycling

Figure 3.6: Results of inversion attack for the 3D backbone model (5 convolutional layers) of SECOND-IoU and PV-RCNN. The example 3D point cloud scene is in KITTI.

inverted [87, 88].[4] More details are in the supplementary material.

Result analysis. We conduct an inversion attack on the 3D backbone network in SECOND-IoU and PV-RCNN.[5] Figures 3.6(b)-(d) present the restoration results for intermediate features at three different convolutional layers of the backbone network: 1st, 3rd, and 5th (last) layers, respectively. While the restored point cloud from the first layer is relatively similar to the original scene (Figure 3.6(b)), it becomes significantly

[4]To the best of our knowledge, there has been no research that particularly focuses on inversion attacks for 3D point clouds.

[5]The 3D point cloud scene in Figure 3.6(a) is from KITTI dataset, and the point cloud range covers the x, y, and z-axis ranges 17.6, 20, and 4 meters.

different when applied to deeper layers' features (Figures 3.6(c) and (d)). As the number of nonlinear layers increases, it becomes more difficult to accurately restore the original data. Furthermore, restoring a point cloud from its intermediate feature is particularly challenging since each raw point needs to be positioned precisely in voxelized spaces. UpCycling utilizes unlabeled features at the last (deepest) layer, making it impossible to accurately recover the original scene from an intermediate feature. Supplementary material contains more inversion examples.

3.5.4 Domain Adaptation Experiments

Although UpCycling offers privacy protection by using only intermediate features, it is crucial to evaluate whether it provides competitive detection accuracy compared to the SOTA methods that use raw-level point clouds (Sections 3.5.4 and 3.5.5). In domain adaptation experiments, we use the Waymo dataset as the source domain and the Lyft and KITTI datasets as the target domains. The model is first pre-trained on the source domain's labeled data (called the baseline model), adapted using unlabeled training data in a target domain, and then tested on the target domain's test data.

Comparison schemes. We compare UpCycling with various methods. **Baseline** evaluates the baseline model directly and **Oracle** adapts the model with fully supervised learning in the target domain, which provide the lower- and upper-bound performance, respectively. **ST3D** [5] and **SN** (Statistical Normalization) [4] are the SOTA domain adaptation methods on 3D object detection that utilize unlabeled raw 3D scenes. ST3D generates pseudo labels from unlabeled data in the target domain to adapt the baseline model. SN assumes that statistical object sizes in the target domain are given and trains the baseline model in the source domain using the target domain object size information. We also evaluate variants of ST3D and our UpCycling by combining SN together, denoted as **(w/ SN)**.

Result analysis. Table 3.2 shows the results of UpCycling and the various comparison methods on SECOND-IoU and PV-RCNN. Surprisingly, the results show that although

Table 3.2: Domain adaptation results with two target datasets: KITTI and Lyft. Difficulty of the KITTI test dataset is set as Moderate. Baseline is a pre-trained model with Waymo whereas Oracle is trained with fully labeled target dataset.

Dataset	Method	SECOND-IoU	PV-RCNN
		AP_{BEV} / AP_{3D}	AP_{BEV} / AP_{3D}
Lyft	Baseline	30.20 / 21.32	33.00 / 24.49
	SN	28.38 / 19.25	33.44 / 25.64
	ST3D	60.53 / 29.90	62.28 / 42.63
	UpCycling	**68.83 / 45.66**	**63.38 / 46.83**
	ST3D (w/ SN)	52.86 / 21.25	60.15 / 44.02
	UpCycling (w/ SN)	**65.10 / 49.24**	**63.58 / 49.35**
	Oracle	76.70 / 61.70	78.68 / 64.54
KITTI	Baseline	54.14 / 10.16	62.24 / 9.24
	SN	60.80 / 37.30	60.08 / **38.86**
	ST3D	**70.90 / 40.16**	**66.19** / 23.26
	UpCycling	58.26 / 11.71	62.09 / 11.35
	ST3D (w/ SN)	80.97 / 57.68	54.30 / 48.79
	UpCycling (w/ SN)	**84.12 / 67.65**	**85.90 / 61.12**
	Oracle	90.36 / 82.02	90.84 / 84.56

Table 3.3: Partial-label scenario results with three portions of labeled data in the KITTI dataset: 2%, 10%, 25%.

AP_{3D}		2%			10%			25%		
		Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard
SECOND-IoU	Baseline	56.69	44.11	37.19	70.58	56.00	47.94	84.47	71.06	62.87
	3DIoUMatch	63.57	49.58	43.00	71.76	57.01	50.08	81.71	68.51	60.92
	improved (%)	12.13	12.39	15.62	1.67	1.80	4.47	-3.26	-3.59	-3.11
	UpCycling	70.19	59.97	44.83	76.09	60.41	51.84	85.22	72.87	63.93
	improved (%)	**23.81**	**35.96**	**20.54**	**7.81**	**7.87**	**8.14**	**0.89**	**2.55**	**1.69**
PV-RCNN	Baseline	68.10	53.27	46.20	81.23	68.67	60.32	87.63	76.03	68.62
	3DIoUMatch	81.04	65.77	58.83	85.26	70.64	63.32	85.08	72.37	65.02
	improved (%)	**19.00**	**23.47**	**27.34**	**4.97**	**2.87**	4.98	-2.91	-4.81	-5.25
	UpCycling	76.46	61.44	52.94	83.64	69.60	63.53	88.05	76.61	70.80
	improved (%)	12.28	15.34	14.59	2.97	1.35	**5.32**	**0.48**	**0.76**	**3.18**

UpCycling (or w/ SN) does not utilize raw-point scenes for privacy protection, it *provides the best accuracy* in most cases. Specifically, UpCycling (or w/ SN) significantly outperforms the two SOTA methods (ST3D and SN) in the Lyft case. When compared to the better option between ST3D (or w/ SN) and SN in each case, UpCycling improves accuracy by **1.3**$\sim$**19.71** $\mathrm{AP}_{\mathrm{BEV}}$ and **5.33**$\sim$**19.34** $\mathrm{AP}_{\mathrm{3D}}$. The results demonstrate the effectiveness of hybrid pseudo labels and feature-level augmentation schemes in UpCycling and also suggest the potential of using unlabeled features to advance 3D object detection models.

Taking a deeper look, SN significantly improves UpCycling performance in the KITTI dataset. Since object sizes in KITTI are different from those in Lyft and Waymo, adjusting object sizes with SN for UpCycling is effective.

3.5.5 Partial-label Experiments

In partial-label experiments, we use the same setting as in Section 3.5.2 but train both SECOND-IoU and PV-RCNN.

Comparison schemes. In this scenario, **Baseline** trains the model using only the limited

amount of labeled data. **3DIoUMatch** [42] is the SOTA SSL method using unlabeled raw-point scenes. For consistency regularization, 3DIoUMatch uses Flip and RS to augment raw data and filters pseudo labels in the IoU-guided NMS.[6]

Result analysis. Table 3.3 shows that UpCycling outperforms 3DIoUMatch in most cases by effectively utilizing unlabeled feature-level data. In the case of 25%, 3DIoUMatch even underperforms Baseline but UpCycling maintains performance improvement on both SECOND-IoU and PV-RCNN. The results are interesting because the scenario is unfavorable for UpCycling in that (1) UpCycling trains the 3D backbone only using the small portion of labeled data and (2) the effect of *F-GT* could be marginal since the number of GT samples are proportional to that of labeled data. UpCycling successfully overcomes the disadvantages, verifying that it achieves significant performance improvement even when using a relatively immature backbone network and *F-GT* effectively augments a large number of unlabeled data when only a small number of GTs are available.

3.5.6 Ablation Studies

Since UpCycling freezes the backbone during the SSL process for effective feature sharing, we evaluate the effect of the backbone freezing. To this end, we devise a comparison scheme ***UpC-R***, the application of UpCycling at the raw-level input. *UpC-R* augments a raw-level 3D scene using GT samples and trains the whole network including the backbone using unlabeled data and hybrid pseudo labels. Note that this approach not only sacrifices privacy but also takes much longer to train compared to UpCycling.

Result analysis. Figure 3.7 compares *UpC-R* and UpCycling in the partial-label scenario in Section 3.5.5. While sacrificing privacy, *UpC-R* outperforms UpCycling by training the backbone further. Interestingly, *UpC-R* performs even better than the SOTA

[6] Since the authors in [42] did not use the rearranged KITTI dataset in their experiments, we measure the performance of 3DIoUMatch again in the rearranged KITTI dataset. In addition, we newly implement 3DIoUMatch on SECOND-IoU for more extensive comparison.

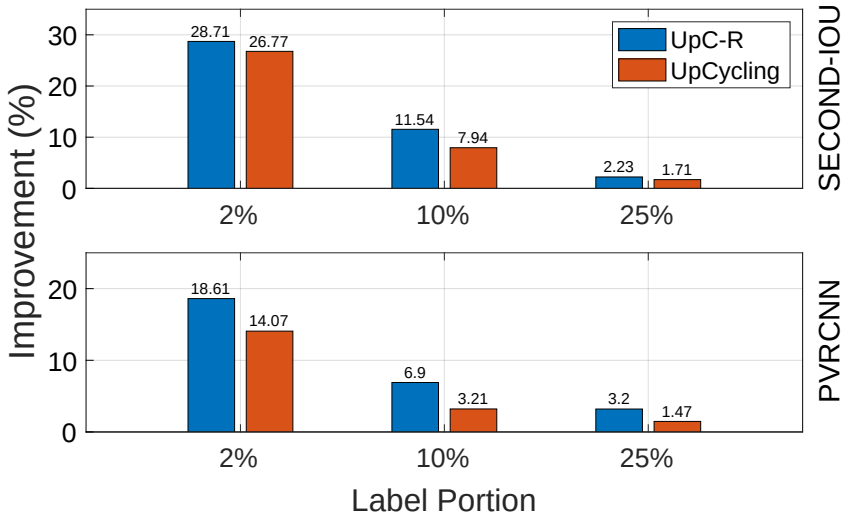


Figure 3.7: *UpC-R* vs. UpCycling: Partial-label results in the KITTI dataset. The average performance improvement in all KITTI test cases (easy, moderate, and hard).

3DIoUMatch (Table 3.3), demonstrating that GT sampling is more effective augmentation than the combination of Flip and RS *even at the raw-input level*. On the other hand, the performance gap between *UpC-R* and UpCycling decreases as the number of labeled data increases, meaning that once the backbone is well-trained, the combination of hybrid pseudo-labels and GT-based augmentation can be applied flexibly to any layer without performance degradation. We see this as the unique advantage of GT sampling that other point cloud augmentation methods cannot provide.

3.6 Implementation Details

3.6.1 Experiment settings

Training. For the pre-training stage, we train on 4 RTX 3090 GPUs with a batch size of 16 and 8 for SECOND-IoU and PV-RCNN, respectively. Then, following the original model training settings, we use epochs 80, and 30 for KITTI dataset and Waymo dataset, respectively. Especially, pre-training on small amounts of KITTI labeled data 2%, we

lengthen the epoch to 120 for the model to converge. For the semi-supervised learning stage, we train with a batch size of 32 (16 labeled + 16 unlabeled, 4 GPUs) and 16 (8 labeled + 8 unlabeled, 4 GPUs) for SECOND-IoU and PV-RCNN, respectively. We set the ratio of unlabeled data to twice that of labeled data in domain adaptation experiments. The learning rate is initialized as the value of the original model usage and updated by cosine annealing strategy.

Table 3.4: Waymo [1], KITTI [2], and Lyft [3] dataset overview. † and $*$ indicate obtaining information from [4] and [5], respectively.

	Waymo	KITTI	Lyft
LiDAR Type	64-beam	64-beam	64-beam
Beam Angles †	[-18.0°, 2.0°]	[-23.6°, 3.2°]	[-29.0°, 5.0°]
Points per Scene $*$	160,139	118,624	69,175
Training Frames	158,081	3,712	18,900
Validation Frames	39,987	3,769	3,780
Night / Rainy	Yes / Yes	No / No	No / No
Location	USA	Germany	USA

Dataset and Source Code License. We implement our UpCycling based on OpenPCDet [63] (v0.5.1) which is licensed under the Apache License 2.0. According to `https://paperswithcode.com/datasets`, the license of Waymo dataset [1] and KITTI dataset [2] is the custom (non-commercial) and the CC BY-NC-SA 3.0, respectively, and the license of Lyft dataset [3] is unknown. The details of each datasets are in Table 3.4.

3.6.2 Architecture details – 3D backbone network

In this paper, the 3D backbone network of SECOND [55] (see Table 3.5) is used for generating the grid-type feature data in PV-RCNN and SECOND-IoU experiments. Voxel Feature Extractor (VFE) converts the point cloud data into voxel format covering the entire point cloud range. After that, the output of VFE goes through the SparseConv

Table 3.5: 3D backbone network architecture generating grid-type feature data.

Layers		**BACKBONE Network**	Output size
VFE		Mean VFE	4×41×1600×1408
SparseConv Layers	conv_input	4×3×3×3×3, 16, padding 1,1,1	16×41×1600×1408
	conv_1	16×3×3×3×3, 16	16×41×1600×1408
	conv_2	16×3×3×3×3, 32, stride 2,2,2, padding 1,1,1 32×3×3×3×3, 32 32×3×3×3×3, 32	32×21×800×704
	conv_3	32×3×3×3×3, 64, stride 2,2,2, padding 1,1,1 64×3×3×3×3, 64 64×3×3×3×3, 64	64×11×400×352
	conv_4	64×3×3×3×3, 64, stride 2,2,2, padding 0,1,1 64×3×3×3×3, 64 64×3×3×3×3, 64	64×5×200×176
	conv_out	64×3×1×1×1, 128, stride 2,1,1	128×2×200×176

layers [89] where each Conv layer contains both batch normalization and ReLU, which is a non-linear function. Lastly, the output of SparseConv layers becomes the grid-type feature data which UpCycling utilizes. On the other hand, the 3D backbone network of PV-RCNN additionally generates the set-type features from Voxel Set Abstraction (VSA) (see Table 3.6). In this process, PV-RCNN samples a fixed number of keypoints from raw points following the Farthest-first rule. After that, set abstraction modules create voxel-wise features from each layer in VFE corresponding to keypoint positions. Finally, to generate the final form of set-type features, VSA Point Feature Fusion module concatenates the features from the set abstraction modules to the accurate keypoint positions.

3.6.3 Implementation Details for SECOND-IoU based 3DIoUMatch

We basically follow and reuse the official codes from the SOTA schemes for comparison except for 3DIoUMatch [42]. 3DIoUMatch method uses IoU-guided NMS modules

Table 3.6: 3D backbone network architecture generating set-type feature data.

Layers		**BACKBONE Network**		Output size
Key Point Sampling		Farthest Point Sampling		4×2048
VoxelSetAbstraction (VSA) Layers	SA_raw	radius 0.4 4×1×1, 16 16×1×1, 16	radius 0.8 4×1×1, 16 16×1×1, 16	32×2048
	SA_pv1	radius 0.4 19×1×1, 16 16×1×1, 16	radius 0.8 19×1×1, 16 16×1×1, 16	32×2048
	SA_pv2	radius 0.8 35×1×1, 32 32×1×1, 32	radius 1.2 35×1×1, 32 32×1×1, 32	64×2048
	SA_pv3	radius 1.2 67×1×1, 64 64×1×1, 64	radius 2.4 67×1×1, 64 64×1×1, 64	128×2048
	SA_pv4	radius 2.4 67×1×1, 64 64×1×1, 64	radius 4.8 67×1×1, 64 64×1×1, 64	128×2048
	SA_BEV	Bilinear Interpolation		256×2048
VSA Point Feature Fusion	Concat	[f^{raw}, f^{pv1}, f^{pv2}, f^{pv3}, f^{pv4}, f^{BEV}]		640×2048
	Linear Layer	640, 128		128×2048

for filters pseudo labels. However, the authors did not implement 3DIoUMatch in SECOND-IoU, we have implemented 3DIoUMatch on SECOND-IoU to analyze its effectiveness compared with UpCycling.

According to 3DIoUMatch, among the pseudo labels filtered according to the module in IoU, only terms that help improve box regression are selectively included in the loss. In the first attempt, the experiment † case in Table 3.7 is conducted, including both the box regression and cls loss value from the RPN module among the pseudo labels extracted from SECOND-IoU. The performance, however, is severely degraded compared with the baseline model's performance. Thus, as following implementation of the 3DIoUMatch concept, we select only the loss useful for box regression among RPN module loss terms. It could be confirmed through the results of 3DIoUMatch

Table 3.7: Partial-label scenario results with 2% of labeled data in the KITTI dataset. 3DIoUMatch † indicates the first attempt experiment of not applying selective supervision of box regression loss term.

AP_{3D}		2%		
		Easy	Mod	Hard
SECOND-IoU	Baseline	56.69	44.11	37.19
	3DIoUMatch †	29.12	23.03	20.33
	improved (%)	-48.64	-47.78	-45.32
	3DIoUMatch	63.57	49.58	43.00
	improved (%)	12.13	12.39	15.62

from Table 3.7 that the baseline model performance is well improved by the correct loss selection. Through this, we could judge that the implementation of SECOND-IoU based 3DIoUMatch is reasonable. For training, we follow the original 3DIoUMatch training settings, and more details on configurations are in the CODE supplementary.

3.6.4 Implementation of Inversion Attack

The research on inversion attacks that aim to restore original data from feature data has mainly focused on 2D images. Several studies, such as those referenced in [60, 87, 88], have proposed different inversion attack models based on convolutional neural networks (CNNs) and have shown improvements in performance by using prediction results and explanations. Additionally, an inversion attack model that utilizes a GAN generator and 1x1 convolution has been proposed in [90]. However, to the best of our knowledge, research on inversion attacks for 3D point clouds remains limited.

For this purpose, we employ the inversion attack model utilizing the decoder method [60], which is commonly used to assess the invertibility of a model composed of convolutional layers [87, 88].

We have developed the inversion attack models that reconstruct the raw point clouds from the intermediate features at the 1st, 3rd, and 5th convolution layers in 3D backbone

Table 3.8: 3D reconstructor network architecture from xconv_1.

Layers		**RECONSTRUCTOR Network**	Output size
INPUT		xconv_1	16×41×400×352
Conv3d Layers	conv_1	16×3×3×3, 16, padding 1,1,1	16×41×400×352
	conv_2	16×3×3×3, 16, padding 1,1,1	16×41×400×352
	conv_3	16×3×3×3, 16, padding 1,1,1	16×41×400×352
ConvTranspose3d Layers	upconv_1	16×3×3×3, 4, padding 1,1,1	4×41×400×352

Table 3.9: 3D reconstructor network architecture from xconv_3.

Layers		**RECONSTRUCTOR Network**	Output size
INPUT		xconv_3	64×11×100×88
Conv3d Layers	conv_1	64×3×3×3, 64, padding 1,1,1	64×11×100×88
	conv_2	64×3×3×3, 64, padding 1,1,1	64×11×100×88
	conv_3	64×3×3×3, 64, padding 1,1,1	64×11×100×88
ConvTranspose3d Layers	upconv_1	64×3×3×3, 32, stride 2,2,2, padding 1,1,1/0,1,1	32×21×200×176
	upconv_2	32×3×3×3, 16, stride 2,2,2, padding 1,1,1/0,1,1	16×41×400×352
	upconv_3	16×3×3×3, 4, padding 1,1,1	4×41×400×352

Table 3.10: 3D reconstructor network architecture from xconv_out.

Layers		**RECONSTRUCTOR Network**	Output size
INPUT		xconv_out	128×2×50×44
Conv3d Layers	conv_1	128×3×3×3, 128, padding 1,1,1	128×2×50×44
	conv_2	128×3×3×3, 128, padding 1,1,1	128×2×50×44
	conv_3	128×3×3×3, 128, padding 1,1,1	128×2×50×44
ConvTranspose3d Layers	upconv_1	128×5×3×3, 64, stride 2,1,1, padding 1,1,1	64×5×50×44
	upconv_2	64×5×3×3, 64, stride 2,2,2, padding 1,1,1/0,1,1	64×11×100×88
	upconv_3	64×3×3×3, 32, stride 2,2,2, padding 1,1,1/0,1,1	32×21×200×176
	upconv_4	32×3×3×3, 16, stride 2,2,2, padding 1,1,1/0,1,1	16×41×400×352
	upconv_5	16×3×3×3, 4, padding 1,1,1	4×41×400×352

network in Table 3.5, following the decoder method [60]. The inversion attack model structures for reconstructing features from the 1st, 3rd, and 5th layers are consistent with the structures presented in Tables 3.8, 3.9, and 3.10, respectively. The initial part of each inversion attack model consistently consists of three convolution layers. After that, the number of transposed convolution layers in the model corresponds to the count of layers that generate the input feature data.

To reconstruct the point clouds from input features for each dataset (KITTI, Waymo, and Lyft), we have developed independent inversion attack models for every dataset and followed the training settings in the decoder method [60]. More details on configurations are in the CODE supplementary.

3.7 Supplementary Evaluation

In this section, we provide additional supplementary experiment results that reinforce the arguments of this paper.

3.7.1 Feature-level Augmentation

Comparing point set features as with voxel-based features is not precise since raw-point augmentation impacts point sampling; feature augmentation is performed based on point samples different from those when raw-point augmentation is applied. Doing our best, however, we conducted additional experiments by comparing the closest point features in pairs. The RMSE results are 1.605@FLIP, 1.297@ROT, and 0.906@GT, confirming a similar trend as voxel-based.

3.7.2 Privacy Protection

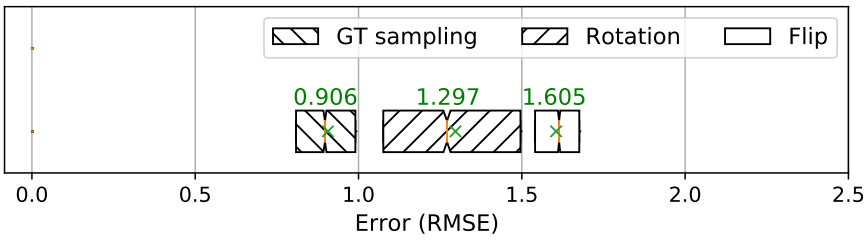
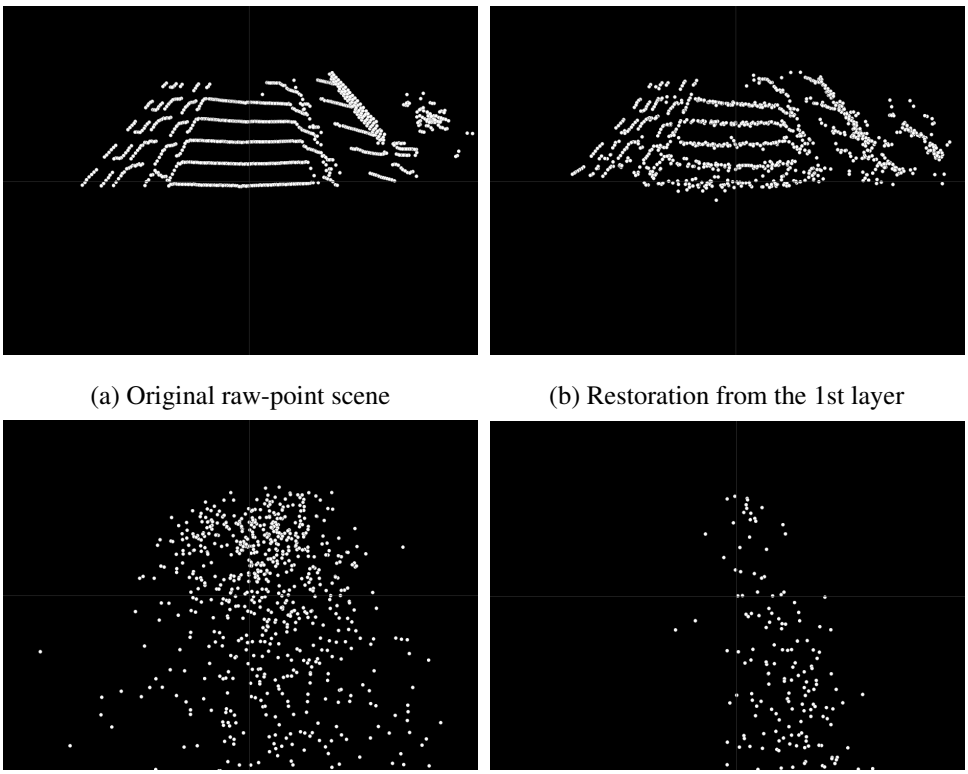


Figure 3.8: RMSE between raw- and set-type feature-level augmentations of the entire KITTI training dataset. Box range covers the first quartile to the third quartile and the mark '×' indicates the mean value.



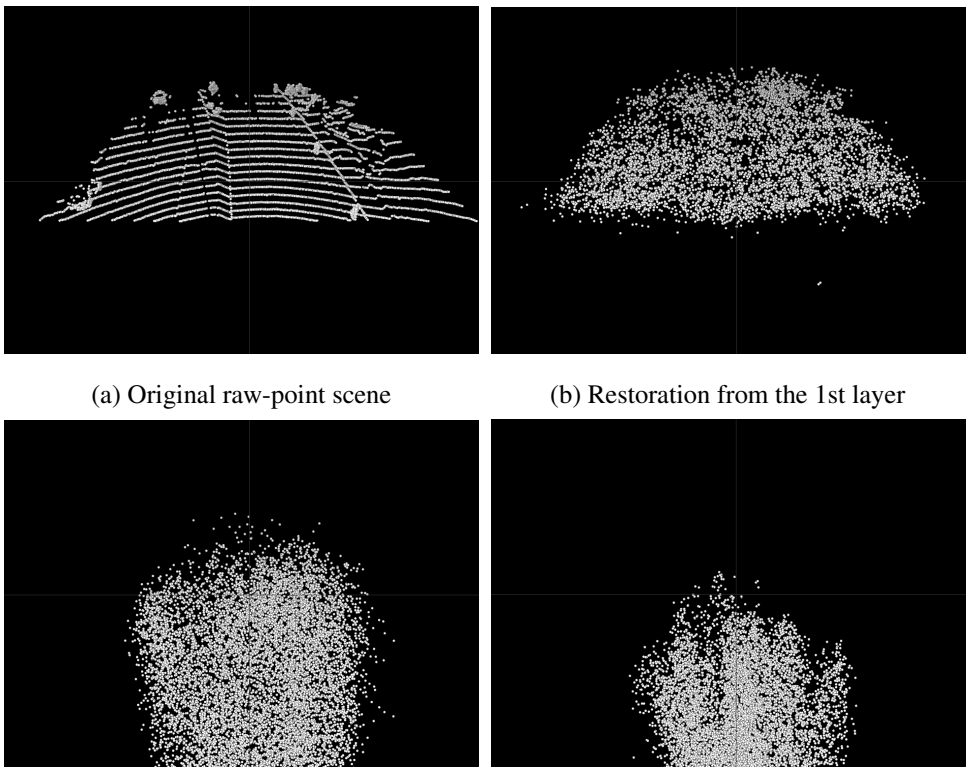
(a) Original raw-point scene

(b) Restoration from the 1st layer

(c) Restoration from the 3rd layer

(d) Restoration from the 5th layer, same as UpCycling

Figure 3.9: Results of inversion attack for the 3D backbone model of SECOND-IoU and PV-RCNN. The example 3D point cloud scene is in **KITTI**.



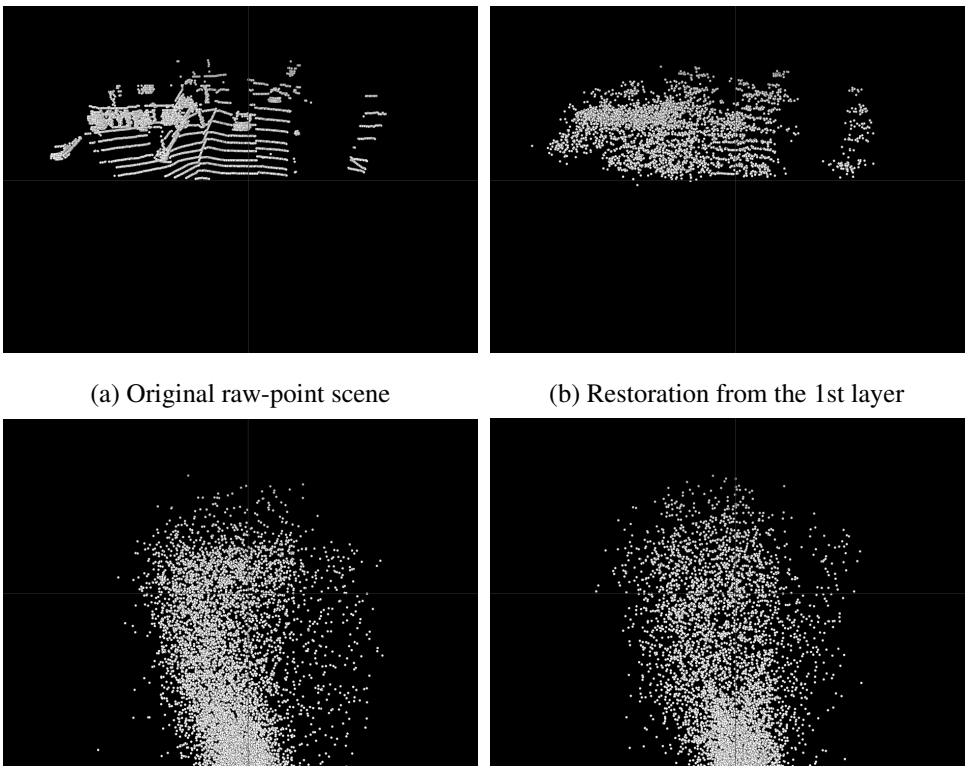
(a) Original raw-point scene

(b) Restoration from the 1st layer

(c) Restoration from the 3rd layer

(d) Restoration from the 5th layer, same as UpCycling

Figure 3.10: Results of inversion attack for the 3D backbone model of SECOND-IoU and PV-RCNN. The example 3D point cloud scene is in **Waymo**.



(a) Original raw-point scene

(b) Restoration from the 1st layer

(c) Restoration from the 3rd layer

(d) Restoration from the 5th layer, same as UpCycling

Figure 3.11: Results of inversion attack for the 3D backbone model of SECOND-IoU and PV-RCNN. The example 3D point cloud scene is in **Lyft**.

Feature data produced from the 3D object detection network.

Figures 3.12-3.13 shows the grid-type feature's activation heatmaps and set-types feature's positions corresponding to GTs at the raw-point data. In UpCycling, the state of the feature-level data after passing the 3D Backbone networks is very coarse. UpCycling uses these de-identified feature-level data for SSL of 3D object detection. In order to extract identifying information from this de-identified data, inversion attacks must be employed. We will discuss the attempts to reconstruct data via inversion attacks in the following section. Additionally, since a regular detection pipeline with the 3D scene naturally produces an unlabeled intermediate feature, UpCycling eliminates the need for extra AV-side computation (e.g., local training) or server-side annotation effort.

Restored point cloud scene using the inversion attack.

We perform an inversion attack on the 3D backbone network in SECOND-IoU and PV-RCNN. The 3D point cloud scenes in Figures 3.9-3.11(a) originate from the KITTI, Waymo, and Lyft datasets, respectively. Figures 3.9-3.11(b)-(d) present the restoration results for intermediate features at three different convolutional layers of the backbone network: 1st, 3rd, and 5th (last) layers, respectively. Although the point cloud restored from the first layer is relatively similar to the original scene, it becomes considerably different when applied to features from deeper layers in all cases. We confirm that intermediate feature data generated from the deepest layer utilized by UpCycling in all datasets, including KITTI, Waymo, and Lyft, makes it impossible to accurately reconstruct the original scene.

3.7.3 Effect of Feature Augmentation Schemes in Domain Adaptation

In Section 5.2, we investigate feature augmentation by evaluating the superiority of *F-GT*, which is utilized for UpCycling, to other augmentation schemes (*e.g.,* Flip, Noise, RS, and Rotation) in a partial-label scenario. Further, we investigate the superiority of

Table 3.11: Effects of feature augmentation schemes in the domain adaptation scenario with the same settings as Sections 5.2 and 5.4

Dataset	Method	SECOND-IoU (Closed Gap[%])	
		AP_{BEV}	AP_{3D}
KITTI	Baseline	54.14 (0.00)	10.16 (0.00)
	Flip(w/ SN)	76.68 (62.23)	48.73 (53.67)
	Noise(w/ SN)	81.46 (75.43)	51.21 (57.12)
	RS(w/ SN)	78.59 (67.50)	46.52 (50.61)
	Rotation(w/ SN)	77.98 (65.83)	44.25 (47.44)
	UpCycling (w/ SN)	**84.12 (82.77)**	**67.65 (80.00)**
	Oracle	90.36 (100.0)	82.02 (100.0)

UpCycling to other feature augmentation schemes in the domain adaptation scenario with the same settings as Sections 5.2 and 5.4.

In this experiment, **Baseline** evaluates the baseline model pre-trained with Waymo dataset directly in target domain (KITTI) and **Oracle** adapts the model with fully supervised learning in the target domain, which provide the lower- and upper-bound performance, respectively. For feature-level augmentations, we adopts Flip, Noise, RS, and Rotation described in Section 5.2. We utilize SECOND-IoU and adopt **SN** option for adaptation to KITTI domain since object sizes in KITTI are different from those in Waymo.

Table 3.11 performs the same comparison in the domain adaptation scenario described in Section 5.4, showing each scheme's $\mathrm{AP}_{\mathrm{BEV}}$, $\mathrm{AP}_{\mathrm{3D}}$ performances and its relative position between Baseline (0) and Oracle (100). The results show that our UpCycling provides *the best performance* in all cases.

3.7.4 Other Class Detection Results

We report per-class average precision on other classes of the KITTI dataset in Table 3.12. We use the same settings as in Sections 5.5. The experiment using a 10% partial-label

Table 3.12: The AP results for Car and Pedestrian classes in a partial-label scenario, utilizing 10% labeled data from the KITTI dataset.

$AP_{3D}(10\%)$		Car (@0.7 IoU)			Pedestrian (@0.5 IoU)		
		Easy	Mod	Hard	Easy	Mod	Hard
SECOND-IoU	Baseline	75.77	58.75	52.27	15.40	13.10	12.27
	UpCycling	**76.01**	**61.09**	**54.34**	**18.08**	**15.13**	**14.49**
	Improved (%)	0.32	3.98	3.96	17.40	15.50	18.10
PV-RCNN	Baseline	80.98	66.80	59.60	14.81	13.39	12.42
	UpCycling	**83.82**	**69.52**	**62.47**	**16.10**	**15.18**	**15.00**
	Improved (%)	3.51	4.07	4.82	8.71	13.37	20.77
$AP_{BEV}(10\%)$		Car (@0.7 IoU)			Pedestrian (@0.5 IoU)		
		Easy	Mod	Hard	Easy	Mod	Hard
SECOND-IoU	Baseline	82.59	73.63	65.80	22.02	18.30	17.48
	UpCycling	**86.81**	**75.87**	**67.28**	**23.59**	**19.67**	**18.95**
	Improved (%)	5.11	3.04	2.25	7.13	7.49	8.41
PV-RCNN	Baseline	89.22	80.95	73.30	16.87	15.26	15.01
	UpCycling	**91.33**	**83.25**	**75.85**	**20.03**	**18.27**	**18.10**
	Improved (%)	2.36	2.84	3.48	18.73	19.72	20.59

scenario on KITTI training data is essential to understand UpCycling's effectiveness to the Pedestrian class as well as the Car class. As shown in Table 3.12, UpCycling achieves to improve the detection accuracy in other classes significantly, regardless of the class, model, and task difficulty.

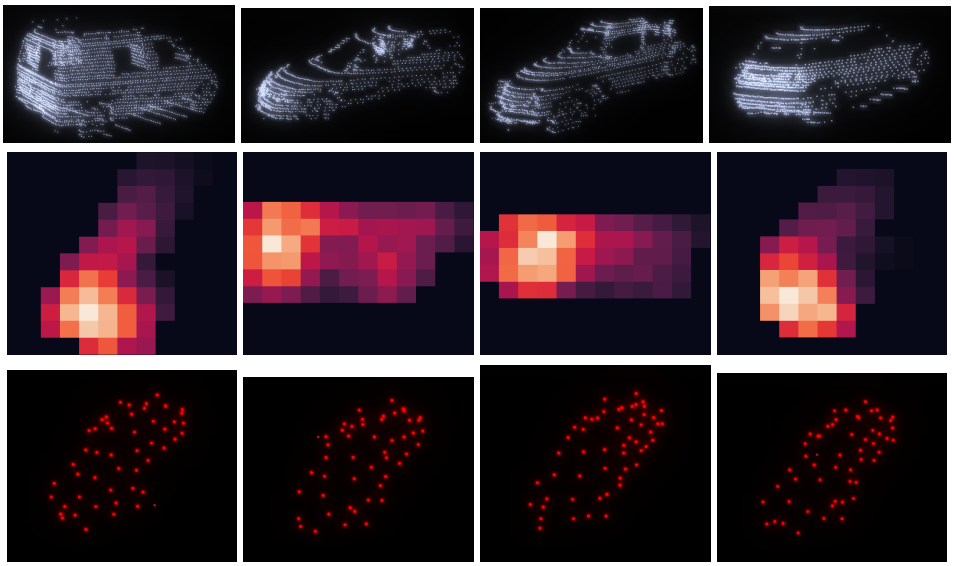


Figure 3.12: Figures for the Car class in KITTI dataset. GT point clouds and corresponding grid-type feature's activation heatmaps and set-type feature's positions.

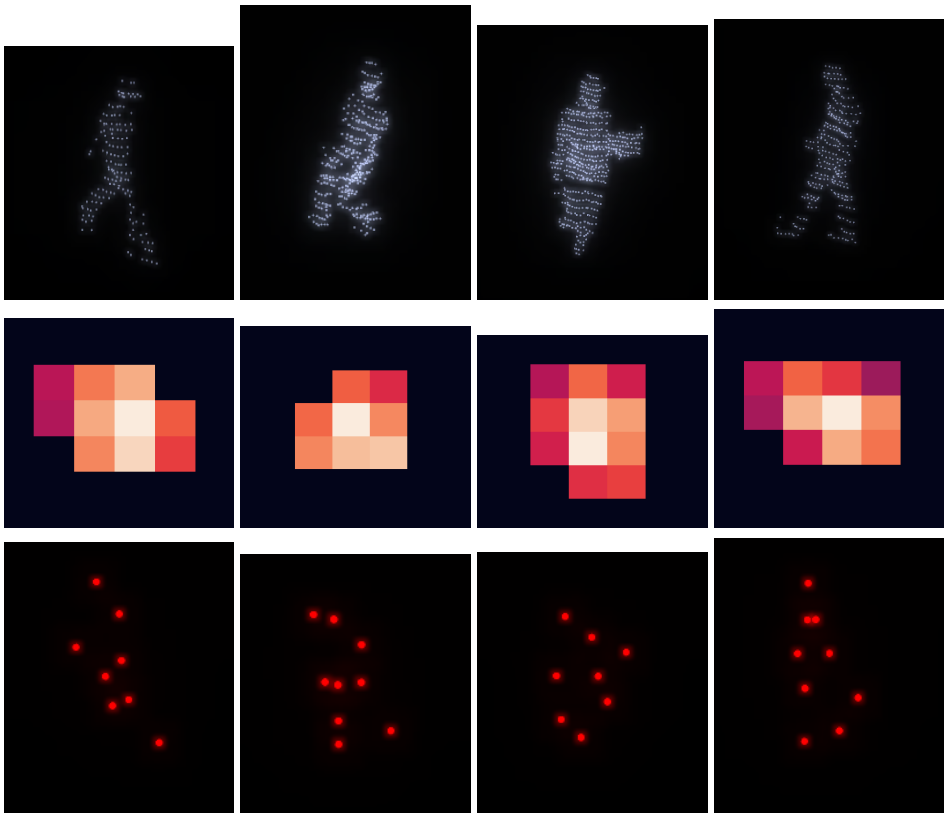


Figure 3.13: Figures for the Pedestrian class in KITTI dataset. GT point clouds and corresponding grid-type feature's activation heatmaps and set-type feature's positions.

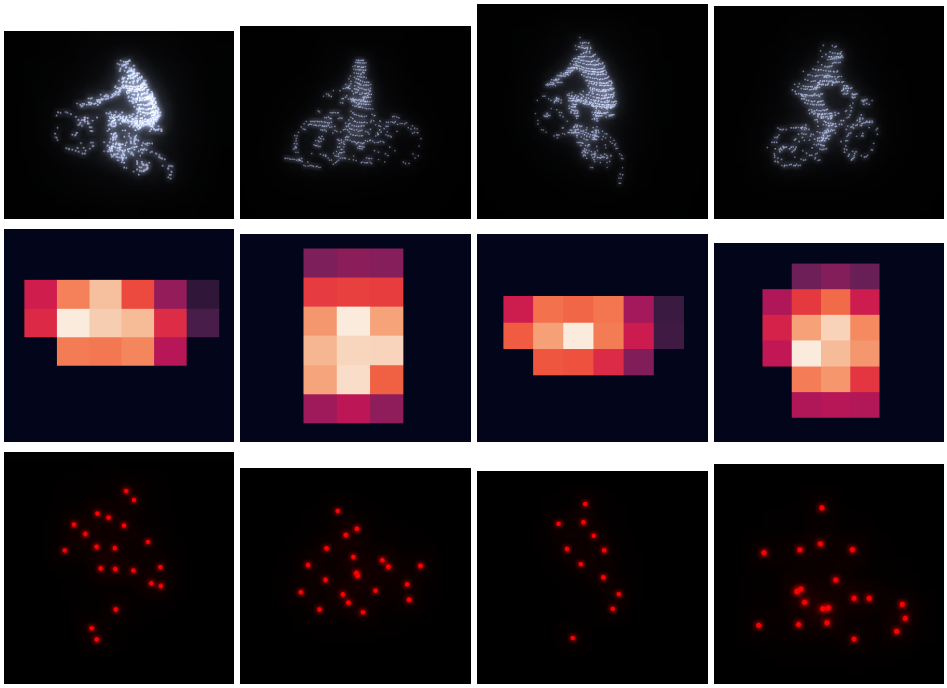


Figure 3.14: Figures for the Cyclist class in KITTI dataset. GT point clouds and corresponding grid-type feature's activation heatmaps and set-type feature's positions.

Chapter 4

Conclusion

4.1 Research Contributions

In this dissertation, we dealt with autonomous vehicles' object detection performance enhancement techniques using communication systems.

In Chapter 2, we have presented Beyond-Vision, a standard-compliant relay system in C-V2V, that aims to guarantee stable MRR in vehicular communications. To ensure effective relaying performance, each V-UE should be able to distinguish CAMs that are not likely to be received at nearby V-UEs. Beyond-Vision enables V-UEs to examine eMRR of received CAMs with no overhead by utilizing previously unused bytes in the conventional CAM. By doing so, Beyond-Vision relays CAMs more efficiently than the other comparison schemes. Based on our realistic simulation results, we have verified the performance of Beyond-Vision in various environments, demonstrating that Beyond-Vision significantly improves MRR performance compared with the other comparison schemes and that its relaying transmission is very effective.

In Chapter 3, we have presented UpCycling, a novel semi-supervised learning framework for 3D object detection models that does not utilize unlabeled raw-level 3D scenes but only de-identified intermediate features. To the best of our knowledge, UpCycling is the first framework that tackles labeling cost, privacy leakage, and AV-

side computation burden altogether. Our deep investigation of feature-based learning reveals that combining *hybrid pseudo label*, *F-GT*, and *F-RoT* significantly improves pseudo-label quality and data diversity. Results from various experiments demonstrate that UpCycling achieves SOTA accuracy with large margins in both partial-label and domain adaptation scenarios, regardless of the model, dataset, and task (difficulty setting or average precision of BEV/3D view). With the superior performance, UpCycling discloses the value of unlabeled feature-based learning in the context of 3D object detection, in terms of both privacy and accuracy.

4.2 Future Research Directions

Based on the results of this dissertation, there are new future research directions which require further investigation. We highlight some of them as follows.

First, we plan to investigate the relaying message traffic in various V-UE density environments, and to propose a novel adaptive relaying-mode for high density environments. Second, regarding the machine learning system design for a real application, we plan to propose a novel system for UpCycling considering the communication costs, and the enhancement of privacy leakage.

Bibliography

[1] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[2] 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 (CVPR)*, 2012.

[3] J. Houston, G. Zuidhof, L. Bergamini, Y. Ye, A. Jain, S. Omari, V. Iglovikov, and P. Ondruska, "One thousand and one hours: Self-driving motion prediction dataset," *CoRR*, vol. abs/2006.14480, 2020. [Online]. Available: https://arxiv.org/abs/2006.14480

[4] Y. Wang, X. Chen, Y. You, L. E. Li, B. Hariharan, M. Campbell, K. Q. Weinberger, and W.-L. Chao, "Train in germany, test in the usa: Making 3d object detectors generalize," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 713–11 723.

[5] J. Yang, S. Shi, Z. Wang, H. Li, and X. Qi, "St3d: Self-training for unsupervised domain adaptation on 3d object detection," in *Proceedings of the IEEE/CVF Con-*

ference on Computer Vision and Pattern Recognition, 2021, pp. 10 368–10 378.

[6] Simulation of urban mobility. [Online]. Available: http://sumo.dlr.de

[7] K. Schofield, M. L. Larson, and K. J. Vadas, "Vehicular vision system," May 10 2005, US Patent 6,891,563.

[8] 3GPP TS 36.213, "Technical specification group radio access network; evolved universal terrestrial radio access (E-UTRA); physical layer procedures ," ver. 15.5.0, Mar. 2019.

[9] 3GPP TS 36.321, "Technical specification group radio access network; evolved universal terrestrial radio access (E-UTRA); medium access control (MAC) protocol specification," ver. 15.5.0, Mar. 2019.

[10] 3GPP TR 36.885, "Technical Specification Group Radio Access Network; Study on LTE-based V2X Services," ver. 14.0.0, June 2016.

[11] ETSI TS 302 637-2, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service," V1.3.2, Nov. 2011.

[12] IEEE 802.11p, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Amendment 6: Wireless Access in Vehicular Environments," July 2010.

[13] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proc. IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011.

[14] F. J. Martinez *et al.*, "A Street Broadcast Reduction scheme (SBR) to mitigate the broadcast storm problem in vanets," *Wireless personal Commun.*, vol. 56, no. 3, pp. 559–572, 2011.

[15] G. Korkmaz *et al.*, "Urban multi-hop broadcast protocol for inter-vehicle communication systems," in *Proc. ACM VANET*, 2004.

[16] M. Li, K. Zeng, and W. Lou, "Opportunistic broadcast of event-driven warning messages in vehicular ad hoc networks with lossy links," *Comput. Netw.*, vol. 55, no. 10, pp. 2443–2464, 2011.

[17] N. Wisitpongphan *et al.*, "Broadcast storm mitigation techniques in vehicular ad hoc networks ," *IEEE Wireless Commun.*, vol. 14, no. 6, pp. 84–94, 2007.

[18] A. Wegener *et al.*, "AutoCast: An adaptive data dissemination protocol for traffic information systems," in *Proc. IEEE VTC*, 2007, pp. 1947–1951.

[19] Q. Xiang *et al.*, "Data preference matters: A new perspective of safety data dissemination in vehicular ad hoc networks," in *Proc. IEEE INFOCOM*, Apr. 2015, pp. 1149–1157.

[20] Kim, Taehyung and Park, Yosub and Kim, Hyunsoo and Hong, Daesik, "Cooperative superposed transmission in Cellular-based V2V systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 11 888–11 901, 2019.

[21] M. M. Taha and Y. M. Hasan, "Vanet-dsrc protocol for reliable broadcasting of life safety messages," in *Proc. IEEE ISSPIT*, Dec. 2007, pp. 104–109.

[22] A. Bazzi, B. M. Masini, A. Zanella, and I. Thibault, "On the performance of IEEE 802.11p and LTE-V2V for the cooperative awareness of connected vehicles," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10 419–10 432, 2017.

[23] B. Kim, S. Kim, H. Yoon, S. Hwang, M. X. Punithan, B. R. Jo, and S. Choi, "Nearest-first: Efficient relaying scheme in heterogeneous v2v communication environments," *IEEE Access*, vol. 7, pp. 23 615–23 627, 2019.

[24] S. Park, B. Kim, H. Yoon, and S. Choi, "RA-eV2V: relaying systems for LTE-V2V communications," *J. Commun. Netw.*, vol. 20, no. 4, pp. 396–405, 2018.

[25] J. Heo, B. Kang, J. M. Yang, J. Paek, and S. Bahk, "Performance-cost tradeoff of using mobile roadside units for V2X communication," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 9049–9059, 2019.

[26] B. Kang, S. Choi, S. Jung, and S. Bahk, "D2D Communications Underlaying Cellular Networks on Licensed and Unlicensed Bands with QoS constraints," *J. Commun. Netw.*, vol. 21, no. 4, pp. 416–428, 2019.

[27] B. Kang, S. Jung, and S. Bahk, "Sensing-based power adaptation for cellular V2X mode 4," in *Proc. IEEE DySPAN*, 2018.

[28] ETSI TS 302 637-3, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specification of Decentralized Environmental Notification Basic Service," V1.3.2, Nov. 2014.

[29] R. Molina-Masegosa and J. Gozalvez, "LTE-V for Sidelink 5G V2X Vehicular Communications: A New 5G Technology for Short-Range Vehicle-to-Everything Communications," *IEEE Veh. Technol. Mag.*, no. 4, pp. 30–39, Dec. 2017.

[30] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro, "Lte for vehicular networking: A survey," *IEEE Commun. Mag.*, no. 5, pp. 148–157, May 2013.

[31] 5G Americas white paper, "V2X cellular solutions," Oct. 2016.

[32] 3GPP TR 36.885, "Technical Specification Group Radio Access Network; Study on LTE-based V2X Services," V2.0.0, June 2016.

[33] M. Rupp, S. Schwarz, and M. Taranetz, *The Vienna LTE-Advanced Simulators: Up and Downlink, Link and System Level Simulation*, 1st ed., ser. Signals and Communication Technology. Springer Singapore, 2016.

[34] 3GPP TS 36.101, "UE radio transmission and reception," ver. 15.1.0, Jan. 2018.

[35] OpenStreetMap. [Online]. Available: https://www.openstreetmap.org/

[36] Draft new Report ITU-R M.[IMT.EVAL], "Guidelines for Evaluation of Radio Interface Technologies for IMT-Advanced," 2008.

[37] WINNER+ Deliverables, "WINNER+ Final Channel Models," D5.3, June 2010.

[38] 3GPP TR 36.843, "Technical Specification Group Radio Access Network; Study on LTE Device to Device Proximity Services; Radio Aspects," V12.0.1, Mar. 2014.

[39] The network simulator-3. [Online]. Available: https://www.nsnam.org/

[40] D. Jiang, Q. Chen, and L. Delgrossi, "Optimal data rate selection for vehicle safety communications," in *Proc. ACM VANET*, 2008, pp. 30–38.

[41] N. Zhao, T.-S. Chua, and G. H. Lee, "Sess: Self-ensembling semi-supervised 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 079–11 087.

[42] H. Wang, Y. Cong, O. Litany, Y. Gao, and L. J. Guibas, "3dioumatch: Leveraging iou prediction for semi-supervised 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 615–14 624.

[43] J. Xiong, R. Bi, M. Zhao, J. Guo, and Q. Yang, "Edge-assisted privacy-preserving raw data sharing framework for connected autonomous vehicles," *IEEE Wireless Communications*, vol. 27, no. 3, pp. 24–30, 2020.

[44] D. Eckhoff and C. Sommer, "Driving for big data? privacy concerns in vehicular networking," *Security & Privacy, IEEE*, vol. 12, pp. 77–79, 01 2014.

[45] Y. Ming and X. Yu, "Efficient privacy-preserving data sharing for fog-assisted vehicular sensor networks," *Sensors*, vol. 20, no. 2, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/2/514

[46] T. Mulder and N. E. Vellinga, "Exploring data protection challenges of automated driving," *Computer Law & Security Review*, vol. 40, p. 105530, 2021.

[47] F. Pittaluga, S. J. Koppal, S. B. Kang, and S. N. Sinha, "Revealing scenes by inverting structure from motion reconstructions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 145–154.

[48] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016.

[49] O. Gupta and R. Raskar, "Distributed learning of deep neural network over multiple agents," *Journal of Network and Computer Applications*, vol. 116, pp. 1–8, 2018.

[50] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, "Split learning for health: Distributed deep learning without sharing raw patient data," *arXiv preprint arXiv:1812.00564*, 2018.

[51] A. Singh, P. Vepakomma, O. Gupta, and R. Raskar, "Detailed comparison of communication efficiency of split learning and federated learning," *arXiv preprint arXiv:1909.09145*, 2019.

[52] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization," in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, S. Chiappa and R. Calandra, Eds., vol. 108. PMLR, 26–28 Aug 2020, pp. 2021–2031. [Online]. Available: http://proceedings.mlr.press/v108/reisizadeh20a.html

[53] N. Shlezinger, M. Chen, Y. C. Eldar, H. V. Poor, and S. Cui, "Uveqfed: Universal vector quantization for federated learning," *IEEE Transactions on Signal Processing*, vol. 69, pp. 500–514, 2021.

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

[55] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, Oct 2018. [Online]. Available: http://dx.doi.org/10.3390/s18103337

[56] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019, pp. 12 689–12 697.

[57] J. Deng, S. Shi, P. Li, W. Zhou, Y. Zhang, and H. Li, "Voxel r-cnn: Towards high performance voxel-based 3d object detection," in *AAAI*, 2021.

[58] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[59] P. Vepakomma, T. Swedish, R. Raskar, O. Gupta, and A. Dubey, "No peek: A survey of private distributed deep learning," *arXiv preprint arXiv:1812.03288*, 2018.

[60] A. Dosovitskiy and T. Brox, "Inverting visual representations with convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[61] J. Jeong, S. Lee, J. Kim, and N. Kwak, "Consistency-based semi-supervised learning for object detection," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer,

F. d 'Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 10 759–10 768. [Online]. Available: http://papers.nips.cc/paper/9259-consistency-based-semi-supervised-learning-for-object-detection.pdf

[62] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," 2016.

[63] O. D. Team, "Openpcdet: An open-source toolbox for 3d object detection from point clouds," https://github.com/open-mmlab/OpenPCDet, 2020.

[64] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," 2017.

[65] K. Sohn, D. Berthelot, C.-L. Li, Z. Zhang, N. Carlini, E. D. Cubuk, A. Kurakin, H. Zhang, and C. Raffel, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *arXiv preprint arXiv:2001.07685*, 2020.

[66] T. Miyato, S. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1979–1993, 2019.

[67] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 702–703.

[68] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[69] D.-H. Lee, "Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks," *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*, 07 2013.

[70] P. Li, D. Li, W. Li, S. Gong, Y. Fu, and T. M. Hospedales, "A simple feature augmentation for domain generalization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 8886–8895.

[71] P. Chu, X. Bian, S. Liu, and H. Ling, "Feature space augmentation for long-tailed data," in *European Conference on Computer Vision*. Springer, 2020, pp. 694–710.

[72] F. Cen, X. Zhao, W. Li, and G. Wang, "Deep feature augmentation for occluded image classification," *Pattern Recognition*, vol. 111, p. 107737, 2021.

[73] B. Liu, X. Wang, M. Dixit, R. Kwitt, and N. Vasconcelos, "Feature space transfer for data augmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[74] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 6438–6447. [Online]. Available: https://proceedings.mlr.press/v97/verma19a.html

[75] V. Kumar, H. Glaude, C. de Lichy, and W. Campbell, "A closer look at feature space data augmentation for few-shot intent classification," *arXiv preprint arXiv:1910.04176*, 2019.

[76] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. Fort Lauderdale, FL, USA: PMLR, 20–22 Apr 2017, pp. 1273–1282. [Online]. Available: http://proceedings.mlr.press/v54/mcmahan17a.html

[77] S. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečnỳ, S. Kumar, and H. B. McMahan, "Adaptive federated optimization," *arXiv preprint arXiv:2003.00295*, 2020.

[78] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, and Y. Khazaeni, "Federated learning with matched averaging," *arXiv preprint arXiv:2002.06440*, 2020.

[79] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[80] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[81] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[82] M. Hahner, D. Dai, A. Liniger, and L. Van Gool, "Quantifying data augmentation for lidar based 3d object detection," *arXiv preprint arXiv:2004.01643*, 2020.

[83] bostondiditeam, "Exploratory findings for the kitti vision benchmark suite," https://github.com/bostondiditeam, 2017.

[84] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5188–5196.

[85] A. Dosovitskiy and T. Brox, "Inverting visual representations with convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4829–4837.

[86] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9446–9454.

[87] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/file/371bce7dc83817b7893bcdeed13799b5-Paper.pdf

[88] X. Zhao, W. Zhang, X. Xiao, and B. Lim, "Exploiting explanations for model inversion attacks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 682–692.

[89] B. Graham and L. van der Maaten, "Submanifold sparse convolutional networks," *arXiv preprint arXiv:1706.01307*, 2017.

[90] R. Herdt, M. Schmidt, D. O. Baguer, J. L. Arrastia, and P. Maass, "Model stitching and visualization how gan generators can invert networks in real-time," in *arXiv*, 2023. [Online]. Available: https://arxiv.org/abs/2302.02181

초 록

자율주행기술은 물체 감지 기법에 따라 크게 두 가지 방식이 존재한다. 하나는 차량이 V2X 통신을 통해 차량 정보를 공유하여 주변 차량의 위치와 도로 상황을 이해하는 협력하는 협력적 자율주행 방식이고, 다른 하나는 비전센서에서 얻은 정보를 딥러닝 모델로 처리해 물체의 종류와 차량과 물체 사이의 거리를 알아내는 독립적 자율주행 방식이다.

위의 두 가지 방법은 각각 다음과 같은 장단점이 있다. V2X 통신을 활용한 협력적 자율주행 방식은 비전 센서로 보이지 않는 영역에서 차량을 감지할 수 있는 장점이 있다. 그러나 모든 차량이 통신 인프라를 통한 정보 공유에 협력해야 한다는 제한 조건이 있으며, 각 차량이 보내는 상태 정보에 따라 신뢰성 문제가 발생한다. 반면, 비전 센서를 통한 독립적 자율주행 방식은 감지 신뢰도는 높지만 장애물에 가려진 영역은 감지할 수 없다. 따라서 자율주행차량의 사용자 안전을 보장하기 위해서는 이 두 가지 물체 감지 방식의 장점을 강화하기 위한 연구가 필요하다.

본 논문에서는 협력적 자율주행과 독립적 자율주행을 위한 물체 감지 성능을 향상시키기 위한 방법을 각각 제안한다.

첫째, 우리는 보조적인 피드백 과정이 없는 C-V2V 주문형 중계 시스템을 제안한다. 이 목표를 달성하기 위해 기존 CAM (Cooperative Awareness Message)에서 이전에 사용되지 않은 저장 공간을 활용하여 통신 범위 내에서 감지한 주변 차량ID (Adjacent Vehicle IDs)를 포함하는 새로운 CAM 구성을 도입한다. 이 기법을 통해서 차량 간 메시지는 NLOS (Non-Line-of-Sight) 환경이나, resource collision, channel 변동 등의 문제로 수신률이 저하가 초래되는 환경에서 중계 지원차량을 효과적으로 선정하여 차량간 통신이 이루어지게 한다. 우리는 제안 기법이 기존의 통신 표준

성능과 다른 비교 중계 기법들에 비해 향상된 메시지 전송 성공 비율을 보이는 것을 확인하였다.

둘째, 비식별화된 중간 데이터(feature data)를 활용하여 3D 물체 감지 모델 성능을 향상시키는 새로운 방식을 제안한다. 이 방식은 자율주행 차량이 운행 중에 물체를 감지하는 과정에서 발생하는 feature data를 활용함으로써, 자율주행 차량 측에 추가적인 계산 부담 없이 라벨링 비용과 개인 정보 유출 문제를 동시에 해결할 수 있다. 더욱이, 개인 정보를 보호하면서도, 이 방식은 도메인 적응 및 부분 레이블 시나리오에서 원시 수준의 레이블되지 않은 데이터를 활용하는 최신 방법들에 비해 더욱 우수하거나 동등한 성능을 보인다. 이러한 탁월한 성능을 바탕으로, 우리가 제안하는 이 방식은 3D 객체 감지의 맥락에서 개인정보 보호와 정확도를 동시에 고려하는 레이블되지 않은 중간 데이터 기반 학습의 가치를 입증한다.

본 학위논문에서는 통신시스템과 비전 기술을 모두 사용하여 자율주행의 물체 감지 성능을 향상시키는 다양한 기법들을 제안한다. 우리는 앞서 간단히 소개한 연구들을 통해 자율주행 사용자의 안전을 더욱 확보하여 자율주행차 상용화에 한 걸음 더 다가서고자 한다.

주요어: 자율주행, 차량간 통신, 준지도 학습, 3D 객체감지, 물체감지

학번: 2016-20989