Enhancing Perception Systems for Autonomous Vehicles Using Radar-Generated Occupancy Grid Maps

Alireza Najafi

Thesis submitted to the University of Ottawa
in partial fulfillment of the thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Declaration of Authorship

I hereby certify that this thesis is entirely my own original work except where otherwise indicated. I am aware of the University's regulations concerning plagiarism, including those concerning consequent disciplinary actions. Any use of the works of any other author, in any form, is properly acknowledged at their point of use.

Abstract

The continuous development in the field of Intelligent Transportation Systems (ITS) and Autonomous Vehicles (AV) promises to revolutionize transportation and society by enhancing road safety, traffic management, and environmental sustainability. Central to the realization of AVs are Perception systems that enable the vehicle to interact with and understand the surroundings. These systems utilize various sensors for measuring and recording data which are then processed by complex algorithms to derive meaningful information. With the emergence of Deep Learning (DL) and comprehensive driving scenario datasets, state-of-the-art methods have significantly matured, especially in the object detection field. These models mostly rely on cameras and LiDAR sensors for perception purposes. Despite their superior performance under favorable stations, these models often degrade in performance under real-world scenarios such as inadequate illumination or different weather conditions. These challenges hinder the reliability of AVs and raise concerns for safe navigation.

To tackle these challenges, various solutions have been proposed by the community of computer vision, however, camera and LiDAR sensors have inherent limitations against environmental challenges, in particular, which prevent them from performing reliably. On the other hand, Radar sensors have been gaining more popularity and recent studies have shown their potential in tackling these challenges. Despite their advantages, these sensors are still under-explored compared to the cameras and LiDARs, and the inadequate number of available Radar-inclusive datasets and sparse Radar recordings make it more difficult to develop Radar-aided perception models.

In our research, we propose a unified framework that aims to enhance the quality of Radar data frames by transforming the sparse point clouds into high-quality and high-resolution Occupancy Grid Maps (OGM). Our system utilizes Bayesian filtering and a Radar-specific sensor inverse modeling to adaptively estimate the occupancy state of different locations, represented by cells, across the mapping environment. The final product is the construction of a rich representation of the environment preserving structural details, compared to the indistinguishable structure of Radar point clouds. Our system reduces the extreme sparsity in Radar point clouds by more than 60% and eliminates the uncertainty in the whereabouts of obstacles by more than 25%. Furthermore, the experiments show the effectiveness of the OGMs in robustifying the performance of a baseline object detection model against foggy weather conditions.

By addressing the quality-related issue for the Radar portion of currently available datasets, we served to facilitate the study of the impact and benefits of this sensing modality. Utilizing the unique characteristics of these sensors and combining them with cameras and LiDARs through a more complex fusion algorithm can lead to a more robust and reliable perception system, taking us one step closer to the realization of Avs.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my parents for their unwavering love, support, and belief in me. Your sacrifices and encouragement have been the foundation of everything I have accomplished, and I am truly grateful for everything you have done.

I am also very thankful to my supervisor, Prof. Azzedine Boukerche. Your guidance, support, and valuable feedback have been essential to my growth and success throughout this journey. I truly appreciate the opportunity to work with you and the trust you have shown in me. I would also like to extend my thanks to my examination committee for their valuable time, thoughtful comments, and insightful suggestions, which greatly contributed to the quality of this work.

Last but not least, I want to thank my dear friends for always being there for me, for the good times and the tough ones, for when I made breakthroughs and the times I was stuck and spiraling around. Your support has meant so much to me. A special thank you to Mozhgan for her constant and thoughtful support. It was with the support of all of you that my Master's thesis became possible.

Dedication

To my dear family, especially my beloved mother and my dear life-long friend, who never ceased to support me during the ups and downs and the turns and twists of my master's journey.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

For centuries, humans developed and used different modes of transportation for many of the essential activities in their lives. Historically, transportation has been the enabler for building communications, making trades, relocating, and establishing civilizations. With the development of vehicles, steamboats, and railroads in the 1800s, transportation experienced a revolution becoming more crucial for the prosperity of nations. For millenniums and centuries, the existence and availability of tools that could provide autonomous transportation used to be among the far-fetched dreams of mankind. However, with the advancements in technology in the modern world, it is now only a matter of time before this dream comes true. Intelligent Transportation Systems (ITS) and Autonomous Vehicles (AV) will revolutionize the life of every human being and the cycle of every industry because transportation is and has been one the most vital pillars of our lives [38, 43, 46, 47, 52, 54, 59, 62, 65, 73, 75, 264, 314, 352].

The impact of the deployment of AVs is expected to range from road safety and traffic management to environmental benefits. Most road incidents occur because of human error such as carelessness, inattention, and impaired driving. According to the 2023 statistics provided by the government of Canada [229], in the year 2022, around 118,853 people were injured in various scenarios driving scenarios which led to about 2000 fatalities. Distraction and impaired driving were among the reasons, contributing by 20% and 23%, respectively. The use of AVs on the road can expel human error contribution factors, and significantly increase road safety.

Furthermore, the AV and ITS propose potential in traffic management, especially, in populated urban areas. As tailored by Forbes [124], as the ITS infrastructure enables the AVs to communicate with each other, they can optimize their speed, timing, and choice of route to maintain a steady traffic flow, reduce congestion, and cut commute times. The usage of AV provides accessibility for all people, especially for the ones who are not able to drive such as the elderly and the young.

Global warming, wildfires, and severe changes in climates across the world have attracted much attention and concern towards the carbon footprint of various activities. Accordingly, the environmental benefits offered by AVs can be potentially among the

solutions to address these challenges. AVs have no tailpipe emissions and over their lifespan leave less than half of the carbon footprint of regular vehicles [124]. The United States Environmental Protection Agency (EPA) addresses the myths about EVs in an article [6], highlighting that considering the manufacturing and recharging required for EVs, their carbon footprint is still lower than regular gasoline vehicles during their lifetime. Conclusively, the reduction in engine roaring and greenhouse pollution, will lead our metropolitan areas to have better air quality and healthier living standards.

An AV is realized when a vehicle can perceive and comprehend the events in its surroundings, and based on this understanding perform reliable decision-making and respond appropriately to various driving scenarios for safe navigation to the destination. Initially and at the heart of these vehicles, lies their perception system which is primarily responsible for acquiring the mentioned comprehensive understanding of the vehicle's environment. Perception systems in AVs consist of various sensors, including cameras, LiDAR, and Radar sensors which capture data from the surroundings. However, these sensors' representation of the world is raw and cannot be used directly for decision-making or navigation. This raw data undergoes complex processing and analysis, utilizing advanced algorithms and techniques to extract high-level information. Tasks such as object and obstacle detection, lane tracking, and pedestrian recognition are essentially performed by the perception systems.

In recent years, significant advancements have been made in perception-related technologies [144], fueled by the rapid development of artificial intelligence and powerful hardware. Deep Learning (DL) algorithms, in particular, have revolutionized perception systems, especially in the object detection field where the goal is to identify and localize the instances of various objects across a driving scenario. These advancements have allowed AVs to get closer than ever to human-like perception and interpretation of their environment, exhibiting high accuracy and inference speed.

1.1 Problem Statement

Despite the advancements in perception-related technologies, several challenges still remain in achieving reliable and robust perception systems. Firstly, a perception system should be robust to varying situations and environmental factors, such as adverse weather conditions that significantly reduce visibility, to ensure reliable and safe navigation. Secondly, selecting a practical and cost-efficient sensor suite is crucial for providing adequate and accurate information for the perception systems. Given that each sensor has its own limitations, a suitable configuration should be designed where the combination of sensors compensates for each other's shortcomings. Furthermore, sensor fusion and data integration add complexity, requiring algorithms to combine recordings of different sensors to achieve an accurate and unified representation of the driving scenario. The computational load of processing the large volume of information needed for real-time decision-making presents a challenge that requires powerful hardware and efficient algorithms.

Among these challenges, adverse weather conditions remain particularly problematic. These conditions leave a degrading impact on perception sensors by imposing noise, attenuation of measurement signals, and reduction of visibility. This leads to an unreliable and inconsistent understanding of the surrounding events. Thus, it makes it a critical focus area for ongoing research and development.

More specifically, cameras and LiDARs were the most commonly used sensors for the development of perception systems, traditionally [144, 220]. These sensors experience significant degradation of performance, affected by noise, attenuation, or inadequate illumination caused during adverse weather conditions. These challenges arise from their operating principles which makes them inherently vulnerable to such situations. Conversely, the unique features of Radar sensors have shown potential in tackling these shortcomings [250]. Despite the recent attention toward Radar sensors, they are still considered under-explored compared to cameras and LiDARs. The number of benchmark models based on cameras and LiDARs far exceeds those utilizing Radar sensors [80, 133]. Consequently, further research is required to develop mature Radar-aided perception models. This highlights a knowledge gap and the need for more comprehensive studies on Radar-aided perception.

Furthermore, the DL approaches used for constructing reliable perception rely on a vast volume of data, often provided through extensive datasets for training and learning how to perform perception tasks. However, out of more than 60 comprehensive perception datasets, under 10 of them include Radar recordings [144]. These recordings are often organized as point clouds where the sparsity of information limits the scope for in-depth studies on Radar's potential benefits. This introduces additional obstacles on the way of the research community to explore the advantages that Radar sensors can bring to the perception systems.

1.2 Motivation and Contributions

The motivation for this thesis arises from the unique characteristics of Radar sensors, which set them apart from other commonly used perception technologies like LiDAR and cameras. Unlike cameras and LiDAR, Radar operates by emitting electromagnetic waves at higher wavelengths, enabling it to detect objects while being less affected by adverse weather conditions. This gives Radar an edge in challenging environmental conditions such as rain, fog, or dust, where other sensors often struggle. Additionally, Radar has a broader range and performs consistently over long distances, making it a valuable tool in AV perception.

More specifically, the analog range and azimuth scans of Radars, which offer a continuous and rich representation of the environment, would be a great candidate to address the aforementioned challenges. This thesis aims to take advantage of this property by developing a method that transforms Radar's point cloud into a structured representation that mimics its scanning behavior.

To achieve this, the problem is formulated as an occupancy grid mapping (OGM) framework. This approach estimates the likelihood of different regions in the radar's field of view being occupied or free while accounting for the inherent uncertainty in Radar measurements. By utilizing the OGM framework, the research seeks to harness Radar's advantages and address the challenges by building high-quality, rich, and more informative maps for more robust perception. Toward this objective, the key contributions of this thesis are outlined as follows.

- **PARMG: Pseudo Analog Azimuth-Range Radar Maps Generator.** We designed a unified novel system for Radar-aided perception that addresses the information loss stemming from sparse sampling. Our approach significantly improves the quality of the Radar point cloud measurements by utilizing the temporal redundancy between frames and transforming them into OGMs. By projecting the environment onto a grid map, our system uses the Bayesian filtering algorithm to estimate the occupancy state of the grid cells. This estimation is performed through an inverse sensor modeling module, which is designed specifically for Radar sensors, to mimic how they perceive their surroundings. Although occupancy grid mapping is mainly developed for static environments, we extend our system to account for dynamic objects in driving scenarios. Furthermore, our framework supports arbitrary multi-sensor configuration and is capable of constructing maps from an aggregation of different views. Notably, our proposed framework can provide accessibility to high-quality OGMs across different Radar-inclusive datasets with slight modifications and facilitate the study of Radar-related studies on the various publicly available datasets.

- **Runtime optimization through multi-processing.** Constructing maps at high resolution with minimal quantization error requires substantially high computational power. In this contribution, we introduce a multi-processing scheme that performs occupancy mapping through parallel tasks, therefore optimizing the overall required processing time.

- **Multi-sensor synchronization.** In a sensor suite consisting of various kinds of sensors, the synchronization issue between sample outputs occurs due to differences in the sampling frequencies. Accordingly, we implement a multi-sensor synchronization scheme that upsamples the output of the sensor with a lower sampling frequency and associates the closest samples with the ones generated by the other sensor, therefore temporally aligning samples for consistent representation.

1.3 Thesis Outline

The remainder of this thesis is organized as follows. In Section 2 we provide a comprehensive overview of perception systems. This overview discusses the different types

of sensors used in perception systems and their key characteristics, as well as examples of well-known datasets suitable for different purposes. Furthermore, a categorization of perception-related tasks is given which is expected from AVs to perform, followed by the relative environmental challenges. Chapter 3 presents perception systems' basic concepts and comparative analysis of several object detection benchmarks using different sensor combinations. Our proposed framework is discussed in Chapter 4 explaining the mechanisms used for OGM construction in detail. Chapter 5 provides an in-depth experimental analysis of the characteristics and the effectiveness of our proposed method both in generating an informative representation of environments and enhancing detection performance under adverse weather conditions. Finally, a conclusion of our work is provided in Chapter 6 highlighting advantages and the areas requiring further improvement accompanied by future research directions.

Chapter 2

Background and Related Work

2.1 Introduction

Autonomous Driving (AD), once considered a distant aspiration, is significantly close to becoming a reality owing it to recent advancements in hardware, software, and communication technologies. Connected vehicles, enabled by vehicle-to-vehicle (V2V) and vehicle-to-everything (V2X) communication systems, play a critical role in facilitating seamless interactions among vehicles and between vehicles and infrastructure. These technologies enhance situational awareness, enable cooperative driving, and improve traffic management, contributing to safety and efficiency [3, 8–12, 24, 27, 39–42, 44, 45, 48–51, 53, 55–58, 60, 61, 63, 64, 66–72, 74, 76, 85, 101–107, 111, 115, 147, 215, 230, 234, 235, 263, 293, 300, 301, 312, 313, 315, 353, 354, 368]. On the other hand, the core of autonomous driving lies in the vehicle's ability to accurately perceive its environment. The perception system, responsible for sensing and interpreting the vehicle's surroundings, is fundamental to safe navigation and informed decision-making. This system must be robust, real-time, and capable of handling complex, dynamic environments to ensure the vehicle can respond appropriately to obstacles, other road users, and varying traffic conditions. AVs are poised to revolutionize transportation, offering enhancements in safety, efficiency, and environmental sustainability.

Perception systems in AVs include various sensors, including LiDAR, Radar, and cameras, which gather information from the vehicle's surroundings [133]. However, the representation of the world acquired from these sensors is raw and cannot be used directly for decision-making or navigation. This raw data should undergo complex processing and analysis, utilizing advanced algorithms and techniques, to extract high-level information. Tasks such as object detection, lane tracking, and pedestrian recognition are essentially performed by the perception systems, and the next stages mainly depend on the results from this stage.

In recent years, significant advancements have been made in perception systems, fueled by the rapid development of artificial intelligence and sensor technology [144]. Deep Learning (DL) algorithms, in particular, have revolutionized perception systems,

allowing AVs to perceive and interpret their environment with higher accuracy and speed, and in a more generalized way [133]. Despite these advancements, challenges remain in the way of achieving robust and reliable perception systems for AVs. Two of the most crucial practical challenges are the effect of adverse weather conditions and nighttime illumination on the performance and robustness of these systems [200].

The studies in the field of perception systems review different components from various aspects. *Guo et al.* [144] conducted their survey with a focus on driving scenario challenges. They introduce a framework with factors to measure the safety and the "driveability" of driving conditions. The authors provide a comparative review of various publicly available datasets for different perception tasks. *Muhammad et al.* [220] gave a survey focusing on semantic segmentation in scene understanding based on visual sensory data. They provide critical analysis of the performance, time complexity, and limitations of state-of-the-art approaches while also discussing available datasets and unresolved challenges. *Fend et al.* [133] surveys DL techniques for multi-modal object detection and semantic segmentation and highlights key challenges in combining data from cameras, LiDARs, and Radars for robust performance. *Xie et al.* [333] reviews point cloud semantic segmentation techniques, comparing traditional and DL methods while highlighting challenges like dataset limitations and noise. It suggests future research should improve robustness and address noise issues in practical applications. *Arnold et al.* [19] surveyed 3D object detection methods for AVs, highlighting their advantages over 2D methods for tasks requiring depth information. It categorizes recent approaches based on sensor modalities and summarizes the results to identify research gaps. *Guo et al.* [145] provides a comprehensive study on the recent advances in DL for point clouds, focusing on 3D shape classification, object detection and tracking, and segmentation accompanied by comparative results from public datasets.

2.2 Preliminaries

The Society of Automotive Engineers (SAE) has created a framework that defines vehicle autonomy levels from manual driving (Level 0) to full autonomy (Level 5) [255]. Achieving full autonomy requires the vehicle to have a deep understanding of its surroundings, accurate positioning, and safe navigation under all conditions. In fully autonomous vehicles, passengers do not need to control their driving. The vehicle's operations involve several stages, starting with sensors collecting raw data and progressing through perception, localization, behavioral estimation, and decision-making, ultimately guiding the vehicle safely from start to finish.

Scene perception provides awareness of the driving scenario and the entities involved, while Simultaneous Localization and Mapping (SLAM) focuses on determining the vehicle's position and creating a detailed map of the surroundings in real-time. For example, [285] offers a comprehensive overview of SLAM algorithms based on visual sensors, discussing challenges and limitations in achieving full autonomy, and emphasizing the

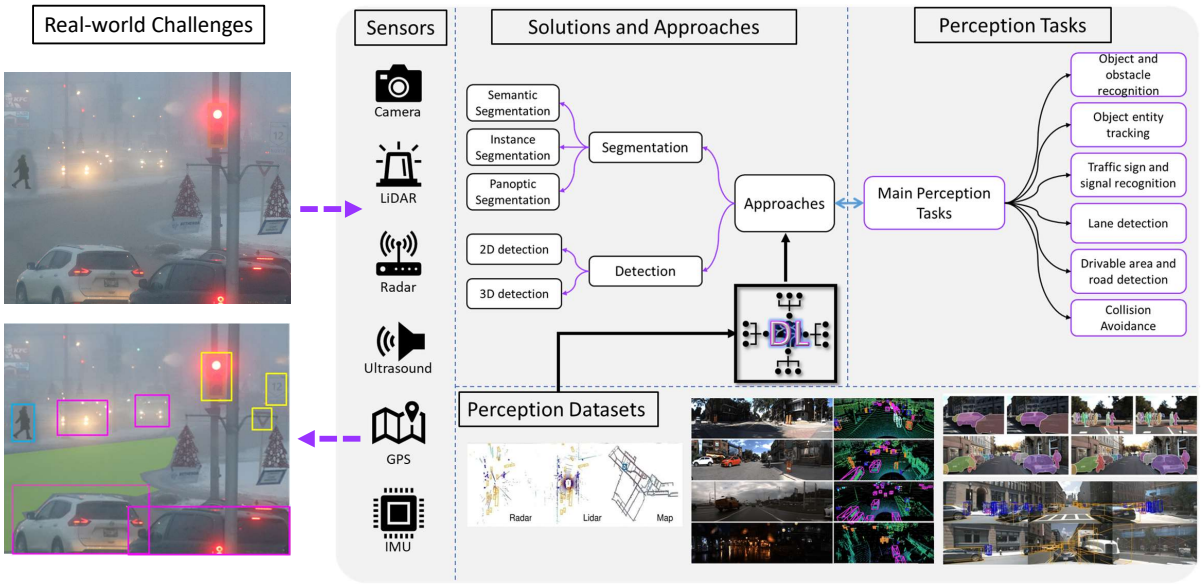


Figure 2.1: This figure provides an overview of the tasks and challenges in perception systems, starting with the sensors and moving through algorithms and approaches. It highlights the use of deep learning methods and publicly available driving scenario datasets. The input picture is adapted from [17].

need for self-learning models to handle dynamic environments. Similarly, [213] reviews visual-based SLAM techniques, including visual-only, visual-inertial, and RGB-D SLAM, highlighting their algorithms, advantages, and limitations, and providing guidance on selecting appropriate methods based on application needs. Additional studies [77, 97, 277] further explore SLAM advancements. After perception and SLAM stages, the data is used for behavioral prediction of road participants, which is crucial for navigation planning, as discussed in [20], [224], and [137]. With this high-level understanding, AVs can issue control commands to actuators and navigate safely.

To understand the surrounding events, an AV must perform several tasks such as recognizing pedestrians, understanding traffic signs, detecting roads and lanes, and ensuring safe driving trajectories. Historically, early object detection methods relied on visual sensors like cameras and used traditional techniques for object identification [84, 209, 283]. For instance, [7] combined radar and vision data to enhance vehicle detection and reduce false positives. Other approaches used SVMs, such as linear SVM with HOG descriptors [109] and latent SVM with multi-scale deformable part models [132].

Recent advancements in Deep Learning (DL) have led to more robust and accurate solutions, breaking down perception tasks into smaller, manageable components [357]. Initially, perception issues were tackled using vision-based methods, which can generally be divided into two categories: detection and segmentation. Detection focuses on identifying and localizing objects in an input frame by drawing bounding boxes and providing

confidence scores for classification. Segmentation, on the other hand, involves dividing an image into segments or regions based on certain criteria to simplify or enhance analysis [133]. A study [220] highlights scene understanding using only vision sensors, detailing the generic pipeline of scene understanding and offering a comprehensive review of recent methods and their performance.

The evolution of perception systems in autonomous vehicles has moved from vision-only methods to integrating various sensors, driven by advancements in deep learning. Moving from the initial reliance of perception on cameras for object detection and segmentation, advancements have made additional sensors like LiDAR and Radar more cost-effective [310]. These sensors offer complementary data, enhancing accuracy and robustness [167]. Deep learning has been crucial in this transition, facilitating the fusion of multi-sensor data for better scene understanding and situational awareness. This multi-modal approach addresses sensor limitations and leverages their strengths, resulting in more reliable and effective perception systems [133].

2.3 Perception Sensors

The sensory interface of every self-driving vehicle is the first layer of the perception system. These devices gather data from the surrounding environments by interacting with them in different ways. These data are to be further processed for the extraction of high-level information that is a requirement for the decision-making procedures. Each type of sensor is distinct in how it captures the events of the surroundings which determines the advantages as well as the handicaps. In this section, most representative sensors are introduced and their working principles are investigated. By delving into the inner workings of these sensors, readers will gain a deeper understanding of how they perceive and interact with their surroundings. This knowledge will not only provide valuable insights into the capabilities of each sensor but also point out their potential applications across various scenarios and environments suitable for AVs.

2.3.1 Camera

Cameras are the most common sensing modality used in AVs for many purposes. They capture the surrounding environment in the form of images from which contextual information can be derived. There exist different types of cameras such as RGB, Thermal, Stereo, Gated, FIR, etc [32, 35]. A proper type should be selected concerning the application [266]. However, *RGB* cameras are the most widely used sensors in perception applications.

RGB cameras provide 3-channel colored images, namely, Red, Green, and Blue, and are similar to human beings' sight mechanism which captures visible light with the wavelength of 380 to 700 nm [5]. These cameras are offered in different settings such as short-focal length which provides a wide angle of observation whereas long-focal length

which enables the camera to focus on objects from far away [86]. A stereo vision setting, as opposed to using only one single camera, is sometimes used to help with depth recognition [168]. Thermal cameras are another popular type commonly used on AVs. These sensors capture radiation of infrared waves which are not visible by human eyes. The output is images that contain heat information of the objects existing in the respective frame [302].

2.3.2 LiDAR

LiDAR (Light Detection And Ranging) sensors can be considered a cutting-edge technology used in the development of AV perception systems, ranking as the second most popular sensor type after cameras. LiDARs operate on principles different from those of cameras which allows them to capture detailed three-dimensional information about the environment, offering unique advantages for perception in autonomous driving systems. LiDARs rely on measuring the total travel time of laser beams emitted from the sensor to the surrounding objects and back which is then used to convert this data into precise distance measurements. The operating wavelength of LiDAR beams is commonly reported to be 905 or 1550 nm which is not visible to human eyes [198]. The results of the mentioned spatial sampling form a 3D point cloud and include point-wise depth information.

Compared to the cameras, LiDARs, typically, have lower spatial resolutions and are not able to capture fine spatial textures. However, they provide 3D contextual information and are robust to changes in illumination and lighting conditions, which can be advantageous in certain scenarios. There is a trade-off between LiDAR's inherent capabilities and shortcomings and the decision to include them in the sensor suite of an AV depends on the application as well as the type of challenges that are aimed to be tackled.

2.3.3 Radar

Radar sensors, short for Radio Detection and Ranging, operate on principles similar to those of LiDAR systems. They emit electromagnetic waves at a specific frequency from the transmission antenna or antenna arrays. These waves are reflected by the obstacles present in the surrounding environment which cause changes in transmitted signal properties, such as phase shifts or modulation frequency. Upon returning to the sensor, signal processing modules analyze these changes to extract information, including the location and velocity of objects. Traditionally, Radar systems were large and heavy, making them unsuitable for mobile applications. However, advancements in technology have enabled cost-efficient and lightweight implementation of both Radar hardware and techniques that make it suitable for AV use.

Among these advancements, Frequency Modulated Continuous Wave (FMCW) Radars have emerged as a notable development. Building on foundational research [267, 294, 295],

FMCW Radars offer significant advantages, including cost-efficiency, effective detection capabilities, and low hardware requirements. These features have contributed to their growing popularity in the field of autonomous vehicles [2, 373]. FMCW Radars' operating wavelength is in the millimeter range which has its own benefits and drawbacks. Due to their wavelength of operation, Radars are low in spatial resolution which makes it challenging for tasks such as object classification. However, with a greater detection range and far more robustness to adverse weather conditions such as heavy rain, snow, or fog, compared to cameras and LiDARs, they can be advantageous in numerous scenarios as a complementary modality [254]. Additionally, illumination and lighting conditions do not affect their functionality, therefore, there will be no degradation of performance at nighttime. Despite these benefits, Radar sensors are still considered under-explored, compared to cameras and LiADR. Therefore, further study is required for the development of more mature Radar-aided models to address perception challenges. Examples of Radar-inclusive datasets are given in Table. 2.12 and a comparative analysis of the camera, LiDAR, and Radar sensors are depicted in fig. 2.2.

2.3.4 Ultrasound

Exploiting mechanical waves, Ultrasound sensors have a similar working principle to LiDARs and Radars. By sending out high-frequency sound waves and measuring to total round-trip time of the wave, the distance to objects can be measured. However, due to the inherent properties of mechanical waves, they can be easily affected by changes in temperature, atmospheric pressure, and the humidity of a place. Compared to the propagation speed of electromagnetic waves through free space, approximately $3 \times 10^8 m/s$, the propagation speed of sound through air, approximately $330 m/s$, is far less. Therefore, this limits the application of ultrasound sensors to low-speed and near-range scenarios like automatic parking systems [133], which are already exploited in modern vehicles nowadays.

2.3.5 GNSS and IMU

Unlike the previous sensing modalities, which were mostly used for local scene understanding, GNSS-related sensors are mostly used for localization purposes. According to the EUSPA, GNSS (Global Navigation Satellite Systems) refers to a system of satellites, with a global coverage, that provides geo-spatial information such as position and time. GNSS instances of regional positioning systems are the USA's GPS, Europe's Galileo and Russia's GLONASS [128]. GPS sensors are commonly used to find global positioning parameters of the ego-vehicle, dubbed latitude and longitude, and in some applications are jointly used with HD maps for predicting the path of other entities on the road and also planning and navigating the ego-vehicle [21]. Moreover, Inertial Measurement Units (IMU) also provide information about the ego-vehicle states rather than the surrounding environment. Information such as body acceleration, rotation rates, and magnetic field

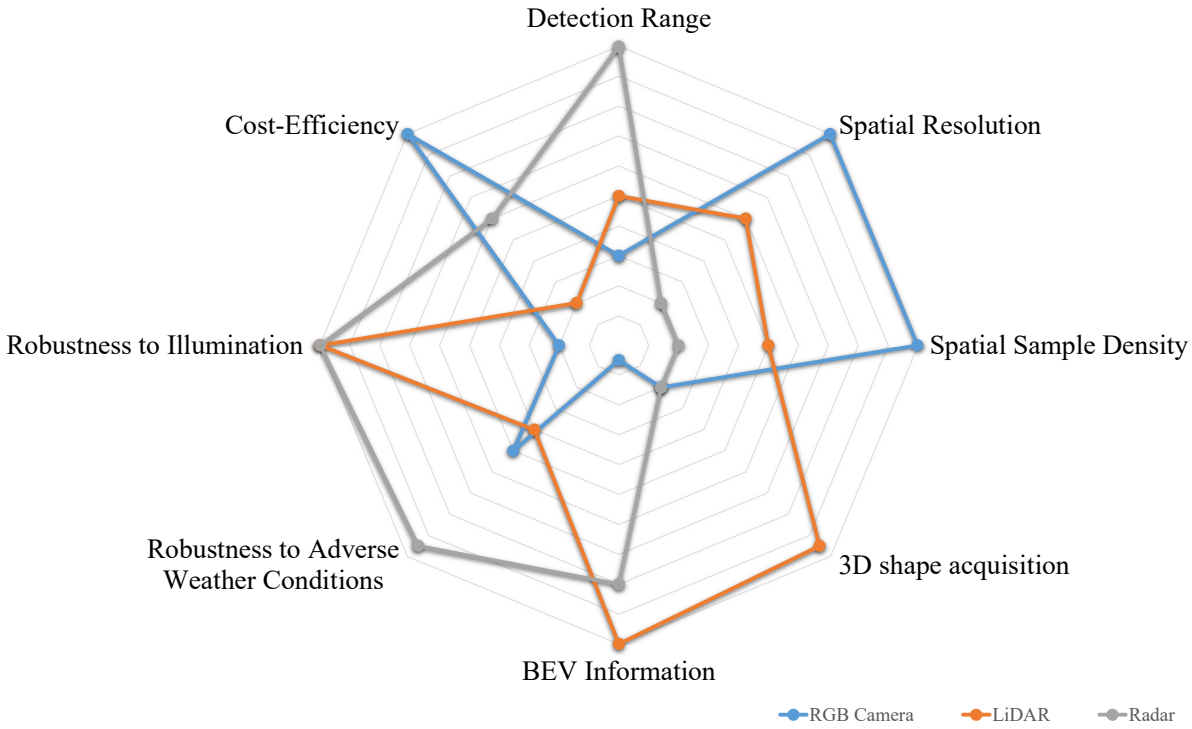


Figure 2.2: Critical comparison of camera, LiDAR, and Radar sensors with respect to their key features including detection range, spatial resolution, spatial sample density (as opposed to sparsity), 3D shape acquisition, bird's-eye view (BEV) information, robustness to adverse weather conditions, robustness to illumination, and cost-efficiency.

measurements are captured by the micro-accelerometers and micro-gyroscopes [1] which then can be used for localization and state-estimation purposes with GPS data as the counterpart. The information provided by the IMU unit is essential to AD development. It is the backbone for most of the sensor calibration, synchronization, and fusion methods and plays a vital role in the accurate perception and decision-making of self-driving vehicles [117, 118, 323].

Table 2.1: Summary of the characteristics, applications, advantages and shortcomings of commonly used sensors.

Sensor	**Features**	**Application**	**Strengths**	**Shortcomings**
Camera	· Range ≈ up to 75 m · Operating Wavelength: 380-700 nm	· Detection, Segmentation · Suitable for main sensing modality	· High spatial resolution · Mature models available	· Sensitive to: -Weather condition -Illumination and lighting condition
LiDAR	· Range ≈ up to 200 m · Operating Wavelength: $905, 1550$ nm · Update rate: 20 Hz [26]	· Detection, Segmentation · Suitable for secondary or main sensing modality	· Including depth information · BEV representation · Mature models available	· Sensitive to Weather condition · Sparse response for far objects · Medium spatial resolution
Radar	· Range ≈ up to 500 m [251] · Operating Wavelength ≈ 4 mm · Update rate: 4-13 Hz [26, 80]	· Detection, Segmentation [243] · Suitable for complementary sensing modality	· Robust to weather conditions · Very long detection range · Direct target velocity measurement · BEV representation	· Low spatial resolution · Small number of mature models available
Ultrasonic	·Sound waves freqency ≈ $40 \sim 70$ KHz ·Range ≈ up to 5 m [310]	·Autonomous parking system ·Short-distance obstacle detection	·Accurate short distance measurement ·Cost-efficient	·Not robust to changes in temperature, humidity and atmospheric pressure ·Response latency when measuring long distances
GPS	·Location accuracy ≈ $5\,m$ [225]	·Global positioning and localization ·Suitable for sensor fusion schemes	·Widely accessible ·Cost-efficient	· Vulnerable performance when environmentally challenged phenomena such as Urban Canyons [20]
IMU	· Update rate ≈ 200 Hz [285]	·Dead-reckoning, localization ·Suitable for sensor fusion schemes	·Widely accessible ·Cost-efficient ·Real-time performance	·Error accumulation over long run

2.4 Perception Dataset

The performance of a perception model depends on many factors, among which data quality plays an important and pivotal role. In learning-based approaches, the efficacy of the models heavily relies on the data used for training, impacting their ability to generalize, handle diverse scenarios, and maintain robustness. Therefore, the availability of well-organized and diverse datasets is essential as they are valuable resources to study the ways to tackle the challenges of autonomous driving. Researchers have put forward effort to collect useful data in various ways and build datasets while having the needs of autonomous driving in mind.

Many datasets and benchmarks have been gathered and organized in the literature [133, 144, 357, 373] and are often accompanied by critical analysis and suggestion for relevant applications. In this section, the focus is put on a number of relatively more recent datasets suitable for perception tasks. These datasets have gained popularity in the research community due to the unique properties they bring to the table that make them suitable for addressing specific challenges.

2.4.1 BDD100K

University of California, Berkeley introduced a large-scale diverse dataset dubbed as *BDD100K* [355]. This dataset is built on over 100,000 video clips recorded by Nexar dashcam at 720p resolution and 30 frames per second. It comes in 1.8TB of volume and unlike many past datasets that were limited to only one city or at a certain period of daytime, BDD100k offers recordings in a wide variety of conditions. The videos are collected in four different cities in the USA, New York, Berkeley, San Francisco, and the Bay Area. The dataset also contains approximately the same number of daytime recordings as nighttime and considers different weather conditions, such as clear, partly cloudy, overcast, rainy, snowy, and foggy. The variety of recording conditions allows the research community to explore a more general and realistic domain compared to previous datasets. BDD100K data can be used for a variety of perception tasks such as object detection, segmentation, and the applications of drivable area and road detection, object tracking, and image tagging.

2.4.2 ApolloScape

ApolloScape is a large-scale dataset designed for various autonomous driving tasks [163]. This dataset includes over 140,000 RGB video frames, dense labeled 3D point clouds, and depth maps generated for each image frame which creates pixel-level depth annotation. Utilizing video cameras and LiDARs, the data was recorded in the various urban areas in China, capturing various traffic conditions with objects ranging from small to large, pedestrians, motorcycles, and bicycles. The 2D object labels on the video frames can be used in object detection and tracking tasks. The recordings encompass differ-

ent weather conditions and extreme illumination scenarios, making the dataset suitable for studies under challenging conditions. One of the unique features of this dataset is its extensive semantic-level annotations which cover 25 object categories, making it an excellent resource for segmentation-related studies. Additionally, there are 28 classes of lane markings labeled, which, combined with high-definition maps, can be used as a backbone for road detection/segmentation and auto-driving and navigation tasks.

2.4.3 The Oxford RobotCar Dataset

This challenging large-scale dataset comes in two parts. The first part of *The Oxford RobotCar Dataset* is built with a focus on long hours of driving conditions [214]. Data collection was performed over the period of one year in central Oxford, UK. For data collection, the platform vehicle traveled a designated route repeatedly which resulted in capturing the variant nature of the urban environments. As a consequence, collected data is rich and diverse in the sense of weather conditions as well as time of the day. As opposed to uni-modal datasets, the Oxford RobotCar platform vehicle was mounted with 6 cameras, LiDARs, GPS, and INS sensors which enables the platform vehicle to acquire 360-degree visual coverage, 2D and 3D scans, and accurate positioning information.

The second part of this dataset was released in 2019, recorded in the same setting as the first part, encompassing a volume of 4.7TB of data [26]. However, the second part builds upon the first part and adds azimuth-range Radar scans. The platform vehicle comes fairly with the same sensory setting with the addition of an mm-wave Radar. Adding the new modality to the sensor suite, the dataset targets a broader audience of researchers who are seeking to exploit Radar benefits in perception systems. This approach has recently gained popularity in the community.

Overall, this multi-modal dataset is a valuable contribution to the research community for studying challenges in the object detection and tracking, semantic segmentation domains, particularly, for those focusing on modality-fusion-based solutions under different weather conditions.

2.4.4 NuScenes

The *nuScenes* dataset, a recent addition to the AD field, offers a large-scale and comprehensive collection of multi-modal recordings. The data is gathered in the urban environments of Boston and Singapore to capture their well-known heavy and complicated traffic conditions [80]. It features an all-encompassing sensor suite including 6 cameras, 5 radars, and 1 lidar, providing 360-degree coverage in all modalities, as well as GPS and IMU data for accurate localization and positioning information. This dataset comprises around a thousand 20-second video snippets called scenes and each recorded scene is further broken into keyframe samples and intermediate-frame samples also referred to as sweeps. The keyframe samples are fully annotated with 3D bounding boxes across 23 object classes and support various weather and daytime conditions.

In multi-modal datasets, one challenge arises from the synchronization issues in the sensor suite, as not every sensor operates at the same sampling frequency. To this end, the sample frame portion of the dataset is collected where every sensor recording is timely aligned across different modalities. Additionally, in the SDK offered by the authors, linear interpolation is adopted to regress the bounding boxes for sweep frames that reside between each two sample consecutive frames. All in all, the nuScenes dataset serves as a valuable resource for perception tasks like 3D object detection and tracking, and segmentation, as well as non-perception tasks like motion and trajectory prediction. This dataset supports studies in sensor fusion fields and challenging environmental situations such as adverse weather and nighttime.

2.4.5 Waymo Open Dataset

As a large multi-modal dataset, *Waymo Open* dataset is introduced with a focus on the quality of data and the diversity of conditions [299]. One of the issues that researchers find challenging in working with multi-modal datasets, is the lack of consistency or synchronization of data acquisition among sensors. In the Waymo Open dataset, synchronization and sensor calibration schemes are used to ensure data consistency between data points, which is one of the advantages compared to the others. Similar to the recently released datasets, the Waymo Open dataset cares for diversity and includes recorded data from six different geographic locations in the US, in a wide variety of environmental, daytime, and weather conditions.

This dataset can be divided into two partitions called the Perception and the Motion sets. The data belonging to the perception set is collected with a sensor suite of five high-resolution cameras and four short-range and one mid-range LiDARs the result of which is in the format of 20-second long clips recorded from more than 2000 scenes. On the other hand, the same combination of LiDARs accompanied by eight cameras was used to record more than 570 hours of data for the purpose of motion forecasting and trajectory prediction tasks. It includes sequences of vehicle poses and associated timestamps, along with annotations that predict the future trajectories of objects in the scene [249]. This dataset supports the development and testing of advanced perception algorithms, path planning, sensor fusion techniques, and behavior prediction models. By offering data captured under various weather conditions and lighting scenarios, the Waymo Open Dataset is a valuable source in adverse weather studies.

2.4.6 Argoverse

The *Argoverse* dataset comes in two parts namely, Argoverse1 and Argoverse2. The initial Argoverse dataset was recorded in Pittsburgh and Miami to account for distinct local driving habits and challenges [88]. Furthermore, the dataset is diverse in times of day and weather conditions as it is recorded across various seasons. The platform vehicle is equipped with 7 ring cameras, 2 forward stereo cameras, and 2 LiDARs which

are mounted on the roof. The dataset contains three types of maps, 3D annotations for tracking, and vehicle trajectories. Consequently, 3D object detection and tracking, segmentation, and scene parsing into roads, sidewalks, and vegetation as well as motion and behavior prediction are among the applications the dataset caters to. Moreover, Arogverse2 builds upon Argoverse1 by expanding its properties and offerings [327]. The second version comes in four partitions which are called, the Sensor Dataset, LiDAR Dataset, Map-change Dataset, and Motion Forecasting Dataset. It is recorded in 6 different cities and is far larger in volume of samples and scenes compared to its prior version.

It is noteworthy to mention that class imbalance is a well-known issue among learning algorithms. When one class of data significantly outnumbers the others in a dataset, it can lead to biased models that perform poorly in predicting the minority class. To this end, the Sensor Dataset, which covers about 1000 scenes coupled with HD maps, has been organized in a way to fight off skewness in the class distributions. Additionally, the Arogverse2 LiDAR Dataset includes more than 20,000 scenes that carry HD maps and can be considered one of the largest open-source collections. It can be beneficial in research related to performing self-supervised learning with LiDARs. The map change dataset consists of HD maps, with two hundred scenarios that have recorded changes in real-world environments. The Motion Forecasting Dataset includes over 250,000 scenarios with a span of 11 seconds and 10 non-overlapping classes of static and dynamic actors.

2.4.7 PixSet

LeddarTech Holding Inc. introduced their publicly available dataset named *PixSet* in 2021 to help with solving sensing, fusion, and perception challenges [121]. The sensor suite used for data collection includes four cameras, one solid-state flash LiDAR, one mechanical LiDAR, and one FMCW Radar which are all closely positioned on the platform vehicle. These sensors are accompanied by IMU and GPS sensors for tracking and odometry purposes. The notable innovation of this dataset is the inclusion of full-waveform data from the solid-state flash LiDAR, in contrast to the common point cloud format. The sensor captures two waveforms for each channel of every frame, with one being high-intensity and the other a quarter of that intensity, effectively expanding the dynamic range of the sensor. This feature mimics the behavior of high-dynamic range imagery in cameras, which is accomplished by blending images captured at different exposure levels.

The data recording is done in high-density urban areas of Quebec City and Montreal during summer. The dataset comes in almost 29,000 frames and 20 object categories with 3D bounding boxes. It features various city locations, normal and cloudy weather conditions, and a portion of rainy scenes, predominantly during the daytime with some nighttime footage. This dataset supports a range of applications including object detection, full-wave LiDAR research, depth estimation, and sensor fusion studies.

Table 2.2: Summary of the sensor suite, key features, and the applications of the discussed publicly open datasets for autonomous driving. In this table, C, L, and R correspond to Camera, LiDAR, and Radar, respectively.

Dataset	**Sensors** C	L	R	**Key features**	**Application**
BDD100K	✓			· Front view RGB images · Diverse in: Weather conditions Time of day Location	· Detection · Segmentation · Object Tracking
ApolloScape	✓	✓		· 360° coverage · Diverse in: Weather conditions Illumination conditions Location · Lane markings ·Semantic label of 25 classes	· Detection · Segmentation · Object tracking · Auto-driving ·Fusion models
Waymo Open Dataset	✓	✓		· Multiple datasets · 360° coverage · Diverse in: Weather Conditions Time of day Location	· Detection · Segmentation · Motion forecasting · Fusion Models
Argoverse 1&2	✓	✓		·Multiple datasets · 360° coverage · Balanced classes in Argoverse2 · Diverse in: Weather Conditions Seasons Time of day Location	· Detection · Segmentation · Out-dated map detection · Motion forecasting · Fusion models
DENSE	✓	✓		·Multiple datasets · 360° coverage · Including Gated Cameras Thermal Cameras Stereo Cameras · Diverse in: Weather Conditions Time of day Location	· Detection · Depth Estimation · Image Enhancement · Weather Condition · Fusion models

nuScenes	✓	✓	✓	· Multiple datasets · 360° coverage · Diverse in: Weather Conditions Time of day Location	· Detection · Segmentation · Object tracking · Motion forecasting · Fusion models
Oxford Robot-Car	✓	✓	✓	·Multiple datasets · 360° coverage · Range-Azimuth Radar Scans · Diverse in: Weather Conditions Seasons	· Detection · Segmentation · Fusion models
PixSet	✓	✓	✓	· Full waveform LiDAR data · 360° coverage · Diverse in: Weather Conditions Time of day Location	· Detection · Fusion models · Full wave LiDAR research

2.4.8 DENSE

The *DENSE* project is a comprehensive collection of datasets, each with its unique features, to facilitate the studies on weather condition impacts on perception systems from different aspects. One segment, *Gate2Depth*, provides researchers with images recorded by low-cost CMOS gated cameras that are transformed into depth maps that achieve high resolutions comparable to LiDARs [143]. Another component, *Seeing Through Fog*, is introduced for object detection purposes [33]. It includes different weather conditions recorded in real-world scenarios and simulated in weather chambers. Different camera types and a LiDAR build up the sensor suite used for data collection. This dataset is a rich resource for researching signal enhancement and fusion algorithms. Complementing this dataset is an extension, *Gated2Gated*, that includes temporal gated imaging data which provides depth information and detailed scene annotation [316].

Another partition is named *Pixel Accurate Depth Benchmark* provides an evaluation benchmark platform for depth estimation tasks. Mono and stereo camera setups and a LiDAR scanner are used for data collection [142]. The final segment of the DENSE project is a dataset dedicated to the study of enhancement algorithms for image data, known as *Image Enhancement Benchmark*. This dataset addresses the inadequate availability of realistic data under different weather conditions by providing a novel benchmark. It consists of images captured under controlled real weather conditions within a weather-controlled chamber, complemented by relevant evaluation metrics. [36].

This group of datasets supports diverse studies in object detection by providing detailed environmental information, including weather and road conditions. They are

valuable for depth estimation and mapping, image enhancement methods, and sensor fusion research. The datasets offer extensive sensor suites and cover a variety of weather conditions, enabling comprehensive analysis and development of robust algorithms for autonomous vehicles.

2.5 Perception Tasks

The first stage of tasks that a self-driving vehicle is required to perform are those to understand the surrounding environment. These tasks are often called perception tasks, the representative examples are brought in figure 2.1. More precisely, during the perception phase, the vehicle uses the raw data gathered by its sensors and processes them to identify both static and dynamic entities in the environment. Information such as the type of the road participants, which could be a pedestrian or another vehicle, as well as their properties such as velocity and their location with respect to the ego-vehicle is often of interest. Moreover, for an AV to perform safe navigation on the streets, is it essential to detect the driving lanes and drivable areas as well as the traffic signs to abide by the traffic rules. Identifying the mentioned parameters is often a challenging task, especially because of the dynamic nature of the environments. Therefore, to study and address the complex problem of perception, it is broken into several small tasks allowing researchers to tackle them individually and one at a time.

In this section, the focus will be put on the most essential tasks, their respective challenges, and various methods proposed to overcome these challenges. The reader must note that, although these tasks will be defined distinctly, many models demonstrate the capability of performing more than one task.

2.5.1 Object Detection

Object Detection in the context of AV is the act of locating instances of objects in a scene observed by the sensors mounted on the car. The goal is not only the recognition of object categories present in the observed frame but also to precisely locate them by drawing a fitting bounding box. This task is commonly performed on images taken by cameras. However, cameras are not the only sensors used for this purpose. LiDARs, second to most popular, and Radars, recently gaining popularity, are other modalities that can be used to detect the objects around the ego-vehicle, each of which has its advantages and drawbacks. In that direction, different approaches are adopted and a variety of models are designed to perform detection. The object detection problem is mainly addressed in two ways, namely, 2D and 3D object detection. The approaches are different and can be mostly categorized into different groups with respect to the combination and types of sensor modalities they use to acquire and process data. Some models have only used a single modality whereas some have come up with multi-modal solutions. A summary of the main idea, challenges, and application of object decision

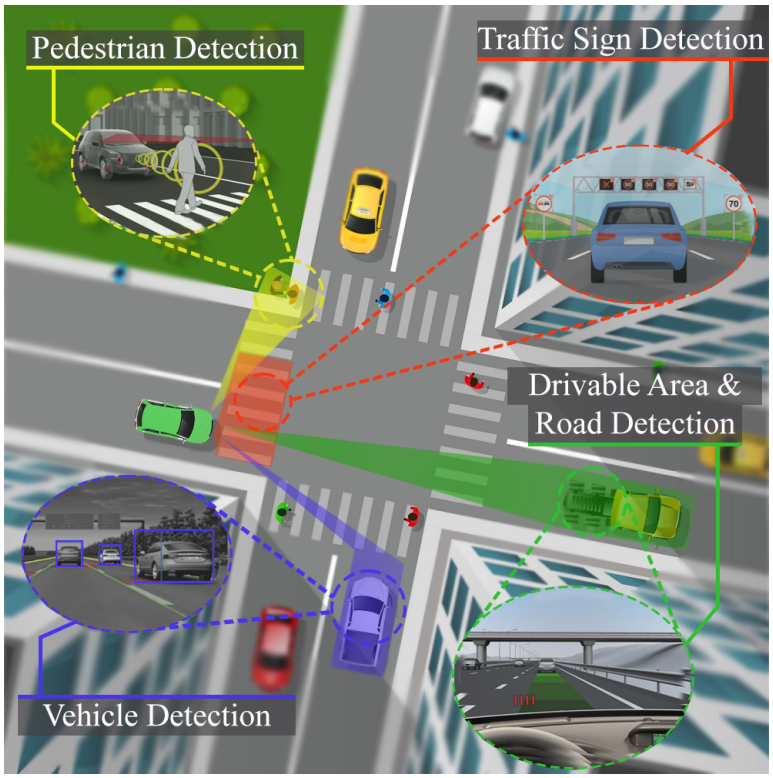


Figure 2.3: Examples of perception tasks. The original image in the background is adopted from [231] and modified regarding our desired presentation purposes.

tasks is provided in Table 2.3.

2D Object Detection

2D object detection is the problem of localizing object instances in a given frame by drawing an enclosing 2D bounding box accompanied by an object category. These 2D bounding boxes typically are determined in two distinct ways one of which is to specify a bounding box by its four corner points. The other method uses the center point (x,y) of the rectangle with the width and height as well as a rotation angle. Each of the definitions is used in the literature depending on the application, however, the latter is more popular. 2D object detection is a well-studied problem in the field of computer vision and has been mainly adopted for upright images captured by front-faced cameras. A summary of the main idea, challenges, and application of object decision tasks is provided in Table 2.3. YOLO is one of the most well-known algorithms that perform in real-time. The input image is divided into grid cells and with the help of a convolutional neural network (CNN) scheme, feature maps are extracted and bounding boxes and class scores are regressed [258, 259]. The Single Shot MultiBox detector (SSD) model focuses

Table 2.3: Overview of the object detection tasks, respective challenges, and their applications.

Approach	Main Idea	Challenges	Application
2D Object Detection	Identification and localization of object instances within an image using bounding boxes in a 2D space accompanied by a detection confidence score.	Lack of depth information, Detection of occluded objects in front-view images, Sensitivity to illumination and weather conditions.	Road participants detection (pedestrian, cars, bus, etc.), Traffic sign and signal recognition, Lane detection
3D Object Detection	Identification and localization of objects in a 3D space, often using data from multiple sensors (LiDAR, Radar, and cameras) accompanied by depth and rotation information.	Computational complexity, Proper sensor fusion method, Sensitivity to illumination and weather conditions	Road participants detection (pedestrians, cars, buses), Obstacle detection (traffic cone, barrier), Parking assistance, 3D environment mapping.

on multi-scale feature map extraction which enables the algorithm to detect objects of different sizes more accurately [203]. Even though SSD is slower in performance compared to the YOLO models, it can localize objects more accurately and detect small objects better. Being able to keep a fair balance between speed and accuracy, SSD is used as the head module (detector) in various more complex detection algorithms [184, 284, 340, 363]. Moreover, the proposal of the Regional-based Convolutional Neural Networks (R-CNN) family revolutionized the computer vision community. The first model was introduced in 2014 and was a two-stage object detector [138, 139, 262]. This approach first generates proposals of the regions with the probability of being occupied by object instances using a selective search algorithm. Then, using CNNs, features are extracted from the regions of interest (RoI) and further processed to refine bounding boxes and regress class scores. The two-staged algorithms are generally more accurate than the single-staged ones such as SSD and YOLO, however, they are more computationally expensive due to their architecture.

Furthermore, with the emergence of LiDAR and Radar technologies and the inclusion of LiDAR point clouds and Radar scans in datasets, many researchers have extended 2D object detection to 3D object detection and bird-eye-view (BEV) plane, that is a view of the XY-plane from the top. There are many mature LiDAR-based and multi-modal learning models and they mostly focus on 3D object detection from LiDAR point clouds which will be discussed next. However, there are some key ideas used for 2D object detection inspired by the working principle of these sensors.

Azimuth-range maps recorded by Radars are typically continuous and useful for localizing and estimating the velocity of objects [110, 189, 356]. For instance, one study [356] uses a Frequency Modulated Continuous Wave (FMCW) Radar exclusively for object

detection. The raw Radar data is pre-processed to create Radio Frequency (RF) images, with data from all chirps in a frame merged using an M-Net-based network, and Gaussian noise added to mitigate overfitting. The final output, a confmap, is generated through a RODNet backbone and processed by an inference module employing location-based non-maximum suppression to produce detection results. While Radars excel in challenging conditions like rain or snow, they are limited by poor spatial resolution, which restricts them from inferring distance, angle of arrival (AoA), and velocity but not the type or shape of objects. To address this, many researchers have adopted multi-modal approaches that integrate at least two different types of sensors. Additionally, some have focused on analyzing false positives caused by "Ghost Objects" in Radar point clouds [178]. Using a large-scale automotive dataset [274] with annotated ghost objects, they demonstrate the effectiveness of their method in detecting both ghost and real objects and show promise in reducing false positives caused by ghost images.

While traditional 2D object detection is rare in LiDAR-based algorithms, which usually focus on 3D detection using point cloud data, some fusion models enhance performance by combining information from multiple modalities. These models integrate Radar data with other sensors, such as cameras or LiDARs, to boost overall accuracy and robustness. For instance, the RadarNet algorithm [342] employs a fusion approach combining LiDAR and Radar data to detect objects, estimate velocities, and associate these velocities with the detected objects. This deep fusion model uses early fusion to merge the two data streams, which are then processed by a backbone network to generate feature maps. These feature maps are input to a fully convolutional detection head, which provides 2D velocity estimates along with classification and bounding box regression. To address ambiguity in Radar velocity estimation, the model uses an attention mechanism to align the radial velocities from Radar with those estimated during detection, followed by an aggregation step to refine and produce a robust final output.

To address the low spatial resolution of Radar sensors while leveraging their long-range and penetrative capabilities, researchers have combined them with cameras as complementary modalities [110, 189, 304, 361]. For example, one study [86] utilizes Radar's long-range detection to enhance distant object detection in frontal views. The study employs a simple fusion scheme where Radar complements the primary camera sensor. The model, based on ResNet blocks [151], processes Camera and Radar streams separately before concatenating them for feature extraction and further processing through ResNet blocks connected to detection heads. This approach shows performance improvements over camera-only systems, particularly for small or distant vehicles. Another method [221] enhances performance and addresses bottlenecks in Region Proposal Networks (RPN) by using Radar data to estimate the depth and velocity of objects, which informs the region proposal process. By incorporating Radar-derived information as prior knowledge, this method improves the model's focus on high-probability object regions, demonstrating the effectiveness of integrating Radar data.

Table 2.4: Overview of the attributes, strengths, and performance remarks of methods proposed for 2D object detection.

Ref	Sensor	Main Idea	Contributions	Remarks
[258, 259]	C	A CNN divides the input image into grid cells to regress bounding boxes and class scores for real-time detection.	Extremely fast with a unified architecture that generalizes well.	Higher localization errors, sacrificing accuracy for speed.
[203]	C	A single deep neural network predicts bounding boxes and class scores directly from multi-scale feature maps.	Fast and efficient performance, making fewer errors by handling objects of various sizes.	Less accurate than two-stage detectors.
[138, 139, 262]	C	Introduces Region Proposal Networks (RPN) to generate object proposals by sharing features with the detection network.	More accurate than single-stage detectors with nearly cost-free region proposals.	Computationally expensive and slower than single-stage detectors.
[356]	R	Uses FMCW Radar to transform raw Radar data into RF images, merging data for enhanced object detection.	Effective in adverse weather conditions with enhanced robustness.	Poor spatial resolution, often requiring multi-modal approaches.
[178]	R	Focuses on detecting and reducing ghost objects in Radar point clouds which occur due to reflection of EM waves.	Reduces false positives using a state-of-the-art classifier.	Dependence on the scope of the dataset and the auto-labeling system.
[342]	LR	Integrates Radar and LiDAR data with voxel-based early fusion and attention-based late fusion for enhanced detection.	Improves long-distance detection and addresses Radar's noise issues.	Unknown computational demand of the hybrid fusion approach.
[86]	RC	Enhances object detection performance by integrating Radar data with camera images, especially for distant objects.	Improves overall performance and the detection of distant and small objects through data fusion.	Potential noise in automatically generated labels and requiring more advanced Radar representation.
[221]	RC	Leverages Radar data to prioritize regions with higher object likelihood, improving both speed and accuracy.	Significantly faster and more precise than traditional region proposal algorithms.	Limited by the accuracy of Radar detections due to the dependence of proposals on Radar targets.
[189]	RC	Enhances RODNet's pedestrian detection accuracy using a generative model-based target-location algorithm.	Improves detection accuracy and recall significantly in dense situations.	Need for calibration method for position estimation and modification for the feature extraction.

3D Object Detection

3D and 2D object detection share similarities in problem formulation, object localization, and classification tasks. However, in the context of 3D detection, objects are localized using bounding cubes defined either with their corner points or with center points (x,y,z) accompanied by width, length, height, and a rotation angle around the three axes. This localization relies on various sensors. Cameras, LiDARs, and Radars are utilized, each offering unique attributes to derive the properties of objects. Consequently, three main approaches emerge: image-based, point cloud-based, and fusion-based methods. An overview of uni-modal and multi-modal approaches is provided in Table 2.5 and Table 2.6. Image-based methods are highly popular for object detection using cameras, as they utilize rich visual data. Among these, monocular algorithms are cost-effective and capable of estimating depth information during inference [204, 319, 366]. For example, the MonoPSR architecture [181] improves 3D prediction accuracy through a two-stage approach. Initially, feature maps are extracted from the input frame and used for early 2D predictions. These maps then facilitate proposal generation and pseudo-point cloud estimation, refining the final detection results. Another study [206] addresses the limitations of 2D region proposals by using key points 3D object centers projected onto the image plane—to refine 3D bounding boxes, thus enhancing detection accuracy. Additionally, ImVoxelNet [271] explores the benefits of multi-view images by accepting multiple RGB images, extracting features from each, and projecting them into a 3D volume. This approach uses the 3D volume in subsequent convolutional layers to regress final bounding boxes.

On the other hand, point cloud-based methods have proven to be helpful as they maintain the 3D representation of objects that are sampled and stored in a format of 3D point clouds. The most commonly used sensors for point cloud sensing and collection are LiDARs. Fusion-based algorithms aside, the next more accurate models are mainly LiDAR-based [80]. In the early stages, the main approach for object detection with point clouds was to identify point clusters which in turn would be introduced as object candidates [122, 155]. However, learning algorithms, specifically in the image domain, became a source of inspiration for point cloud object detection after their emergence.

Unlike images, point clouds consist of 3-dimensional points, making traditional 2D methods unsuitable for direct application. As a result, researchers have employed volumetric convolution layers to extract information from voxelized input frames [248, 332]. However, 3D convolutions are computationally more intensive than 2D convolutions, leading to performance bottlenecks. To address this, some studies have projected 3D points onto a Bird's Eye View (BEV) plane, leveraging established image-domain techniques for feature extraction [95, 196]. Additionally, other research has focused on the sparsity of point clouds, developing handcrafted features and novel convolution methods to enhance computational efficiency [127, 140, 317].

State-of-the-art methods build on past innovations to tackle challenges and enhance performance [184, 210, 346, 375]. For example, [349] introduces a two-stage architecture

that represents and detects objects as points. This method employs a standard 3D backbone for feature extraction, with a CNN detection head in the first stage for object detection and bounding box regression. In the second stage, MLP layers refine box regressions and produce confidence scores. SECOND [341] addresses the low performance and high complexity of voxel-based 3D convolutions by using an improved sparse convolution layer, significantly speeding up training and inference, and incorporating a novel angle loss regression to enhance 3D bounding box orientation estimation. Additionally, the use of transformers for 3D detection has been explored [23,130,216,359]. For instance, FocalFormer3D utilizes transformers to enhance prediction performance by focusing on challenging objects [96]. The pipeline extracts feature heatmaps at various resolutions, which are encoded for object queries, and employs a transformer decoder with self and cross-attention mechanisms for improved prediction.

The third popular approach in this field is fusion-based models, which leverage data from multiple modalities to address the limitations of each individual sensor. This method enhances accuracy and robustness, particularly in challenging conditions like sensor failures or poor illumination. Fusion can occur at various stages: early fusion combines data at the pixel level from the start, late fusion merges results after several processing stages, middle fusion integrates feature representations for further processing, and deep fusion involves multiple integration points throughout the architecture [133]. Each fusion strategy, such as addition, concatenation, or cross-attention, is selected based on the specific requirements of the application.

The Camera and LiDAR combination is a prevalent approach in object detection that benefits from the high spatial resolution of cameras and the depth and texture information from LiDARs [23,207,245,325,339,347]. For example, [246] introduces a three-stage model primarily using RGB input, supplemented by LiDAR depth data for 3D classification and localization of *Cars*, *Pedestrians*, and *Cyclists*. The first stage applies a 2D CNN to RGB images for detection and classification, then incorporates depth information to create frustum proposals. Instance segmentation is carried out in the second stage using PointNet [247], followed by 3D box regression via T-Net. A limitation of this approach is its sensitivity to lighting and occlusions, as the object detector relies solely on RGB inputs. FusionPainting, proposed by [337], improves upon this by adaptively fusing Camera and LiDAR data using an attention mechanism. The method processes RGB images and LiDAR point clouds separately, then learns an attention mask to integrate the data at the semantic level for 3D detection. Additionally, [197] explores a Deep Parametric Continuous Convolution approach that continuously fuses image features with BEV frames at various resolutions, culminating in predictions from a detection header. The AVOD method, described by [179], employs a two-stage LiDAR-Camera deep fusion architecture to enhance the detection of small road objects. It performs high-resolution feature extraction and fusion in the first stage, and merges image and BEV proposals in the second stage, refining 3D bounding boxes and classification results through an FCN.

Radar sensors have gained popularity recently due to their long-range detection and reliability, as detailed in Section-2.3. Researchers have explored various roles of Radar

in object detection [170, 185, 328]. For instance, [223] introduces a Camera-Radar fusion scheme for 3D object detection, utilizing a frustum-based expansion of Radar detections to the object's center point on the image plane. CenterFusion employs a middle-fusion approach, integrating feature maps from both image and Radar streams to refine predictions. Another study presents CRN, which utilizes images from multiple views and Radar points for feature extraction [173]. The image features are converted to BEV coordinates, guided by Radar measurements, and combined with BEV view feature maps to enhance semantic information. Additionally, [233] combines transformer layers with a middle-fusion scheme. Radar points and multi-view images are initially processed separately, and decoder layers create and refine attention masks based on object queries from the Radar stream. The final output is produced by a detection head.

Furthermore, some researchers have explored LiDAR-Radar sensor combinations, however, this field is still underexplored. While camera images provide rich semantic information, they are affected by illumination conditions, whereas LiDAR and Radar sensors function independently of ambient lighting. These sensors are inherently compatible and offer advantages in various scenarios [254, 282]. RadarNet is a deep-fusion framework that constructs BEV frames from separately rasterized LiDAR and Radar inputs, which are then concatenated for feature extraction at multiple resolutions [343]. This early-fusion approach incorporates both geometric properties and temporal information. The final predictions, including velocity estimations, are derived using a cross-attention-based late fusion mechanism. Additionally, research has addressed sensor failure challenges in fusion models with a modality-agnostic learning scheme for vehicle detection [191]. This approach extends beyond ideal sensor conditions to handle noisy or missing data and sensor failures during the Radar and LiDAR fusion process. It employs a Teacher-Student model comprising two modules: the Teacher, which uses reliable data, and the Student, which processes inputs with missing Radar and LiDAR data. Both modules feature encoders for each sensor stream, and their outputs are fused using self-attention and cross-attention blocks before being processed by a detection head for bounding box predictions. Training occurs in two phases: initially, the Student model learns from annotated data to train the Teacher module, and subsequently, the Student benefits from the Teacher's predictions, with the Teacher updated via an exponential moving average. This method builds on MVDNet [254] and demonstrates improved performance, showcasing the effectiveness of the novel framework.

Lastly, some methods utilize additional information sources such as ground maps [161, 344, 377] or integrate all three main modalities (Camera, LiDAR, Radar) [154, 350] to achieve superior performance. For instance, [195] developed a two-stage real-time model that performs 2D and 3D object detection, ground detection, and depth completion. This model begins with ground mapping using LiDAR point clouds to create BEV representations, which are then used in a dense fusion module. Depth completion is accomplished using image data and projected LiDAR depth information. This combination of road geometry, depth information, LiDAR, and Camera data enables the model to perform single-shot detection for high-quality 3D proposals, followed by RoI fusion for precise

2D and 3D bounding boxes. Another approach, proposed by [123], presents a modular fusion-based architecture combining Camera, LiDAR, and Radar for 3D object detection up to 225 meters away. The first module employs separate FPNs for each sensor to extract modality-specific features. These features are then transformed into the same BEV coordinates and aligned before being fused into a single feature map. The final detection head uses this fused map to produce 3D bounding boxes and classification results. The modular design offers flexibility, allowing for various BEV translation schemes, fusion methods, and sensor combinations. A novel study addresses limitations in current methods by managing uncertainties in categories and bounding box locations [257]. The researchers propose a Bayesian Neural Network and compare it with a Deep Neural Network to demonstrate its superior performance. Their single-stage architecture uses a deep-fusion approach, integrating camera, LiDAR, and Radar inputs for final detections.

2.5.2 Segmentation

Segmentation is a critical approach that enables autonomous vehicles to perceive their environment by obtaining fine contextual information and assigning pixel-level labels to the input frame. There are three main types of segmentation methods, namely *Semantic*, *Instance*, and *Panoptic* segmentation. At a normal traffic junction, the semantic segmentation method searches the scene for predefined classes of objects, and upon detection, all the pixels that are associated with the detected objects gain the same label or the class ID, typically illustrated with colored masks. For instance, all cars might be masked with one color, all pedestrians with another, and so on. Instance segmentation, on the other hand, also assigns different labels to the instances within the same class. For example, all detected cyclists who belong to the class of cyclists are annotated with different labels. Panoptic segmentation combines both previous approaches in one and parses the scene of interest to semantic-level masks with instance-level annotations [126]. A summary of the key features, challenges, and applications of different segmentation tasks is provided in Table 2.7.

The most commonly used sensors for these tasks are different types of cameras [287, 302], LiDARs, and a combination of these sensors. Furthermore, early segmentation methods predominantly relied on low-level visual properties and handcrafted features [79, 183, 286, 296]. However, the emergence of learning algorithms has enabled models to learn and extract complex features without previous design limitations, significantly advancing the field. To support and facilitate the research in this field, many datasets have been collected [29, 80, 100, 177, 226, 355].

Table 2.5: Overview of the attributes and strengths of uni-modal 3D object detection methods with relevant performance metrics reported in percentages.

Ref	Sensor	Main Idea	Strengths	Performance
[319]	C	Proposes a monocular 3D object detection framework that maps 3D targets to the image domain and decouples them into 2D and 3D attributes.	Achieves high performance cost-effectively by avoiding complex 2D-3D correspondence.	nuScenes dataset mAP= 35.8, NDS= 34.2.
[204]	C	Integrates 3D position information into image features for multi-view 3D object detection, levaraging 3D position-aware features to achieve state-of-the-art performance.	Achieves top performance on nuScenes, provides an efficient method to integrate 3D position into 2D features.	nuScenes dataset mAP= 43.4, 49.0; NDS= 48.1, 58.2
[206]	C	Directly predicts 3D bounding boxes from image data using a keypoint-based approach and bypassing the need for 2D region proposals.	Simplifies the detection process, improving accuracy and speed.	KITTI dataset $mAP_{3D,Moderate}$ = 9.76, $mAP_{BEV,Moderate}$ = 14.49, Runtime= 0.2[s]
[271]	C	Handles multi-view RGB-based 3D detection as an end-to-end optimization problem, achieving state-of-the-art performance across multiple benchmarks.	Effective for both indoor and outdoor scenes, and excels across multiple benchmarks.	KITTI dataset $mAP_{3D,Moderate}$ = 10.97, $mAP_{BEV,Moderate}$ = 16.37, Runtime= 0.03[s]
[149]	L	Single-stage detector that improves localization precision in 3D point cloud data by preserving object structure information.	Enhances precision and achieves high inference speed without additional computational cost.	KITTI dataset $mAP_{3D,Moderate}$ = 79.79, $mAP_{BEV,Moderate}$ = 91.03, Runtime= 0.04[s]
[346]	L	A lightweight, single-stage 3D object detector using a fusion sampling strategy and a novel box prediction network to achieve high performance with fast inference.	High accuracy and speed, outperforming voxel-based methods with fast inference.	KITTI dataset $mAP_{3D,Moderate}$ = 62.6, Runtime= 0.04[s], nuScenes dataset mAP=42.6, NDS=56.4
[375]	L	A center-based transformer network for 3D object detection using center heatmaps and cross-attention for multi-frame feature aggregation.	Reduces convergence difficulty and computational complexity compared to traditional transformers.	Waymo Open Dataset mAPH= 75.6
[184]	L	Omitting the need for 3D convolutions by organizing point cloud data into vertical columns and utilizing 2D convolutional layers for feature extraction.	Modifiable detection head, high detection performance achieving up to 105 Hz runtime.	KITTI dataset $mAP_{3D,Moderate}$ = 59.20, $mAP_{BEV,Moderate}$ = 66.19, Runtime= 0.016[s]; nuScenes dataset mAP= 30.5, NDS=45.3
[96]	L	Improves recall and reduces false negatives in 3D detection through a multi-stage process, enhancing robustness in challenging scenarios.	Better recall and robustness in occlusion and low-density point clouds.	nuScenes dataset mAP= 68.7, NDS= 72.6

Table 2.6: Overview of the attributes and strengths of multi-modal 3D object detection methods with relevant performance metrics reported in percentages.

Ref	Sensor	Main Idea	Strengths	Performance
[23]	LC	TransFusion uses a transformer-based method for LiDAR-camera fusion with soft-association to enhance 3D object detection.	Robust against poor image conditions and calibration errors, with state-of-the-art performance in detection and tracking.	nuScenes dataset mAP= 68.9, NDS = 71.7, Waymo Open Dataset mAPH= 65.5
[207]	LC	BEVFusion integrates LiDAR and camera features into a shared bird's-eye view representation for improved 3D perception.	Superior performance in 3D detection and BEV map segmentation with reduced computation cost and latency.	nuScenes dataset mAP= 70.2, NDS = 72.9, Waymo Open Dataset mAPH= 79.5.
[347]	LC	DeepInteraction maintains separate modality representations and utilizes dedicated interaction encoders and decoders for 3D detection.	Preserving modality-specific information, allowing each sensor's unique characteristics to enhance performance.	nuScenes dataset mAP= 70.8, NDS = 73.4
[339]	LC	CMT is an end-to-end 3D detector that outputs 3D bounding boxes from image and point cloud tokens without explicit view transformation.	High performance on nuScenes and faster inference speed.	nuScenes dataset mAP= 72.0 NDS = 74.1, Argoverse2 dataset AP= 36.1, CDS= 27.8
[337]	LC	FusionPainting integrates 2D RGB and 3D LiDAR data through semantic segmentation and adaptive fusion to enhance detection accuracy.	Improves 3D detection performance by incorporating both 2D and 3D semantic information.	nuScenes dataset mAP= 68.1 NDS = 71.6.
[254]	LR	MVDNet is a two-stage deep fusion network for vehicle detection using deep fusion to integrate LiDAR and Radar data.	Outperforms state-of-the-art methods, particularly in adverse weather, and integrates complementary sensor data effectively.	Oxford RobotCar dataset mAP_{clear} = 84.78, mAP_{foggy} = 80.29
[185]	RC	HVDetFusion is a multi-modal 3D detection algorithm that integrates camera and Radar data, optimizing the Camera-based Bevdet4D framework to reduce false positives.	Effective integration of Radar data to reduce false positives and improve accuracy, offering camera-only and camera-Radar inputs.	nuScenes dataset mAP= 60.9 NDS = 67.4.
[170]	RC	Combines Radar and camera data at feature and instance levels, using a radar-guided BEV encoder to enhance detection accuracy and reduce errors.	Effective at both feature and instance levels, leading to improved 3D object detection performance.	nuScenes dataset mAP=50.6, NDS= 58.7.
[328]	RC	HyDRa enhances 3D perception using hybrid fusion and dense BEV representations for improved depth prediction and object detection.	Improves depth prediction and object detection, enhanced occupancy mapping to outperform camera-only methods.	nuScenes dataset mAP= 57.4, NDS= 64.2, Occ3D benchmark mIoU=44.4.
[123]	LRC	DeepFusion integrates lidar, camera, and Radar data with modular feature extractors and a BEV representation for enhanced 3D detection.	Flexible integration of different sensors; effective long-range detection; robust in adverse conditions.	nuScenes dataset AP_{car} = 81.6, $AP_{pedestrian}$ = 84.6
[257]	LRC	CLR-BNN use a Bayesian approach for multi-object detection, leveraging sensor fusion to improve accuracy and handle uncertainties.	Outperforms deterministic models in accuracy; better handling of uncertainties; reliable across diverse conditions.	nuScenes dataset mAP= 76.5, night mAP= 75.0, rain mAP= 73.0

Table 2.7: This table provides an overview of the segmentation tasks, respective challenges, and their applications.

Approach	Main Idea	Challenges	Application
Semantic Segmentation	Classification of each pixel in an input image and assigning predefined category labels to them, without distinguishment among different instances.	Class imbalance and impact of the dominant class, boundary precision, proper understanding of the scene context, sensitivity to illumination and weather conditions.	Road scene understanding, drivable area detection, object detection enhancement, and assisting in path planning.
Instance Segmentation	Classification of each pixel in an input image and assigning object-level labels to them, with distinguishing among different object instances.	Computational complexity, object boundary precision, annotation and training data availability, handling complex scenes with large number of objects, sensitivity to illumination and weather conditions	Assistance to object detection, scene understanding, path planning, collision avoidance.
Panoptic Segmentation	Classification of each pixel in an input image and assigning both class-level and object-level labels to them, with distinguishing among different object categories and instances.	Computational complexity, model complexity, annotation availability and dataset quality, class-level and object-level boundary precision, small object segmentation, sensitivity to illumination and weather conditions.	Scene understanding at a fine-grain level, assistance to object detection and tracking, and assistance with path planning.

Semantic Segmentation

Semantic segmentation involves assigning pixel-level labels to partition input images into distinct regions. Among deep learning methods, Fully convolutional networks (FCN) were primarily used for image partitioning to different regions at the pixel level [125, 141, 208, 227]. In an early study, multi-scale convolutional layers were utilized to encode texture and shape information into the feature space that would be used to dictate different regions [131]. This method is enhanced with a graph-based classification algorithm to refine the boundaries of the detected regions in the output results. In another study, the authors adopt an FCN scheme for semantic-level scene understanding [208]. First, a fully convolutional classifier is used to produce a coarse representation from the input image. Next, a bank of deconvolution filter parameters is learned through training which enables the model to upsample the coarse output of the classifier and refine boundaries at the pixel level.

Further advancement in FCN architectures for scene understanding evolved the models to adopt an encoder-decoder scheme. For instance in SegNet [22], the encoder gradually reduces the spatial resolution of the input image which is then upsampled step-by-step in the decoder module for the final classified maps. The key idea of this architecture is a guided upsampling process in the decoder by using the pooling indices from the corresponding encoder. Furthermore, some researchers explored context-aware methods for segmentation [93, 192, 199, 370]. For instance, DeepLab [93] is a framework that utilizes Atrous convolutions for upsampling which provides the model with a degree of freedom on feature resolution. This results in enlarging the receptive field that accounts for a larger contextual area of the input. These are accompanied by atrous spatial pyramid pooling (ASPP) that samples the input by different rates which enables the model to gain understanding at different levels.

The evolution of LiDAR technology has led to the development of low-cost sensors with higher resolution and a longer range of detection. Coupled with the availability of relevant outdoor datasets, LiDARs have emerged as the second most common modality for segmentation tasks [83, 162, 330, 376]. In an early work, researchers integrated depth information from LiDAR with images through projection [308]. Furthermore, LiLaNet is a framework proposed to address the high demands of 3D convolution layers [265, 305] in 3D semantic labeling by replacing them with 2D layers and making configuration to the input frame [239]. For the data, camera images and LiDAR point clouds are combined to build LiDAR up-front depth-informed images. Then, the model builds up on 2D convolutional layers to parse the input at the semantic level. One recent study approaches 3D semantic segmentation with semi-supervised learning [218]. similar to the previous method, they extract up-right depth images from LiDAR input which is used to generate constraints as well as the annotation portion. Finally, a CNN classifier is used for the learning process. Moreover, some researchers explored the benefits of the Bayesian Filtering algorithm for a more refined point-wise labeled map [120]. They use an encoder-decoder to build their model to first extract features and use them to build semantic maps. Then, they utilize a Bayesian filtering approach to leverage the temporal connection of the output scans to refine the classified regions and improve final results.

Multi-modal algorithms are among the powerful solutions that can be beneficial in enhancing semantic segmentation by combining information from different data streams [136, 302, 376]. Sun et. al [302] proposes a multi-modal model that fuses RGB data with thermal information gathered by an infrared camera to assign semantic labels to the confronted scene. ResNet is adopted to perform feature extraction on both RGB and Thermal inputs and a hand-crafted decoder is used to generate semantic masks. In another study, the deficiency of LiDAR-only methods for distant or small object segmentation is addressed by fusing LiDAR point clouds with images acquired from different views [188]. In this approach, separate features are extracted which are later fused to achieve geometric-aware dual-modal feature maps. Then, these maps accompanied by semantic encoded embeddings of LiDAR and camera streams are consumed by a semantic-based fusion module that is used by a segmentation head for final region pre-

dictions. Even though multi-modal models have been beneficial in certain scenarios, they remain relatively underexplored compared to object detection approaches. This signals a rich area that requires further exploration and development. Table 2.8 provides a critical overview of the semantic segmentation methods.

Table 2.8: An overview of the attributes and strengths of semantic segmentation methods with relevant performance metrics reported in percentages.

Ref	Sensor	Main Idea	Strengths	Performance
[22]	C	SegNet is a deep CNN designed for efficient and practical semantic segmentation with a balance between memory and computational efficiency.	Efficient upsampling through pooling indices reduces computational load. Light-weighted suitable for real-time autonomous driving applications.	CamVid Road dataset mIoU=60.4
[93]	C	DeepLab enhance segmentation using atrous convolution and ASPP, combining DCNNs with CRFs for detailed and accurate results.	Captures multi-scale features for improved accuracy, CRFs enhance boundary precision along object edges.	Cityscapes dataset mIoU= 70.4
[302]	C	A deep network that fuses RGB and thermal images to enhance segmentation in challenging lighting conditions.	he fusion of RGB and thermal data ensures accurate segmentation in various lighting conditions, and robustness during nighttime scenarios.	MFNet dataset mIoU= 53.2, mAcc= 63.1
[136]	L	Presents a method for training 3D segmentation models using 2D image datasets with semantic labels, generating pseudo-labels for 3D data.	Improves 3D segmentation accuracy significantly using pseudo-labels. Robust performance across diverse urban scenes and datasets.	nuScenes dataset mIoU= 80.0
[162]	L	A 3D segmentation framework modeling local dependencies with a local dependency module combining slice pooling and RNN layers.	Surpasses state-of-the-art methods on multiple benchmarks. Achieves superior performance with reduced inference time.	ScanNet dataset mIoU= 39.35, S3DIS dataset mIoU= 51.93
[239]	L	LiLaNet is designed for point-wise semantic labeling of LiDAR data, using virtual image projections and automated data generation for efficiency.	Significantly outperforms existing CNNs with high efficiency. Automated data generation boosts performance with minimal manual effort.	LiLaNet dataset mIoU= 69.7.
[188]	LC	MSeg3D leverages both LiDAR and multi-camera sensors for 3D semantic segmentation, addressing challenges in combining data from different modalities.	Achieves state-of-the-art results on various datasets. Demonstrates robustness and enhanced data augmentation.	SemanticKITTI mIoU= 66.7, nuScenes mIoU= 81.14, Waymo Open Dataset mIoU= 70.51.
[376]	LC	Introduces PMF for 3D LiDAR semantic segmentation, combining RGB and LiDAR data to enhance scene understanding.	Effective fusion of RGB and LiDAR improves accuracy and robustness. Surpasses existing methods with better mIoU scores.	nuScenes dataset mIoU= 76.9, SemanticKITTI mIoU= 63.9.
[154]	LRC	Introduces a unified top-down representation for multi-modal perception, integrating lidar, Radar, and camera data for real-time segmentation and prediction.	Simplifies fusion with a single grid; adaptable to various sensors; extends to future motion prediction.	nuScenes dataset $IoU_{pedestrian}$ = 24.9, $IoU_{vehicle}$ = 44.3, $IoU_{background}$ = 49.7

Instance Segmentation

Instance segmentation is a challenging task that goes beyond semantic segmentation by not only labeling pixels with object categories but also distinguishing between different instances of objects belonging to the same category. Previous studies have explored various approaches to tackle this task, taking inspiration from different fields. For example, some studies, [269] and [320], exploited recurrent networks in their architectures, and some others, [369] and [309], used deep networks as mapper functions to acquire a representation of the input frames and divide the instances later in the post-processing step.

In the context of 2D instance segmentation, typically, two-stage models are used which at first detect the whereabouts of the object instances and then perform the segmentation task [146, 194, 322, 364]. For instance, a framework was proposed that builds on an early object detector to identify object instances in the input frame [91]. The localized objects are then distinguished from the background and the region they occupy is associated with instance-level labels. *Brabandere at al.* [113] took a simple approach to distinguish among different instances of vehicles on the road. First, they use a semantic segmenter network as a function to map the input images to a feature space which enables them to perform clustering on the objects. Consequently, with the use of a handcrafted loss function, intuitively, points belonging to the same cluster are pushed towards each other whereas points from different clusters are pulled farther. In another study, the authors [156] build their model to leverage the advantages of the classical variation models [87] instead of only using a Mask RCNN [150] to avoid low-resolution masks and generate more accurate boundaries. However, a shortcoming of variation models is their dependency on hyperparameters and the first initialization values. Consequently, the authors propose their end-to-end model, *LevelSet R-CNN*, which combines the two methods to adjust the hyperparameters and define the energy function with respect to the detected instances for the optimization phase. Furthermore, a team of researchers explored the benefits of recurrent learning methods in this field [260]. The framework builds on LSTM blocks and uses an attention mechanism to detect and segment objects sequentially.

There have been numerous studies to address scene understanding at the instance level in 3D [94, 159, 201, 291]. For instance, in a study, a mono-ocular setup has been used for this task [94]. In their pipeline, object instances are differentiated from the background which in turn are used as 3D RoIs. Then, several features with respect to shape and location are learned to complete the final labeling process. Meanwhile, LiDARs are preferred over cameras due to their inherent 3D properties [108, 228, 240, 321]. SqeeuzeSeg is a two-stage model, that initially uses an encoder-decoder module to generate voxel-level instance labels [330]. Subsequently, a conditional random field (CRF) module, re-implemented in a recurrent setting, is applied to the output of the first stage to refine the final label maps. In a recent study, a two-staged model is proposed to jointly perform detection and instance segmentation [371]. The first stage utilizes an encoder-decoder architecture to extract features and a novel spatial embedding which are used

to predict early instance-level segments. These early detections are taken as RoIs which are further combined with the extracted features to regress final bounding boxes and instance labels. Moreover, researchers highlight the issues of inadequate resolution in feature maps extracted from 3D point clouds and the negative impact it leaves on the segmentation of small or distant objects [360]. Consequently, they build a high-resolution feature map in BEV view by leveraging KNN in point cloud encoding and self-attention mechanism. The results are consumed by a backbone network to estimate the center and height bounds for each foreground voxel. The estimations are used as constraints to refine the output labeled map. Table 2.9 provides a critical overview of the instance segmentation methods.

Table 2.9: An overview of the attributes and strengths of instance segmentation methods with relevant performance metrics reported in percentages.

Ref	Sensor	Main Idea	Strengths	Performance
[320]	C	A recurrent U-Net architecture that enhances performance while maintaining the original model's compactness.	Light-weighted framework, demonstrating real-time and superior performance road segmentation.	Cityscapes validation set mIoU= 62.7.
[322]	C	Introduces SOLO, an instance segmentation framework that maps input images to object categories and instance masks.	Simplifies segmentation by eliminating post-processing steps. Extends effectively to tasks like instance-level image matting.	Cityscape validation set mIoU= 38.0.
[194]	C	PolyTransform integrates traditional segmentation with polygon-based approaches, generating masks and refining them into polygons for better object boundary alignment.	Produces accurate, geometry-preserving masks and achieved 1st place on the Cityscapes leaderboard with improvements in boundary metrics.	Cityscape test set prediction mask AP= 40.1.
[364]	C	A contour-based method that enhances performance through learnable contour initialization and dynamic matching of vertex pairs.	Improves contour quality and achieves top results on multiple datasets with real-time efficiency (36 fps for 512×512 images).	Cityscape dataset prediction mask AP= 32.9.
[156]	C	Enhances performance by integrating the robust feature extraction from Mask R-CNN with the precision of a variational segmentation framework.	Improves mask boundaries and unifies variational segmentation to enhance robustness and accuracy.	Cityscape dataset prediction mask AP= 40.0.
[321]	L	SGPN is a framework for 3D instance segmentation on point clouds, using a similarity matrix for point grouping and semantic classification.	Utilizes an intuitive similarity matrix, can incorporate 2D CNN features, and achieves strong performance in 3D segmentation.	S3DIS datset prediction mask AP= 54.35.
[330]	L	An end-to-end pipeline for semantic segmentation of road objects from 3D LiDAR point clouds, using CNNs and CRF for classification and refinement.	Achieves high accuracy and speed, and effective data augmentation from synthetic data improves performance.	KITTI dataset mIoU= 33.64.
[371]	L	Introduces a unified framework for 3D object detection and instance segmentation using spatial embeddings to group points by object centers and avoid non-maximum suppression.	Integrates detection and segmentation processes, enhances performance by clustering foreground points, and achieves real-time performance with fewer region proposals.	KITTI prediction mask AP= 47.15, KITTI object detection $AP_{70,moderate}$ = 78.96.
[360]	L	A robust instance segmentation baseline in large-scale outdoor LiDAR point clouds, using dense feature encoding and addressing class imbalances.	Enhances performance for small and distant objects, manages class imbalances effectively, and introduces a large-scale dataset.	Their own LiDAR dataset prediction mask AP= 62.4.

Panoptic Segmentation

Panoptic Segmentation is a relatively new area of research that incorporates both semantic and instance-level information. This task was first introduced in 2019 and the respective metrics were formulated by *Kirillov et al.* [175]. As a part of their work, some relative experiments were conducted on public benchmarks to assess and compare human-level and machine-level performance which gave some interesting insight into the manifestation of this task.

Early attempts, such as [18, 91, 238], focused on combining the two tasks in one model which would output both semantic and instance-level labels simultaneously. As an example, *MaskLab* [91] was designed to give object detection as well as segmentation outputs. More specifically, object detection is done utilizing a Faster R-CNN backbone to generate bounding boxes. Guided by these detections, regions of interest are parsed semantically and are combined with predictions to differentiate among instances of the same semantic labels. However, the early models were mostly performing the two previous tasks separately, without much-shared computation.

To fill this gap, the *Panoptic FPN* [174] was proposed in 2019 to combine the two tasks in a unified architectural manner, all in one single network. The proposed model, firstly, adopts Mask R-CNN with FPN to perform instance segmentation tasks. Next, a semantic segmentation branch consumes the FPN feature maps to generate output at the semantic level. The final output of the network is a refinement of both instance and semantic labels which is done in a post-processing step with a similar mechanism to non-maximum suppression.

Since the introduction of panoptic segmentation, various architectures have been developed to combine multiple techniques and concepts, aiming to enhance performance and address a range of challenges [135, 165, 190, 193, 241, 297, 334]. In one study, *Chen et al.* [92] points out the challenge of data annotation requirement of segmentation tasks and exploits the advantages of semi-supervised learning to train their model, *Naive-Student*, on unlabeled videos from urban environments to perform scene parsing at the panoptic level. While some studies prioritize the accuracy and the quality of their models, some [114, 157, 160, 236] take on the challenge of optimizing runtime performance as one of the most important practical aspects. For example, a recent novel research led to the design of the MGNet framework [275] which jointly performs panoptic scene parsing and depth estimation using a single camera as their perceiver sensor. The input of the model is encoded using a Global Context Module (GCM) and the outputs are fed to separate decoder heads, each of which is customized to render the semantic, instance, and depth-related logits. A post-processing step projects the panoptic-level information onto a final 3D point cloud. MGNet is a lightweight framework tailored for real-time performance. When the runtime-related issues are addressed, panoptic models can be put into practice. As an example, *GenPa-SLAM* [28] is a real-time LiDAR-based simultaneous localization and mapping (SLAM) framework that exploits panoptic-level information from the surrounding environment. The front end of the model performs a

semantic-aware object detection to provide prior knowledge for a guided extraction of geometric landmarks which is, next, used in the back end to optimize vehicle trajectory and landmark positions.

These advancements in segmentation techniques, including semantic, instance, and panoptic segmentation, collectively enhance the ego-vehicle's ability to accurately and efficiently interpret complex scenes which is the requirement of decision-making procedure and safe navigation. Compared to object detection, segmentation tasks are less mature but rapidly evolving. While object detection has a longer history with well-established models and extensive datasets, segmentation methods are gaining popularity due to their ability to provide more detailed scene understanding. Despite these challenges, the progress in segmentation is promising, and ongoing research continues to bridge the gap between these two critical areas of computer vision. Table 2.10 provides a critical overview of the instance segmentation methods.

Table 2.10: An overview of the attributes and strengths of panoptic segmentation methods with relevant performance metrics reported in percentages.

Ref	Sensor	Main Idea	Strengths	Performance
[334]	C	UPSNet combines semantic and instance segmentation using a novel panoptic head for efficient and accurate segmentation.	Efficiently integrates semantic and instance segmentation tasks, achieving top results with faster inference.	Coco dataset PQ= 46.6, Cityscape dataset PQ= 61.8.
[190]	C	Panoptic FCN unifies object and background segmentation with a fully convolutional approach, generating unique kernel weights for each instance or category.	Outperforms previous models with high efficiency and ensures accurate object and background segmentation.	Coco validation set PQ= 44.3, Cityscape validation set PQ= 61.4.
[165]	C	Introduces panoramic panoptic segmentation combining panoramic FoV with panoptic image-level understanding, adapting standard models to panoramic images.	Holistic scene understanding with improved performance in complex environments and efficient domain adaptation.	Cityscape dataset PQ= 31.9, Panoramic Annular Semantic Segmentation dataset mIoU=55.0
[193]	C	Panoptic SegFormer uses transformers with a deeply-supervised mask decoder and enhanced post-processing for improved accuracy and efficiency.	Reduces training epochs while increasing accuracy and shows stronger zero-shot robustness.	Coco test-dev set PQ= 56.2, COCO-C various environmental situation PQ= 47.2.
[135]	C	Introduces a framework for depth-aware panoptic segmentation combining depth prediction with instance-level semantics using dynamic convolution.	Enhances depth accuracy and segmentation quality with instance-specific kernels and innovative depth loss.	Cityscape validation set pQ= 64.1, Depth-aware Panoptic Quality (DPQ)= 60.4.
[160]	C	Presents a single-shot panoptic segmentation network for real-time operation with dense detections and global self-attention, achieving high efficiency.	Achieves real-time performance with reduced complexity and competitive accuracy compared to state-of-the-art methods.	Cityscape validation set PQ= 58.8, COCO validatoin set PQ= 37.1.
[275]	C	MGNet combines panoptic segmentation and self-supervised depth estimation into a single real-time model, providing dense 3D point clouds.	High frame rates (44.4 ms) and efficient architecture for real-time applications, showing competitive performance.	Cityscape validation set PQ= 55.7.
[236]	C	A single-stage network for real-time applications, integrating object detection, semantic segmentation, and instance-level attention masks.	Achieves over 10 FPS on high-resolution images, suitable for real-time applications, with a simplified pipeline.	Cityscape validation set PQ= 59.7, mIoU= 76.0, COCO validation set PQ= 34.4.
[92]	C	The Naive-Student model uses semi-supervised learning, where pseudo-labels from unlabeled videos and images are combined with human-annotated data.	Effectively leverages large amounts of unlabeled data, reducing reliance on expensive human annotations.	Cityscape test set PQ= 67.8, mIoU= 85.2.

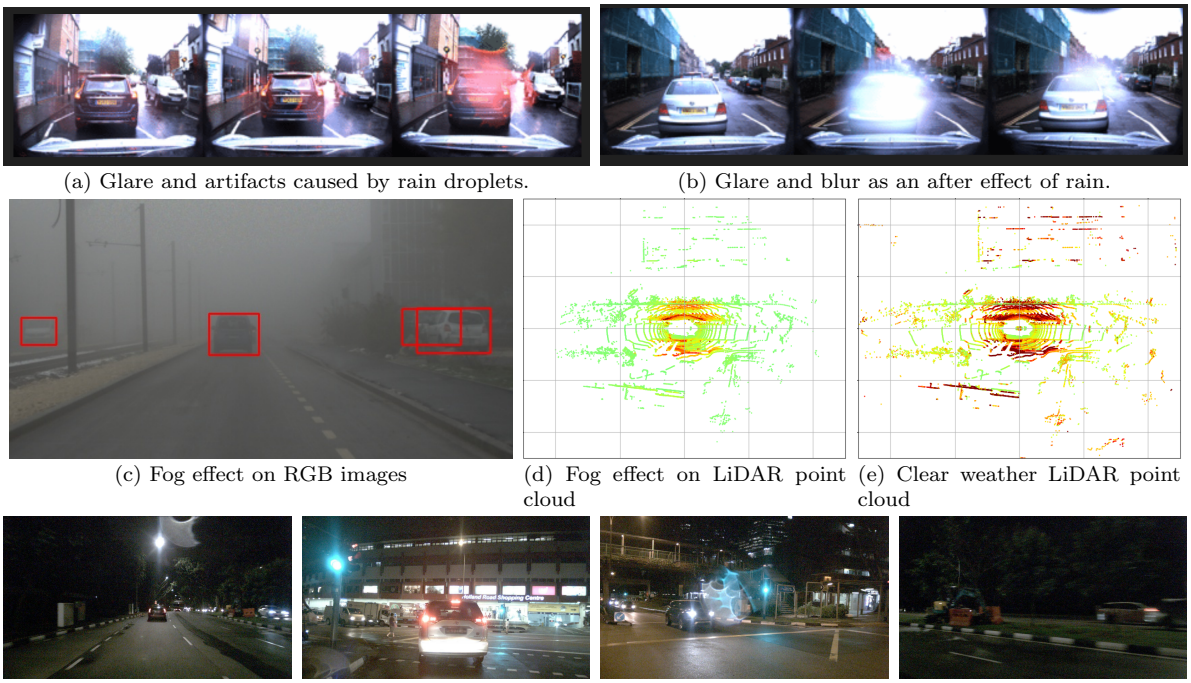
2.6 Perception System Challenges

Typically, perception models are developed under the assumptions of ideal conditions such as normal weather conditions, adequate illumination, and a clean, continuous data stream from sensors. However, these assumptions rarely hold for realistic situations, where varying weather conditions and different times of the day can significantly impact

sensor performance and data quality. In this section, we explore two challenging situations, *Adverse Weather Conditions* and *Times of Day*. We aim to review the challenges and investigate the solutions proposed to address them.

2.6.1 Adverse Weather Condition

One of the most important aspects of a perception system is its robustness against various weather conditions. Different weather conditions leave a specific impact on each type of sensor based on their operating principles. Fig 2.4 illustrates examples of the distortions caused by various conditions.



(a) Glare and artifacts caused by rain droplets.

(b) Glare and blur as an after effect of rain.

(c) Fog effect on RGB images

(d) Fog effect on LiDAR point cloud

(e) Clear weather LiDAR point cloud

(f) Glare effect caused by artificial lighting sources.

(g) Glare effect caused by street and vehicle lights.

(h) Glare caused by streetlights and headlights.

(i) High noise level in low-light situation.

Figure 2.4: Effects of weather and lighting on sensors: (a-b) Rain causing glare and blur in the camera images from Oxford RobotCar dataset [26]; (c-e) Fog reducing visibility in images from DENSE dataset [32] and LiDAR point cloud from nuScenes dataset [80]; (f-i) Nighttime challenges with glare and noise affecting camera performance from nuScenes dataset [80].

Overview of Challenges

Precipitation is a common weather condition where water droplets or frozen particles fall to the ground. During precipitation, several factors can significantly affect sensor performance. According to a NASA report [4], raindrop sizes range from 0.1 to 4 mm.

Furthermore, images are formed by assigning pixel intensity values based on the brightness received from the scene. For LiDAR sensors, the location of targets is determined by measuring the time it takes for the laser pulse to return. As discussed in section 2.3, visible light has wavelengths between 380 and 700 nm, while LiDAR sensors use laser beams with wavelengths of 905 and 1550 nm.

The combination of these factors leads to distortion and degradation in the measurement quality of both the camera and LiDAR sensors. Mie scattering, which affects electromagnetic waves with wavelengths shorter than the precipitation particles, results in two main effects. First, there is attenuation of power for both visible light and laser beams due to energy absorption by the particles. This attenuation reduces brightness and visibility for these sensors. For instance, a study [310] reports that the LiDAR sensing range can decrease by 40% during light rain and up to 80% during heavy rain compared to clear weather conditions. Second, precipitation particles cause refraction of light or laser beams, altering their direction and leading to distortion and visual artifacts.

Moreover, reflection introduces additional distortion in camera and LiDAR measurements. Water droplets and frozen particles reflect light beams, which can create glare in images and lead to false detections for the LiDAR sensor, mostly for droplets close to the sensor. These destructive impacts on camera images become more severe at night when the sun, the primary source of daylight, is absent. During nighttime, the reduced ambient light makes the distortion and glare caused by reflections from street lights and other vehicles more significant [78].

Fog is another weather condition that creates a cloud-like ambient at lower altitudes. Similar to precipitation, fog particles are larger than visible light and laser beams and result in Mie scattering phenomenon. Due to significant absorption by fog particles, visibility is reduced. Some studies report up to 80% limitation of detection range for laser sensors [310], and up to 50% reduction of visibility of camera images [358]. In addition to reduced visibility, false detection in LiDAR measurements and blurry images are among the common results. In a spectrum analysis on normal and foggy images [358], it is observed that while during normal weather, images contain spread spectral components which indicate the presence of object edges, during foggy weather, most of the spectral components fall near lower band of frequencies, indicating loss of edge information and the presence haze effect on the images. During this situation, distinguishing different objects becomes greatly challenging.

Under these conditions, Radar sensors suffer from performance challenges, however, due to their principle differences in operating mechanism, they show greater robustness compared to camera and LiDAR sensors. Using radio waves with a wavelength in the millimeter range, these sensors are less affected by signal attenuation and scattering effects [358]. Nevertheless, signal attenuation at larger distances becomes considerable, which limits the visibility of these sensors, and backscattering caused by water or fog particles adds noise and false detection to the measurements [310, 367]. Conclusively, these challenges create faulty data that must be addressed to ensure reliable perception systems. A comparative summary considering key elements of the camera, LiDAR, and

Radar sensors is provided in Fig. 2.2.

Research to tackle these challenges can be broadly categorized into two groups. The first category mostly focuses on increasing the robustness and enhancing the performance of each sensor, particularly cameras and LiDARs. This involves developing algorithms to mitigate the effects of adverse weather conditions on sensor data. The second category is the methods that leverage fusion techniques to take advantage of the capabilities offered by different modalities as well as the redundancy in the data. These approaches aim to produce more accurate and reliable perception outcomes even under challenging conditions. A summary of the core concepts, advantages, and challenges of these two categories of approaches is given in Table 2.11.

Table 2.11: Overview of the approaches proposed to address challenges occurring due to adverse weather conditions. This table summarizes the main idea of each category of approaches accompanied by their advantages and challenges.

Approach	Main Idea	Advantages	Challenges
Task-specific Enhancement	Enhancing the quality of the input data during a preprocessing stage by focusing on mitigating noise and correcting distorted information. This approach, typically, estimates the key features of the source of noise or distortion (e.g. medium transmission) and proposes solutions to mitigate them.	Boosting object detection performance, availability of scenario-specific modeling (fog, snow, rain) and solutions, reducing computational complexity for the subsequent perception stages, cost-efficiency due to not requiring additional sensors.	Scenario dependency and lack of generalization for different situations, sensor type dependency, real-time performance, potential for over-processing resulting in synthesized artifacts, natural reconstruction of the input frame, LiDAR low density in far range.
Multi-modal Fusion	Integrating information from different sensing modalities creates a more comprehensive, detailed, and accurate representation of the driving scenario. This approach leverages the strength of each sensor modality as well as the redundancy in the data to compensate for noise, distortion, and inherent sensor shortcomings.	Rich and comprehensive representation of the environment, improving detection accuracy and reliability, introducing robustness against sensor failure, enhancing robustness against challenging situations such as adverse weather conditions, compensating for individual sensor weaknesses.	Model complexity, computational processing load, requiring further research for choosing a proper and efficient fusion method, different data resolution, data compatibility, inter-sensor calibration and synchronization.

Task-specific Enhancements

To improve and robustify the performance of perception systems, researchers have developed algorithms that address the challenges posed by adverse weather conditions. These

algorithms often function as pre-processing blocks which enhances the quality of the input data frame. The proposed solutions vary depending on the type of input, primarily focusing on images or point clouds.

For image data, a lightweight model called ReViewNet [252] was designed in an end-to-end manner to perform image dehazing. The algorithm makes use of quadruple color space in a double-look encoder-decoder architecture with different loss weights to produce haze-free images. AODNet [186] is another image-dehazing model adopting a two-stage architecture that introduces a novel formulation for the atmospheric scattering model. The algorithm minimizes the reconstruction error by estimating the merge parameter *K(x)* by a convolutional network, the output of which is used in the second stage to cancel the haze effect from images. They also cascaded their model with RCNN to study the impact of their model on the robustness and accuracy of object detection. A recent study leveraged the attention mechanism and proposed a model to improve the quality of recorded visual information [202]. There are three different stages to their model. Before the dehazing process begins, a multi-resolution attention layer is applied to the input image to estimate the non-homogeneous distribution of the haze. This is followed by a feature extraction module which consists of an encoder-decoder and a dense network in parallel. The encoder-decoder is used to extract the structure and texture-related features of the haze. On the other hand, the dense network investigates the frequency domain in search of complementary features. In the end, a residual network scheme fuses the input from the attention and feature extraction modules to give a clear final image. The model's effectiveness was tested on three object detectors i.e. YOLOv4-Tiny [166], Faster RCNN [261] and YOLOv4 [37], demonstrating enhancement in robustness and accuracy.

Many of the methods use add synthetic haze and base their experiments and analysis on synthesized data [81,90,119,345,362]. One study [361] claims that solving the problem with synthetic data does not necessarily solve the issue with inputs from real-world scenarios. Consequently, the authors put forward a 4-stage architecture that incorporates semantic prior information to better clean natural haze. More specifically, the goal is to maximize the color of each pixel with respect to its semantic information constraint. A residual dilated network is in charge of extracting structural features which are then fused adaptively to produce the final haze-free image with the unnatural artifacts removed.

Many studies have focused on the characterization of LiDAR sensors [242, 329, 348], analyzing the effect of adverse weather conditions on these sensors, and exploring different mitigation strategies [34, 148, 237, 256, 270]. Some researchers have shown that normal rainy weather does not corrupt LiDAR data to a considerable extent [134]. However, when the precipitation rate rises to high levels, the impact becomes more similar to fog which creates false obstacles and lower detection range. Additionally, snowflakes are solid objects and due to freezing, they can accumulate and become larger which again results in a blocked line of sight and creates the haze effect [367].

One study addresses the challenges of AV under snowy weather by enhancing LiDAR image quality and improving vehicle localization [253]. The researchers increased LiDAR

image density using an accumulation strategy and improved texture quality using Principal Component Analysis to reconstruct LiDAR images from maps. They also extracted and matched edge profiles from LiDAR and map images to better detect lane lines and roadside edges, resulting in more robust and accurate localization in adverse conditions. Moreover, another work thoroughly analyzes the impact of fog and rain on LiDAR point clouds and perception systems [153]. To that end, they set up a reproducible experiment in a climate chamber in which controlled production of fog and rain with arbitrary parameters is possible. Additionally, k-nearest Neighbor (kNN) and Support Vector Machine (SVM) algorithms were utilized to classify weather among 'clear', 'fog', or 'rain' categories based on the features computed from the statistics of the point clouds.

One study identified a gap in the field of de-noising [89, 279] LiDAR measurements which are affected by snow and rain [89]. The authors first investigate the performance of the existing 2D and 3D filtering methods prior to their work and highlight their shortcomings. Consequently, they present an improved version of a Radius Outlier Removal filter to clean the point cloud from the snow effect while preserving the contextual features. Finally, a comparison is drawn between the proposed method and others based on manually labeled data collected on the public roads of the Waterloo area. A later study [152] presents a deep learning approach based on CNNs in the direction of noise removal of LiDAR point cloud. The authors propose WeatherNet for the purpose of semantic weather segmentation. The model is built upon several modified LiLaBlocks which include convolution and dilated convolution layers.

Fusion Models

As opposed to many uni-modal solutions that are used for fighting off the distortion in data occurring due to unfavorable weather conditions, the fusion models exploit the existing data redundancy across different modalities to boost performance and robustness. These models are often proposed in an end-to-end manner, leverage data fusion algorithms, and use the result to generate predictions. Various combinations of sensors are used by researchers, with the camera and LiDAR pair being the most common [95, 179, 232, 288].

Cameras are able to capture fine details from the surrounding environment but are affected by different illumination conditions and often struggle in distant-object detection tasks. On the other hand, LiDARs are unaffected by changes in illumination, have a longer detection range, and provide the 3D shape of objects after measurement but their point clouds are often sparse. Consequently, methods that combine camera and LiDAR data [23, 82, 207, 211, 245, 292, 335] can benefit from the depth and 3D contextual information of LiDAR point clouds and the high spatial resolution of camera images, to compensate for each other's shortcomings and build a more robust perception system. For example, one study utilized mean operation to fuse features extracted from RGB images from multiple views and LiDAR point clouds [180]. Initially, feature extraction for each sensor branch is performed individually. Then, the acquired maps are cropped

and configured to achieve coherent dimensions and consequently, are fused. The results are used to generate RoI proposals needed for making final predictions. In another study, an attention mechanism is utilized for data fusion [338]. First, the camera images and LiDAR point cloud pass through 2D and 3D semantic segmentor modules. Then, an attention block to enrich the point cloud with camera information based on the extracted semantic labeled maps. At the end, a 3D detector head produces final predictions.

The shortcomings of cameras and LiDARs are revealed in snowy, rainy, or foggy conditions. Due to their operating principles, the recorded data often experience distortion and attenuation, thus the measurements become noisy and unreliable. In these situations, Radar sensors have shown greater robustness compared to cameras and LiDARs, leading to researchers exploring their capabilities to tackle these challenges.

In search of Radar-based solutions for object detection under adverse weather, researchers have primarily focused on Camera-Radar combinations [172, 182, 221, 223, 233]. In a recent study, a multi-modal framework is proposed that mostly relies on multi-view camera images for detection and incorporates Radar detection points and velocity estimations to improve object location predictions [185]. Another innovative approach is HyDRa, a novel camera-radar fusion architecture designed to enhance depth prediction accuracy and achieve state-of-the-art performance [328]. Hydra begins by encoding multi-view camera images and voxelized radar point clouds into high-level feature maps using a 2D encoder and point pillars, respectively. A transformer is used to associate radar features, which lack height but have sparse depth, with camera features, which lack depth but have dense height information, creating an enriched representation. These features are then fused into a BEV representation with a forward projection module and refined by depth information from Radar recordings. Finally, the processed BEV is fed into task-specific heads for final predictions.

Even though LiDAR-Radar-based solutions have not been extensively explored, promising results have emerged from recent studies [205, 254, 350]. One such study introduces a two-staged model called MVDNet, proposed in an end-to-end manner which fuses LiDAR point clouds with Radar range-azimuth detection [254]. Similar to common pipelines, a network is used to extract and produce feature maps from Radar and LiDAR data streams separately. These maps are concatenated and further processed to generate RoI proposals which guide the sensor fusion process using self and cross-attention mechanisms. A novel aspect of this work is the use of past frames as historical data through a temporal fusion block. This block enables data-sharing along the time axis and improves the final inference of the location of the detection. In a recent work, the authors choose mm-Wave Radar as the main and camera as the auxiliary modality [205]. They use a joint data probabilistic data-association (JPDA) fusion method for data integration. Moreover, in a comprehensive study on fog effect on different sensors, researchers based their fusion algorithm on entropy measured from each sensor recording [32]. In their work, RGB and gated cameras, LiDAR, and Radar are the four components of the sensor suite that are utilized in a deep fusion architecture, and at the end, an SSD block consumes the fused feature map for final predictions.

2.6.2 Night Time

Achieving the highest level of autonomous driving requires safe performance across a variety of environmental situations, including driving during nighttime. The transition from day to night presents a significant challenge due to inadequate natural illumination, creating inherent differences between daytime and nighttime domains.

One effective approach to address this problem is leveraging domain-adaption approaches. Specifically, in this approach, the objective is to use the models that are performing well in the daytime and adapt them to nighttime situations without any nighttime-related labels. For instance, *Dai et al.* proposes the Dark Model Adaptation method to perform semantic segmentation at nighttime. In their model, they use a three-step bridge called the twilight time bridge to progressively transfer the knowledge learned from the daytime to the nighttime. To this end, they build their own dataset of images ranging from daytime to twilight time. Similarly, other approaches exist in which twilight bridging is utilized [272], [273]. The downside of these methods is the cost of additional and complex training.

Furthermore, some studies utilize neural style transfer to translate daytime images to match the nighttime style and properties [268, 298, 331]. *Wang et al.* [318] aims to tackle the shortage of nighttime labeled data in their work and build two datasets, *Synthesiscarla* and *Cyclecity*, by synthesizing data with the addition of style-transferred images from daytime. They also propose a two-stage end-to-end model to compensate for the effects of inadequate illumination and perform segmentation on enhanced images.

In addition to domain adaptation, fusion-model capabilities offer another avenue for addressing nighttime challenges. As an example, infrared detection is a sensing modality that keeps the outline and form of objects at the cost of losing color information. However, improvement is observed when fused with RGB images. In one study [303], the authors choose RGB as well as three types of infrared cameras to collect multi-spectral information from the surroundings. Each sensor modality performs object detection individually, single-spectrum detection, and an ensemble scheme is used to aggregate the final results of each branch. The authors also built a multi-spectral dataset as none was available at their time of work. Some relevant datasets can be found in [164], [307], [336] and Table 2.12. Other sensor configurations are used in the literature to accommodate the needs of perception [30]. For instance, LiDARs are robust sensors against poor illumination conditions and some studies have exploited advantages that LiDARs bring to the table alongside cameras to boost performance [171]. A recent study [98] fuses LiDAR and infrared camera gatherings to identify objects in the surroundings under conditions with inadequate visibility. GANs are another choice for fusion architectures. GAN-based fusions are not mostly end-to-end models for detection and detection headers are generally used to give detection on the fused-reconstructed outputs [212], [31]. *Gao et al.* choose RGB and infrared cameras as their sensor suite and purpose GF-Detection to perform vehicle detection at nighttime. They exploit GANs to fuse encoded features of the RGB and infrared branches.

While there has been research and progress in the nighttime perception area, it can still be considered under-explored compared to the discussed algorithms which are designed with a bias towards daytime scenarios. Therefore, this area requires further exploration and development of robust algorithms and sensor technologies to ensure safe and effective autonomous driving at night.

Table 2.12: Categorization of datasets relevant to the study of practical and environmental challenges for perception systems.

Dataset Context	**Dataset Name**
Radar-included	NuScenes [80], Oxford RobotCar - Radar Extension [26], Astyx 6544 HiRes [219], DenseRadar [343], PixSet [121]
Image Dehazing	D-Hazy [16], Dense-Haze [13], HazeRD [365], NH-Haze [14], RESIDE [187], O-Haze [15]
Adverse Weather Conditions	Oxford RobotCar [214], Oxford RobotCar Radar-Extenstion [26], DENSE [33, 36, 142, 143, 316], NuScenes [80], Argoverse1 [88], Argoverse2 [327], Waymo Open Dataset [299], BDD100K [355]
Nightime	NuScenes [80], Waymo Open Dataset [299], Argoverse1 [88], Argoverse2 [327], Multispectral Pedestrian Detection [164], The TNO Multiband Image Data Collection [307], U2Fusion [336], Person Detection in Thermal Imagery [112], Pedestrian Detection in Far Infrared Images [169]

2.7 Conclusion

Despite the current maturity of perception models, single-modal solutions often lack the robustness and reliability of multi-modal approaches, especially in realistic scenarios. Multi-modal solutions have demonstrated superior performance when faced with challenges such as adverse weather conditions and different times of the day. However, integrating multiple modalities raises questions about sensor selection and the appropriate fusion techniques.

While different combinations of sensors have been explored in the literature, most research has focused on camera-LiDAR pairs, with less emphasis on Radar-based sensor suites. Thus, there is a gap that needs more comprehensive studies on radar capabilities and limitations, both as a standalone modality and in combination with other sensors. Additionally, evaluating the impact of adding radar to learning-based perception systems requires substantial annotated data to establish quantitative performance metrics.

Despite the availability of numerous multi-modal datasets, the majority include only camera images and LiDAR point cloud data. There are relatively few datasets with radar recordings which indicates a need for the development of more diverse and inclusive large-scale datasets to facilitate the study of the impact of this sensor on different perception tasks.

In this chapter, we provided a comprehensive overview of the different types of sensors used in the field of AD. Sensors are the foundational layer of every perception system, and selecting the appropriate sensor suite requires an understanding of the characteristics, advantages, and disadvantages of each sensor. Additionally, we discussed the operational principle of these sensors and areas in which each type can be advantageous. We also summarized various datasets, covering the recent and popular ones across aspects of recording conditions and sensing modalities. This overview helps bridge the gap between perception tasks and relevant datasets, enabling researchers to make informed choices when starting their investigations. The core of this paper is the organized presentation of perception-related tasks, allowing for a clearer identification of task-specific challenges. We explored a variety of methods, categorizing them by task, and provided insight into how researchers address these challenges. Finally, we highlighted two critical scenarios that challenge the ideal assumptions of many current methods: adverse weather conditions and nighttime driving. Developing robust solutions for these scenarios is essential for the realization of reliable perception systems to meet the crucial requirements of real-life autonomous driving situations.

Chapter 3

Case Study: Performance Evaluation of Perception Models

3.1 Introduction

Object detection is a critical task in AV perception systems, where the choice of sensor type and combination plays a pivotal role. Methods can be broadly categorized into single-modal approaches, using a single sensor, and multi-modal approaches, which leverage multiple sensor types. Each approach has its trade-offs in terms of efficiency, robustness, and applicability to real-world scenarios. These trade-offs must be carefully evaluated to identify solutions that meet the demands of AV applications.

Key challenges for object detection models include maintaining low latency and power consumption to support high-speed decision-making while ensuring robustness under diverse conditions, such as inclement weather or sensor failures. High-latency models are unsuitable for real-time reasoning, and unreliable performance in adverse conditions can jeopardize safety. Hence, evaluating models against these criteria is essential for practical deployment.

This chapter provides a comparative study of several deep learning-based object detection models, focusing on their performance across accuracy, latency, and robustness metrics. The selected models reflect a variety of sensor configurations and approaches, ranging from camera-only to camera-radar fusion techniques. By analyzing these models on the Nuscenes dataset, a comprehensive benchmark in AV research, this chapter aims to provide insights into the relative strengths and limitations of each approach.

3.2 Evaluation Dataset

Every presented model will be evaluated on the NuScenes dataset [80]. NuScenes is an extensive multi-modal dataset that consists of a thousand 20-second long scenes and is officially split into 700, 150, and 150 scenes for training, validation, and testing, respec-

tively. The scenes were recorded in two cities known for their dense traffic characteristics to capture a wide range of happenings that are possible to occur anytime while driving. With the goal of creating a multi-modal dataset in mind, the sensor suite was constructed upon 6 cameras, 5 radars, and 1 lidar to record each scene with a 360 field of view (FOV) coverage. Each recorded scene is further broken into keyframes (with annotations) and intermediate frames (without annotations). The extensiveness and multi-modality of the Nuscenes dataset enable the support for a variety of tasks that a self-driving car is expected to perform. This encompasses localization, detection, tracking, and prediction tasks. To measure and evaluate the performance of learning models on the dataset and make fair comparisons a standardized framework should be employed. Accordingly, the authors establish a number of metrics for both detection and tracking tasks taking a variety of aspects into account. Since this paper is centered around object detection, we specifically emphasize two key metrics in this domain, namely, the Average Precision (AP) and the Nuscenes Detection Score (NDS) which will be explained later.

3.3 Perception Models

In this section, we provide a thorough explanation of five deep-learning models that are designed for detection. The models are selected in a way to cover different sensing modalities.

3.3.1 PointPillars

Among all the single-modal object detectors, which only rely on one sensor, PointPillars [184] can be considered one of the well-known baseline models. The design of this model has taken place in three stages and tackles the problem of arranging point clouds into a form that is suitable to be used by detection schemes. Accordingly, a uniform grid of size X and Y is created on the xy-plane in the first stage. Points are organized as vertical columns that carry reflectance and location-related information. When the encoding is done, the features are used to generate "pseudo-images". In the second stage, a feature extractor is applied to the pseudo-images and extracts multi-resolution feature maps. Finally, all concatenated features from the previous steps are present to be fed to a head network to perform detection. In the original paper, the authors used the SSD detection head [203] to regress 3D bounding boxes. However, it is worth mentioning that PointPillars can do segmentation by just swapping the detection head with a segmentation one. Furthermore, the model evades using 3D convolutions on the raw point cloud by generating and using pseudo-images. This factor is one of the contributors to its fast-paced performance.

3.3.2 FCOS3D

The goal of the FCOS3D model is to address the problem of 3D object detection by adopting a camera-only framework [319], a departure from traditional LiDAR-based methods. The architecture is a fully convolutional one-stage detector that regresses 3D bounding boxes and other object attributes. The model is built on top of the FCOS proposed by Tian et al. [306]. The initial phase is a ResNet backbone that extracts the primary feature maps from the input image at different resolutions. Next, a Feature Pyramid Network (FPN) structure shapes the neck stage that prepares five-level feature maps which will be used as the basis of the detection procedure. Being an anchor-free method, the network does not depend on predefined anchor boxes to detect targets. Instead, it directly aims to detect the center of the objects and the proper bounding boxes which were assigned to different feature levels. Finally, the network will reason about object location, velocity, and 3D attributes. The authors, employ a 2D Gaussian distribution to assess the center of the objects.

$$c = e^{-\alpha[(x-x_0)^2+(y-y_0)^2]} \tag{3.1}$$

In the equation above, α is a hyperparameter that regulates how strict center-ness boundaries are defined.

3.3.3 RRPN

A regional proposal network of two-stage object detectors is in charge of extracting RoIs where objects possibly exist. These regions are called proposals and are used in the second stage to produce bounding boxes and do classification. The RPN stage is known to be a bottleneck in these detectors which causes limitations in real-time application for AV. Accordingly, Nabati et al. [222] propose RRPN and aim to tackle this bottleneck with a simple yet effective idea. RRPN is a multi-modal object detector that employs camera and radar sensors. Instead of using a deep network to extract regions of interest, the radar detections are taken as RoIs. More specifically, three key steps take place in the pipeline to generate the RoIs. First, the radar detections, covering a 360^o FOV, are mapped to the coordinates of the main camera considering the calibration parameters. Once completed, each image frame with the mapped detections contains valuable information about the whereabouts of an object instance in the view of the camera. Next, the mapped detections are used to generate RoIs for the detection stage. However, radar detection does not necessarily point to the center of the object and does not contain any information about the size of the detected object. Consequently, the authors handcraft a set of bounding boxes with different sizes and aspect ratios. Then, the set of anchor boxes is assigned to every radar detection point (also called Points of Interest or PoIs). As the last step, according to the depth information from each radar detection, the anchors are inversely scaled. This step exists to take object distance from the ego vehicle into account which directly impacts the size of each object in the camera view.

Distance compensation is done as below where S_i is the size and d_i is the distance of i^{th} object, and α and β parameters are optimized through maximizing the IoU between the generated and ground truth bounding boxes adopting grid search.

$$S_i = \alpha \frac{1}{d_i} + \beta \tag{3.2}$$

In the second stage, the generated proposals are given to a Fast R-CNN network with a ResNet backbone to perform classification and regress bounding boxes.

3.3.4 CenterFusion

Although RRPN architecture has enabled the model to achieve close to real-time performance, every radar detection is taken into account to generate proposals. However, only a number of these proposals are kept (about 2000) and the rest are discarded. Therefore, the radar information is not well-refined and efficiently used which could result in inferior performance compared to fusion models. CenterFusion [223] is another camera-radar perception model that uses a middle-fusion scheme to harvest radar advantages toward more robust performance. The architecture of this model can be broken down into three key stages. The first stage solely depends on images from the camera data. The first detections are generated by a CenterNet network [372] in the way that image features are extracted to make predictions on where object centers can be. Then, object properties such as 3D dimension, rotation, and depth are regressed based on the previous predictions. In the second stage, radar point clouds are processed in a number of steps on the radar channel. They are used to generate frustum-shaped RoIs to filter out unrelated points and keep the ones that actually correspond to the objects on the image plane. In addition, point clouds go through a pillar expansion procedure to compensate for inaccurate height data. Having completed the radar associations to the corresponding objects, depth and velocity information are used to create radar feature maps. Next, radar and image feature maps are aggregated and fed to the secondary set regression heads to refine and re-estimate the mentioned attributes. At last, all the primary and secondary information are fed to a decoder to render 3D detections.

3.3.5 Transfusion

Among all fusion models, lidar and camera pairs are popular and well-known in the community. Taking advantage of lidar modality offerings, many fusion models have shown improved performance compared to single-modal ones. In the meanwhile, Bai et al. [23] propose TransFusion that exploits the concepts of transformers in its architecture in the middle of the detection mechanism. The structure of the model can be broken down into two streams, i.e. lidar and camera streams. Lidar points cloud is accepted as the first stream input which passes through a VoxelNet [374] backbone for feature extraction from the bird's eye view (BEV). These feature maps will be further used in

the first decoder transformer section for initial bounding box regression. CenterNet is employed as the 2D backbone in the Camera stream to extract feature maps from the input images.

Having the feature maps from both modalities, object queries for the first decoder layer will be generated. The authors refuse to set queries as learnable parameters and take extra steps to generate them based on the input data. More specifically, each object query contains both position and feature queries. Unlike previous works, the position query is built based on the observed whereabouts of the candidate objects on the extracted heat maps. Feature query embeddings contain information about other properties such as box size, etc. The knowledge from image feature maps is injected into the lidar features to achieve better query initialization. In other words, high-resolution image features help to detect instances that are difficult to acknowledge by lidar due to their sparse nature. Finally, the queries are fed to the first decoder layer and primary predictions are generated. In the second stage, the initial predictions and object queries from the previous stage are fused with image features by a handcrafted module called spatially modulated cross attention (SMCA) to build soft associations. This enables the model to exploit rich contextual information from the camera and perform more robustly. In the end, after refining and fusing the information acquired from the inputs, a feed-forward network will generate the final predictions.

3.4 Evaluation

This section belongs to the performance evaluation of the methods described in the previous section. Evaluation and accuracy metrics are defined followed by a thorough analysis of the performance of each model.

3.4.1 Accuracy Metrics

Nuscenes framework defines a number of metrics with which models can be evaluated. Two of the key metrics, namely, AP and NDS are chosen for evaluation. We use AP both for measuring the detection accuracy of each model for some specific and key categories as well as the mAP for taking an average over all categories.

Average Precision metric The Average precision metric is widely used in detection tasks. It evaluates the precision-recall curve which shows a trade-off between precision (the portion of true positives among all predicted positives) and recall (the portion of true positives among all actual positives) [129]. For object detection tasks, the measurement of the intersection of union (IoU) with ground truth as the matching function is mostly common. One downside of the IoU matcher is that the method is biased inversely towards object sizes and is affected by object orientation as well [100]. However, in Nuscenes, the 2D distance between the centers is considered with AP calculations which helps to solely focus on detection and filter out unwanted impacts from object properties such

as those mentioned earlier. Consequently, the distance threshold for defining a match is defined as below.

$$D \in \{0.5, 1, 2, 4\} \tag{3.3}$$

The mean Average Precision for detection on C classes is defined below:

$$mAP = \frac{1}{|C|.|D|} \sum_{c \in C} \sum_{d \in D} AP_{c,d} \tag{3.4}$$

Nuscenes Detection Score Following the aforementioned method to compute AP, the authors establish five other metrics dedicated to other object properties such as size and velocity. The Nuscenes detection score (NDS) metric is where every 6 evaluation metrics in a weighted combination manner to collectively represent a method's performance. The NDS is defined as below where mAP is the mean Average Precision, and TP is a set containing the other 5 metrics.

$$NDS = \frac{1}{10} \left[5\, mAP + \sum_{mTP \in TP} (1 - min(1, mTP)) \right] \tag{3.5}$$

3.4.2 Experimental Results

In this section, we provide an analysis of the performance of the detectors from three different aspects. First, we will analyze the learning curve of the models with a batch size of one, which we can refer to as *Basic Learning Curve*. Next, the detection accuracy is assessed versus the number of parameters that build up each architecture. The parameter count is an indication of the required computation power for each model which is an important concern for deployment in field. Lastly, the models' robustness against severe weather conditions is assessed. Specifically, the impact of rainy weather on the detection accuracy will be studied. All models are trained on the NuScenes dataset to achieve their optimal performance.

Basic Learning Curve

When a training process is done with a batch size of one, also known as stochastic gradient descent, it reveals information about the learning characteristics. In this section, all models have been trained on the NuScenes dataset with a batch size of one for 40,000 iterations. The total loss of each model is converted to the dB scale for better visualization of large and small scales and the results are depicted in Fig. 3.1.

As observed, among all models, RRPN has the fastest speed of convergence whereas TransFusion-LC requires more training in search of an optimal point. Furthermore, CenterFusion shows superior training stability with a smooth curve and little fluctuations, while others are more sensitive to the ordering of data. Notably, while lower loss values can be an indication of better performance, it is not the only factor and several other aspects should be considered for a comprehensive evaluation.

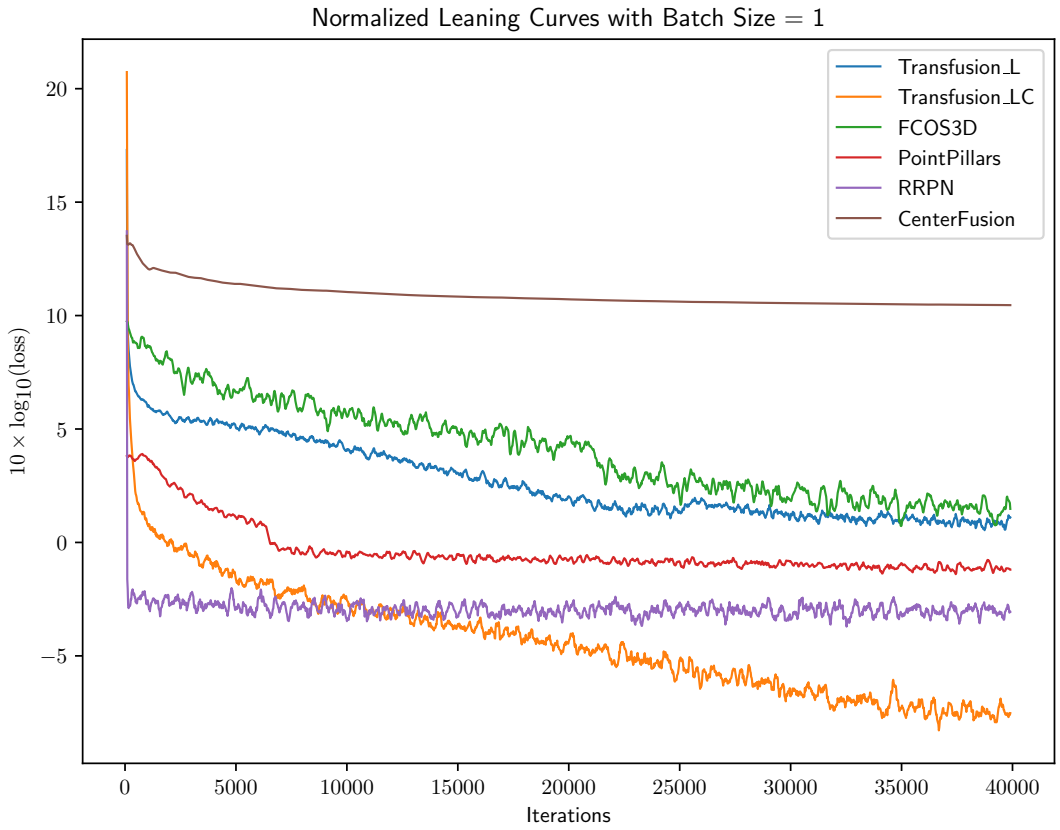


Figure 3.1: The smoothed basic learning curves of the selected models. Linear filtering is applied to provide better visibility.

Accuracy of Detection

The optimum performance of the selected algorithms is brought in Table 3.1. It should be noted that the RRPN model only uses front and back cameras as well as front and two rear radar data for training. PointPillars+ superior performance is achieved through more training, sampling, and finer choice of pillar size while the model is the same.

Figure 3.2 summarizes model accuracy versus parameter count. An ideal model is one that provides accurate detections while having a small number of parameters. This can be found in the upper-left corner of the figure. It can be observed that all models are more accurate than the baseline (PointPillars).

Both mono and multi-modal TransFusion models demonstrate supremacy compared to all other methods in terms of overall mAP and also for each category. One primary factor is the exploitation of very fine voxel resolution (0.075, 0.075, 0.2) [23] which results in the preservation of small details. On lidar-based methods, Transfusion-L performs better than both PointPillars algorithms despite both having about the same parameter

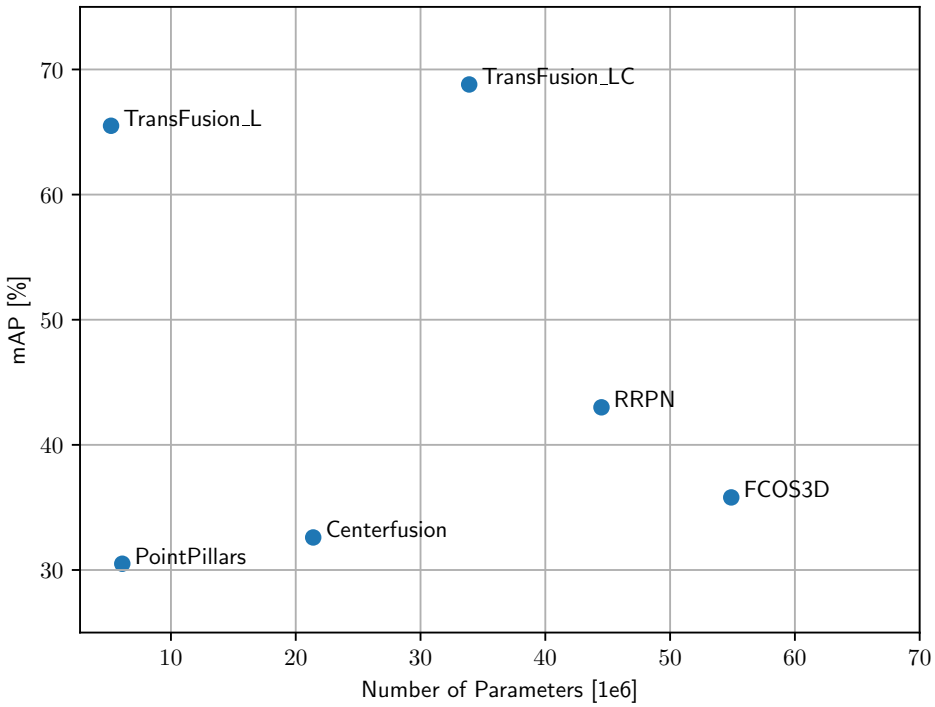


Figure 3.2: Comparison of detection accuracy (mAP) vs number of parameters.

count. However, moving towards a more complex model, TransFusion-LC)improves the mAP only by a margin of 3.3%. Contrarily, while FCOS3D has many parameters, the performance falls short of expectations, compared to RRPN with fewer parameters for more accurate detections.

From the above, it can be inferred that the higher the parameter count is, the higher the complexity will be, and consequently, the more computational power is needed for the model to perform. However, there are limitations to power and computation resources on an AV platform. Therefore, the growth of model complexity should be justifiable by improvements in accuracy.

Robustness to Different Weather Conditions

One aspect of object detectors, which is mostly neglected in many studies, is how reliable their detections are when facing rough weather conditions such as rain, fog, snow, etc. In this section, the effect of rainy weather on the overall and category-base accuracy of three models, including PointPillars, FCOS3D, and CetnerFusion is studied. We divided the NuScenes validation dataset into two sets, normal and rainy weather, and tested the models on both. Among these, PointPillars and Centerfusion show the most and the least robustness to change in weather conditions with a drop of 2.52% and 12.24% in overall accuracy, respectively. The drop in performance mostly stems from

Table 3.1: Perception models detection accuracy measured by mAP and NDS. The values provided are in percentages.

Model	Modality			mAP	Car	Truck	Pedestrian	NDS
	C	L	R					
TransFusion-LC	✓	✓		**68.8**	**87.1**	**60.0**	**88.4**	**71.7**
TransFusion-L		✓		65.5	86.2	56.7	86.1	70.2
RRPN	✓		✓	43.0	44.2	51.6	22.0	-
PointPillars+		✓		40.1	76.0	31.0	64.0	55.0
FCOS3D	✓			35.8	52.4	27.0	39.7	42.8
CenterFusion	✓		✓	32.6	50.9	25.8	37.0	44.9
PointPillars		✓		30.5	68.4	23.0	59.7	45.3

not being able to correctly detect small or narrow objects such as traffic barriers and pedestrians. PointPillars heavily struggles to correctly detect pedestrians, due to how rain impacts the recorded point clouds, while the other two almost experience a small decline. Furthermore, regarding detecting large objects such as trailers, it is observed that FCOS3D performs well under normal conditions but is considerably influenced by image quality degradation when it rains.

3.5 Conclusion

We have performed an intensive analysis of the performance of a wide range of perception models in the field of AV, which consists of mono and multi-modal algorithms. Our comparison framework is based on the extensive NuScenes dataset and its evaluation metrics. This study has delved deep into the learning characteristics of the selected models and emphasizes on how well and quickly each model adapts to the training data. Furthermore, detection accuracy and its relation with the degree of complexity is highlighted. It shows that even though there is a trade-off between model complexity and accuracy, increased complexity does not necessarily lead to more precise detections. Lastly, algorithms are put under rough weather conditions to measure their robustness. Robustness to different weather conditions, noise, etc. is a key to the realization of AV in practice.

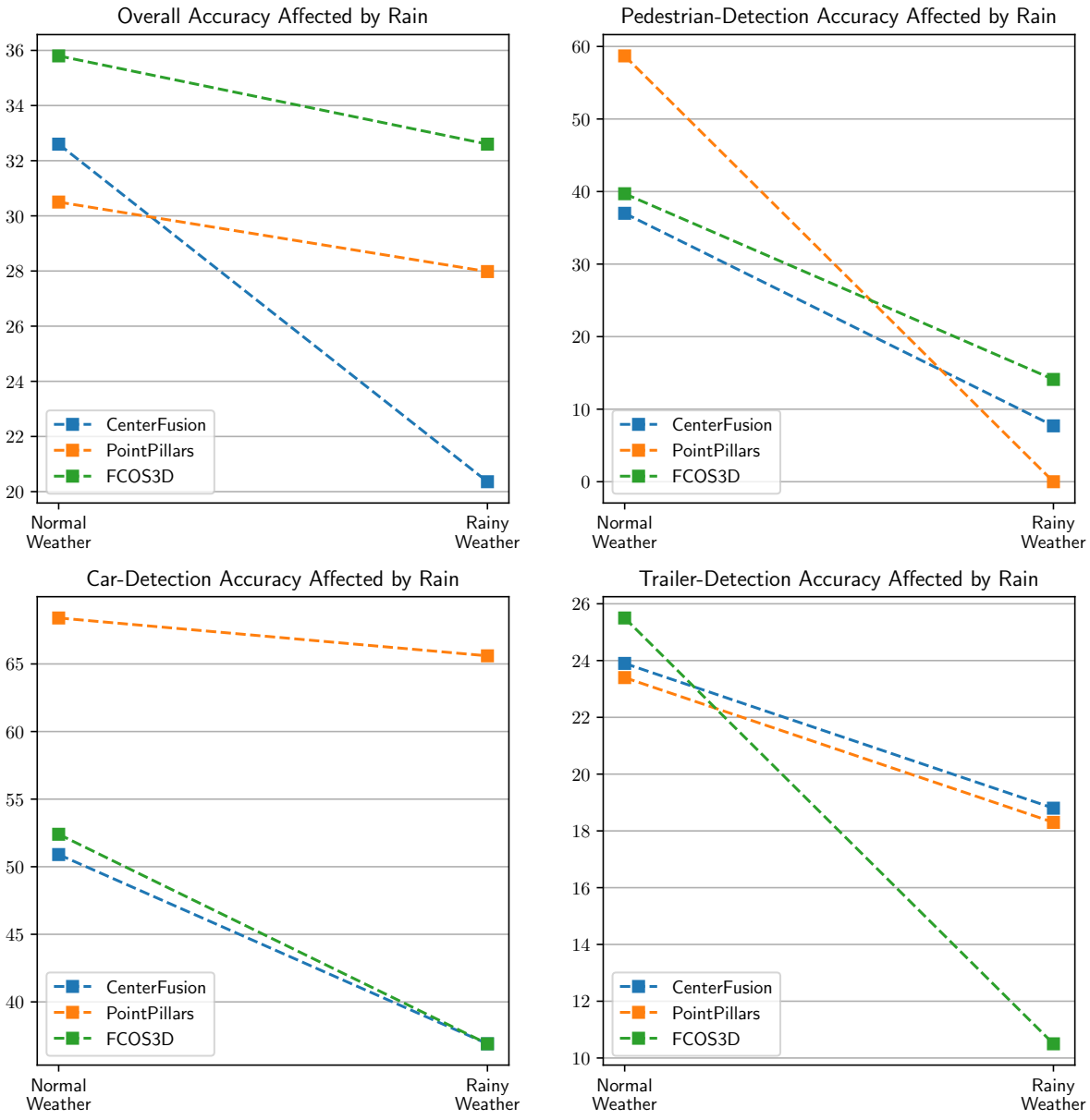


Figure 3.3: The effect of rainy weather on detection accuracy.

Chapter 4

System Design and Methodology

In previous chapters a comprehensive overview of the perception systems of AV was given, highlighting their purpose, characteristics, challenges, and a spectrum of approaches. Perception systems are an essential step for an AV to become aware of the happenings in its surroundings. Traditionally, cameras and then LiDAR sensors were used as the main sensing modalities in system design. These approaches have been shown to perform well under ideal situations, for instance, adequate illumination or normal weather conditions. However, their performance significantly degrades under realistic scenarios where a variety of events occur such as different illumination conditions, or reduced visibility during adverse weather conditions such as heavy rain, fog, snow, etc. Consequently, the sensitivity of these methods to the changes in environmental situations undermines the reliability of perception systems, posing significant safety concerns for AVs.

4.1 Problem Overview

Recent studies, such as [86,205,257,266,351,356], have investigated Radar sensor characteristics which led to uncovering its inherent advantages that are beneficial in enhancing the performance and robustness of the perception system. However, Radar sensor applications in the perception field remain relatively underexplored compared to cameras and LiDARs, creating a knowledge gap and underscoring the need for more comprehensive research on Radar-aided perception methods. Furthermore, high-quality and diverse datasets are an essential requirement and tool in deep learning-related studies. Even though there are more than 60 public datasets [144] for perception purposes, about 10 are available including Radar recordings [25, 26, 80, 121, 217, 219, 276, 280]. Additionally, most of these datasets offer Radar measurements as point clouds which are extremely sparse compared to denser LiDARs point clouds.

To address this gap, our main contribution is the design of a unified system that aims to improve the quality of the existing Radar point cloud portion of the datasets. Inspired by Radar's analog azimuth-range maps, our framework employs a probabilistic occupancy grid mapping system. This system processes sparse Radar point cloud frames

and generates enriched and detailed maps. The semi-continuous properties of these generated maps recreate the environment and capture the contextual details around the ego vehicle. These enriched maps serve as a superior alternative to sparse raw point cloud frames, facilitating the study of Radar sensor modalities and accelerating the development of more robust and performant models. While there have been attempts at occupancy grid mapping using Radar measurements [116, 244, 289, 290, 326], to the best of our knowledge, our system is the first to offer a generalized solution for transforming Radar point clouds into occupancy maps specifically to enhance perception performance in the field of object detection.

4.2 Background

The goal of occupancy grid mapping algorithms, which is extensively studied in the literature, is to calculate the posterior occupancy probability distribution over the map given the measurements of the range finder sensor and the odometry information of the ego vehicle. At each time step, the ego vehicle is located at coordinates (x, y), measured and reported by the odometry sensors, and detects targets located at various positions in its surroundings. Consequently, the algorithm relies on two sets of inputs: sensor measurements and vehicle position. While sensor measurements are directly obtainable from Radar point clouds, accurately estimating vehicle pose remains a complex task, often tackled using Simultaneous Localization and Mapping (SLAM) algorithms for robust localization, as extensively reviewed in literature in [97]. In this chapter, we assume the vehicle position is known at every time step, simplifying the mapping problem to one with known poses.

Initially, the mapping environment is established with arbitrary height and width, which are often equal in this context and denoted as *grid_edge*. Then, the grid is uniformly divided into equally sized grid cells. Each cell has a binary random state toggling between occupied and free which are denoted by 1 and 0, respectively. The expression $p(\mathbf{m}|z_{0:t}, x_{0:t})$ denotes the probability of the map $\mathbf{m}$ which consists of grid cells, $c_{m,n} \in \mathbf{m}$. Here, $z_{0:t}$ is the sensor measurements, and $x_{0:t}$ is the pose information of the ego vehicle from $t = 0$ up to the present time.

Calculating the posterior of a map is a high-dimensional problem with requires further simplifications for practical use. For example, mapping an area of 100m by 100m with cells of size 10x10 cm^2 gives a total number of 1,000,000 variables, if binary, results in $2^{1,000,000}$ different maps. To manage this complexity, a common assumption is made that the states of cells are independent of each other. Consequently, the posterior becomes a product of marginal distributions, as shown in eq. 4.1.

$$p(\mathbf{m}|z_{0:t}, x_{0:t}) = \prod_{m,n} p(c_{m,n}|z_{0:t}, x_{0:t}) \tag{4.1}$$

By using factorization, the estimation of the posterior map given the sensor measure-

ments and the ego vehicle positions can be decomposed into individual binary estimations of cells. This problem can be solved by employing a filtering algorithm known as the Binary Bayesian Filter. Assuming static cell states, Bayes' rule applies as follows:

$$p(c_{m,n}|z_{0:t}, x_{0:t}) = \frac{p(z_t|c_{m,n}, z_{0,:t-1}, x_{0:t})p(c_{m,n}|z_{0:t-1}, x_{0:t})}{p(z_t|z_{0:t-1}, x_{0:t})} \tag{4.2}$$

Further simplification using Markov's assumption, where future information in the prior is disregarded, leads to eq. 4.3:

$$p(c_{m,n}|z_{0:t}, x_{0:t}) = \frac{p(z_t|c_{m,n}, x_t)p(c_{m,n}|z_{0:t-1}, x_{0:t-1})}{p(z_t|z_{0:t-1}, x_{0:t})} \tag{4.3}$$

Applying Bayes' rule to the forward measurement model $p(z_t|c_{m,n}, x_t)$, we derive the following expression:

$$p(z_t|c_{m,n}, x_t) = \frac{p(c_{m,n}|z_t, x_t)p(z_t|x_t)}{p(c_{m,n})} \tag{4.4}$$

Substituting this into the expression 4.3, the final equation is obtained:

$$p(c_{m,n}|z_{0:t}, x_{0:t}) = \frac{p(c_{m,n}|z_t, x_t)p(z_t|x_t)p(c_{m,n}|z_{0:t-1}, x_{0:t-1})}{p(c_{m,n})p(z_t|z_{0:t-1}, x_{0:t})} \tag{4.5}$$

Similarly, an expression for the empty cells, $p(c^*_{m,n}|z_{0:t}, x_{0:t})$, can be derived as show below:

$$p(c^*_{m,n}|z_{0:t}, x_{0:t}) = \frac{p(c^*_{m,n}|z_t, x_t)p(z_t|x_t)p(c^*_{m,n}|z_{0:t-1}, x_{0:t-1})}{p(c^*_{m,n})p(z_t|z_{0:t-1}, x_{0:t})} \tag{4.6}$$

By computing the ratio of eq. 4.5 and eq. 4.6, certain terms will cancel out thus simplifying the computations significantly. Replacing $p(c^*_{m,n}|z_t, x_t)$ by $1 - p(c_{m,n}|z_t, x_t)$ we will have:

$$\frac{p(c_{m,n}|z_{0:t}, x_{0:t})}{p(c^*_{m,n}|z_{0:t}, x_{0:t})} = \frac{p(c_{m,n}|z_t, x_t)}{1 - p(c_{m,n}|z_t, x_t)} \frac{p(c_{m,n}|z_{0:t-1}, x_{0:t-1})}{1 - p(c_{m,n}|z_{0:t-1}, x_{0:t-1})} \frac{1 - p(c_{m,n})}{p(c_{m,n})} \tag{4.7}$$

The expression of $l(x)$ is known as the log odds of probabilities and relates to the probability $p(x)$ as follows:

$$l(x) = log(\frac{p(x)}{1 - p(x)}) \tag{4.8}$$

$$p(x) = 1 - \frac{1}{1 + e^{l(x)}} \tag{4.9}$$

Equation 4.1 computes the product of a large number of probabilities, each ranging between 0 and 1. This results in extremely small values, posing numerical issues in

practical implementation. Thus, the log odds representation helps avoid numerical issues by transforming the eq. 4.7 into a summation of terms:

$$l(c_{m,n}|z_{0:t}, x_{0:t}) = log(\frac{p(c_{m,n}|z_t, x_t)}{1 - p(c_{m,n}|z_t, x_t)}) + log(\frac{p(c_{m,n}|z_{0:t-1}, x_{0:t-1})}{1 - p(c_{m,n}|z_{0:t-1}, x_{0:t-1})}) + log(\frac{1 - p(c_{m,n})}{p(c_{m,n})}) \tag{4.10}$$

Thus, the final update rule for the filter is expressed in log odds representation as:

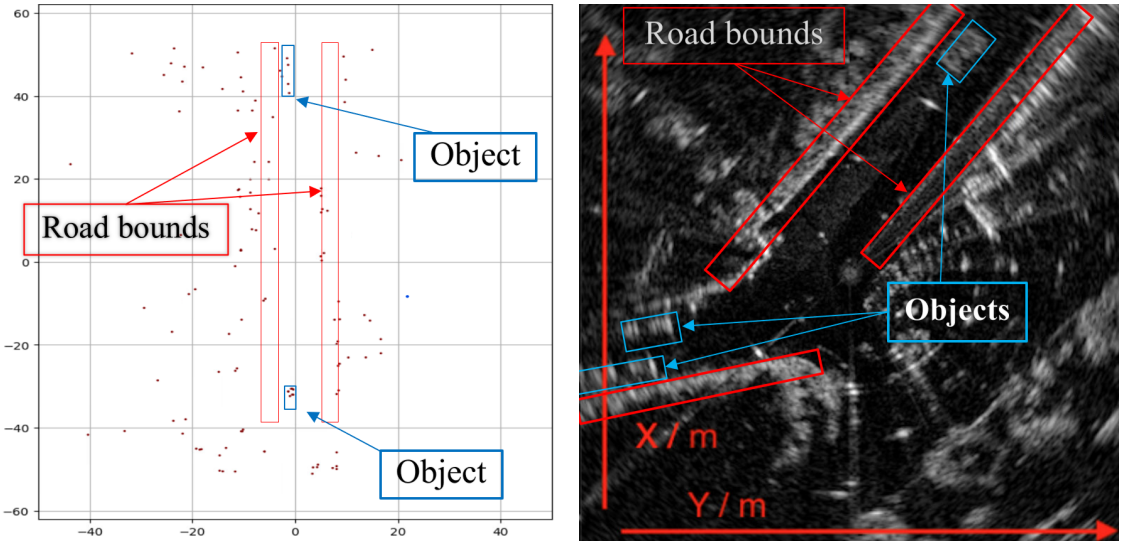
$$l(c_{m,n}|z_{0:t}, x_{0:t}) = \underbrace{l(c_{m,n}|z_t, x_t)}_{\text{Inverse Sensor Model}} + \underbrace{l(c_{m,n}|z_{0:t-1}, x_{0:t-1})}_{\text{Recursive Term}} - \underbrace{l_0(c_{m,n})}_{\text{Map Prior}} \tag{4.11}$$

In the expression above there are three distinct terms, playing a distinct role in adjusting the probability of occupancy for each cell. Starting with the last term, $l_0(c_{m,n})$ represents the prior information of the map at the beginning, often a constant value denoted as l_0. Typically, there is no previously available information from the state of the cells $c_{m,n}$, as they can be either occupied or empty. This results in a zero log odds value. The second term is a recursive relation that contains the posterior information of the previous time step at $t-1$. The first term is the core of this filtering algorithm, known as *Inverse Sensor Model* or ISM. This modeling method is commonly used when measurement space is more complex than state space. The *Inverse Sensor Model* term captures the unique characteristics of the sensor and models how it perceives its surroundings. At each time step t, the cells residing in the perception field of the sensor will be affected and the states will be updated accordingly. It is crucial to define and adjust this term according to the type and operating principles of the sensor utilized.

4.3 PARMG: Pseudo-Analog Azimuth-Range Radar Maps Generator System

In the preceding section, we introduce our proposed framework aiming to address the challenges mentioned at the beginning of this chapter by transforming the sparse Radar measurement to detailed occupancy maps, capable of presenting the contextual details of the surrounding environment. Recent studies [26, 254] have demonstrated that semi-continuous Radar scans are significantly successful in recording and preserving the details of driving scenarios compared to sparse point cloud measurements. Figure 4.1 illustrates a comparison of Radar frames from NuScenes and Oxford Robotcar datasets, presented in the point cloud and range-azimuth scan format. The difference in the level of recorded details from each scene can be clearly observed. While the scenes have successfully captured the approximate outline of objects and road structure, the raw point cloud frame struggles to preserve the structure of the surrounding world, hardly distinguishable without additional images or LiDAR data.

Almost all object detection models rely on the features learned throughout the learning process to find and pinpoint the object instances on an input frame. Even a powerful object detector model cannot perform well on an extremely sparse input without much information on the layout or shape of objects.



(a) Point cloud Radar frame from Nuscenes dataset

(b) Analog Radar scan from Oxford Robotcar dataset

Figure 4.1: Comparison of scenes details measured in analog scan and point cloud format.

Inspired by the properties of analog scans, we propose a unified system called Pseudo-Analog Azimuth-Range Radar Maps Generator System (PARMG) that utilizes the Bayesian filtering algorithm to process Radar point clouds to produce a detailed representation of the surroundings. Our goal is to derive probabilistic occupancy grid maps that mimic the behavior of analog scans. Our system core is an inverse sensor model designed specifically based on the characteristics of a Radar sensor. This design enables our framework to build occupancy maps from the view of a Radar sensor accounting for its operating principle and properties. Additionally, for the two-dimensional input point clouds, we implement an adaptive height augmentation process that assigns height to different cells relative to their occupancy states. This gives slice-wise depth to the generated map, taking advantage of the key 3D information available from Radar intrinsic matrices. Moreover, driving scenarios contain both stationary and moving objects. However, the traditional occupancy grid mapping algorithms are mainly used to map the stationary state of the environment. To this end, we design our system to extend grid mapping to support dynamic objects, such as moving vehicles and pedestrians, as well as static ones. Furthermore, PARMG includes a sensor fusion scheme that supports an arbitrary number of sensors and enables the mapping procedure in different configurations. Therefore, our framework can be customized to be used on different Radar point clouds to generate the occupancy maps which we call *Pseudo-Analog Azimuth-Range Radar Maps*. The

architecture of our system is illustrated in fig. 4.2

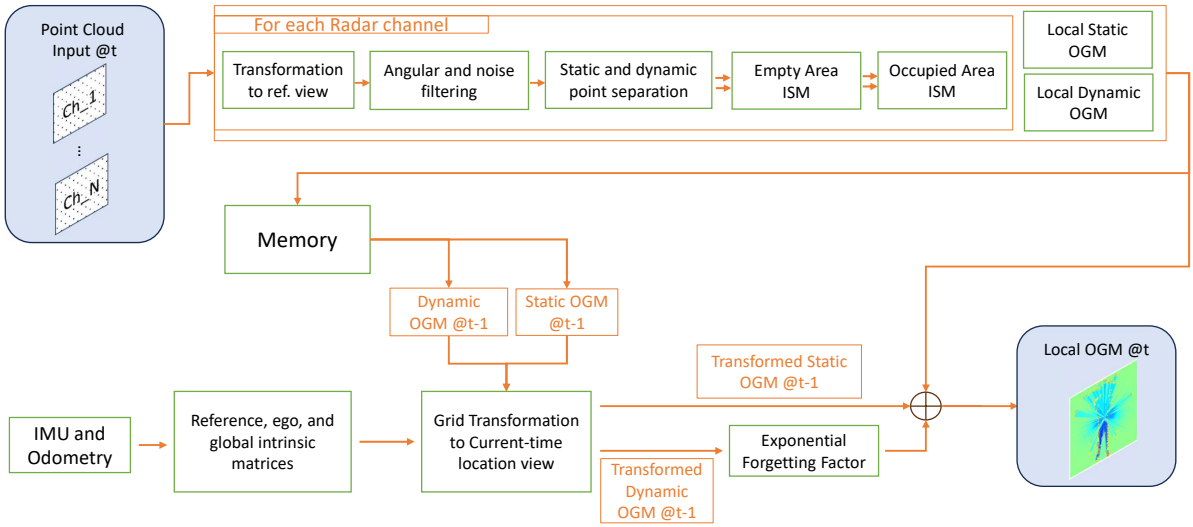


Figure 4.2: An overview of the architecture of PARMG including (1) multi-view sensor fusion for supporting various sensor configurations, (2) the Radar-specific inverse sensor modeling module responsible for the calculation and updating the probability state of grid cells, (3) the dynamic target handling utilizing an exponential forgetting factor, and (4) the motion compensation unit implemented for ensuring continuous, correct and consistent mapping of the environment.

4.3.1 Radar-specific Inverse Sensor Modelling and Adaptive Height Augmentation

Traditional occupancy grid mapping has primarily focused on range sensors that use sound and laser beams to measure the distance of the targets. These methods often employ different approaches, such as ray casting, to characterize the sensor perception behavior. For example, Octomap [158] is a well-known software developed for 3D occupancy mapping for LiDAR sensors. However, the LiDAR-oriented developed models cannot be directly used for Radar sensors due to their fundamental differences. LiDARs use laser beams, with wavelengths of 905 and 1550 nm, to interact with the environment and calculate the distance of targets by measuring the travel time of the beam. In contrast, Radars measure the distance by emitting lower frequency (e.g. 74 GHz) electromagnetic (EM) waves, with the wavelength in orders of millimeters. The spectral difference of these beams results in different characteristics of the sensors.

The laser beams emitted by LiDAR behave similarly to visible light. They can be reflected, absorbed fully or partially, and can pass through transparent objects. In real-world driving scenarios, nearly all objects are opaque, reflecting a portion of the beams and absorbing the remainder. Therefore, in the mapping procedure, the area behind

each detected object is deemed uncertain as no information is available from this mode of measurement. Conversely, Radar's EM waves can penetrate objects, fully or partially, depending on the material and electromagnetic properties. For instance, while metallic surfaces such as vehicle bodies, hoods, and doors are excellent reflectors due to their composition, materials like plastic or composite components, commonly found in vehicles, can be partially penetrated by the EM waves. This means some objects that are opaque to LiDARs can be partially transparent to Radars, allowing the sensor to see behind them. This significant difference between LiDARs and Radar must be considered when designing the inverse sensor model.

In this section, we present details on the initial state of the grid cells and the inverse sensor model module of our system which is designed to account for these unique characteristics. Inspired by [243, 290], this module utilizes two distinct functions denoted as *Occupied* and *Empty Area Inverse Sensor Model*. The occupied area ISM models the presence of an object by a Gaussian distribution and the empty area ISM identifies the unoccupied space based on the information from the whereabouts of the targets. This approach ensures a comprehensive representation of both occupied and empty areas in the environment tailored by the Radar sensor.

Grid Initialization

The occupancy grid mapping algorithm developed in eq. 4.11 is a recursive procedure that utilizes the present-time measurements to update the information from the previous time step. Consequently, an initial condition should be specified at the beginning. This initial condition corresponds to the term $l_0(c_{m,n})$ which contains the prior information of the environment. If previous knowledge is available, it should be incorporated into the algorithm through this term. However, in most cases, no initial information is available from external sources or sensors. Therefore the system is initially uncertain about the state of every observable cell.

$$p(c_{m,n} = occupied)|_{t=0} = 0.5 \;\Leftrightarrow\; l(c_{m,n})|_{t=0} = 0 \quad \forall m, n \tag{4.12}$$

As shown above, to initiate the grid at the beginning, a value of zero is assigned to every cell, representing an equal probability of being occupied or empty. The algorithm will then update these values based on the sensor measurements, progressively refining the occupancy map as more data is collected. This initialization ensures that the mapping starts with a neutral state and gradually builds up an accurate representation of the environment.

Occupied Area Inverse Sensor Model

Many software and mature algorithms such as Octomap [158] have been developed for grid mapping purposes, but they typically model how LiDARs perceive the environment. Our design performs occupancy grid mapping from the perspective of a Radar sensor.

An occupancy map consists of three regions: occupied, empty, and unknown. At the beginning of the process ($t = 0$), the state of every cell in the map is unknown. When the first point cloud frame is fed to the system, containing point measurements of objects, initial occupied places are identified.

These sensor measurements are not perfectly accurate and include a margin of error. The measured target identifies the location with a high likelihood of being occupied and the surrounding areas are considered with decreasing probabilities to account for measurement uncertainties. To model this behavior, we use a 2D Gaussian distribution in the coordinates of the measurement sensor and adjust its parameters with respect to the technical specifications. The occupancy PDF is expressed in eq. 4.13.

$$p_i(r, \theta) = A.exp(-(\frac{(r - r_i)^2}{2\sigma_r^2} + \frac{(\theta - \theta_i)^2}{2\sigma_\theta^2})) \tag{4.13}$$

The expression above describes an occupancy probability around a target located at (r_i, θ_i). The coefficient A is a constant value of $\frac{1}{2\pi\sigma_r\sigma_\theta}$ that is used for the normalization of the distribution. The shape of the Gaussian distribution is determined by the two error parameters, σ_r and σ_θ which correspond to detection range and azimuth angular measurement accuracy, respectively. Intuitively, as we move farther from the measurement location, the likelihood of the cells being occupied by that object decreases. For example, if measurement errors of a sensor are large, a broader span of the distribution is employed to account for these errors. At the end, the affected cells $c_{m,n}$ are assigned the value $l(c_{m,n}|z_t, x_t) = l(p_i(r, \theta))$.

In addition to sensor accuracy parameters, the Radar Cross Section (RCS) is one the outputs from the measurement and is a way of assessing the detectability of an object, often tailored in dBsm (dB per square meter) unit. Objects with with larger physical size or better reflectance exhibit larger RCS values and, therefore become more detectable. In order to incorporate the impact of the RCS values on the occupancy probability, we map the RCS values to a range of 0.5 to 1.5, utilizing a sigmoid function, and multiply the final probabilities of each target by the corresponding mapped RCS coefficient.

In this framework, the ISM is applied to each target individually to process and update the state of the surrounding cell. For generating a high-quality map, discussed in 5.2.1, cells smaller than a certain size are required. However, smaller cells increase the total cell count, leading to computationally expensive processes. Therefore, we employ two estimation strategies to reduce the computational cost of cell processing. Firstly, the cells within the range of $2.5\sigma_r$ and $2.5\sigma_\theta$ of the center of the probability distribution are preserved, totaling more than 95% of the information, whereas the cells falling farther are discarded. Secondly, assuming that the cell area is smaller than the distribution area, we estimate the integration of the PDF as the sum of the cell areas, making a trade-off between estimation error and computational efficiency.

Furthermore, to avoid the strong accumulation of large numbers, we define a maximum and minimum range for the probabilities. According to the log odds definition in eq. 4.8, as the probability approaches to zero or one, the log odds values increase

infinitely. To prevent this, we implement an upper bound of 0.9991 and a lower bound of 9×10^{-4} that constrains the log odds value in the range of $[-7, 7]$. These limitations ensure the model produces smooth output and maintains a consistent dynamic range by avoiding extremely large numbers.

Empty Area Inverse Sensor Model

When a target is detected by the Radar and no other targets are present in the perception field, we can infer that the area between the vehicle and the target is empty. This requires a mechanism that applies to the cells within the perception cone and decreases their probability of occupancy. In LiDAR-based methods, the area between the vehicle and each target is designated as a zone free of any occupants and no other information can be inferred from the area behind the objects, at larger ranges. However, considering the penetrative properties of EM waves, we customize a second function for our ISM that enables the mapping system to consider the targets observed behind closer objects.

In this model, if there are no other objects between the sensor and the detected target, the cells within the detection area are biased towards the empty state by a constant negative log odds value, denoted by $l_{empty} < 0$. However, if there is an object in the way, the empty area shared with farther targets will still experience a decrease in occupancy probability, as well as the area between the closer and the further targets. But the closer target and its neighborhood will not experience this decay. Therefore, farther objects will not be masked by nearer targets, and the empty area among them is preserved. These advantages enable us to make the most use of the measurements provided by the sensor to build more informative and detailed maps with a longer range of detection.

Additionally, unoccupied areas can be inferred between two consecutive targets. If two neighboring targets are detected at (r_i, θ_i) and (r_j, θ_j) and no detection occurs with the angular interval of $[\theta_i, \theta_j]$, this conical zone can be deemed unoccupied. Consequently, the state of all the enclosed cells can be pushed toward being object-free.

The chance of missing a target increases with the distance due to phenomena such as multi-path, low RCS, or signal attenuation. To account for these kinds of measurement errors at larger distances, we implement a range-dependent function, unlike the previous constant function, that associates a low occupancy probability to closer cells and gradually increases the probability to 0.5, with increasing range. The choice of this function is arbitrary and can be configured depending on the application and the dimensions of the mapping area. Some studies [290] have proposed linear functions for this purpose. Experimentally, we chose an exponential form that associates more probability of emptiness to the close neighborhood of the vehicle and makes a transition to the unknown state for the cells farther away.

Figure 4.3 provides an example of the Radar-specific ISM interacting with three objects at different locations. The upper figure 4.3a illustrates the scenario in sensor view, and the bottom figure 4.3b presents the scenario in BEV, all cropped to

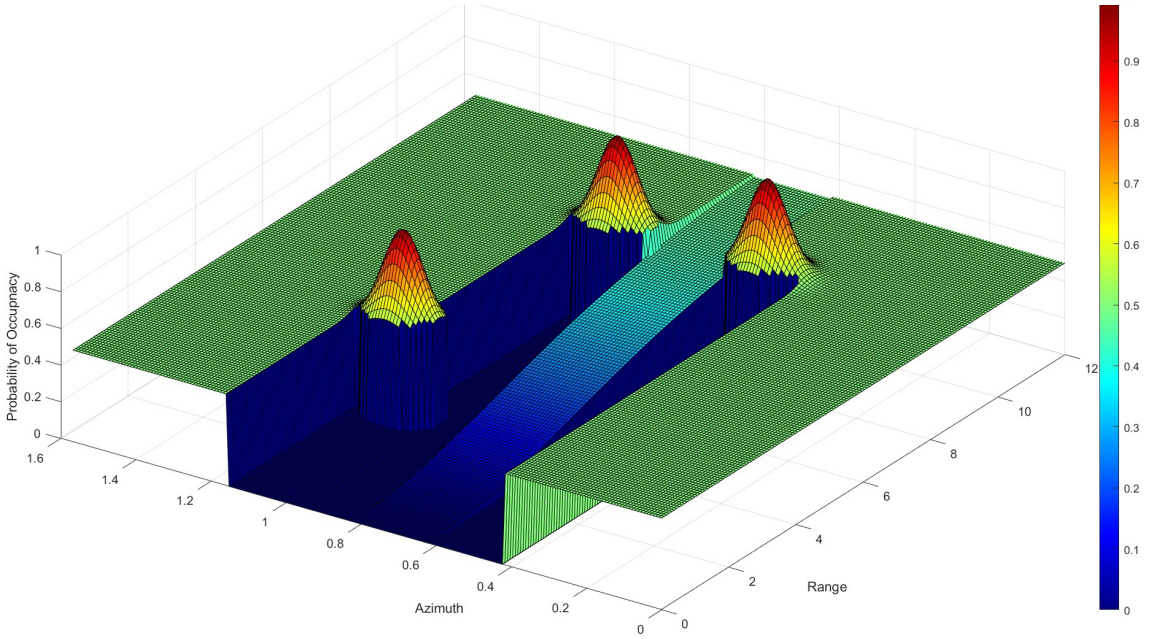
$r \in [0, 12]$ and $\theta \in [0, \pi/2]$. In this scenario, three targets are located at $(r, \theta) \in \{(4, \pi/3), (9, \pi/3.4), (9, \pi/6)\}$. Initially, the algorithm identifies the empty spaces from the ego vehicle to all targets, as well as the empty area between the two at $r = 9$, which are colored dark blue. Then, the occupancy probabilistic distribution affects the state of the cells around the detected targets to produce the final map. Furthermore, it can be observed that the second target located at $(9, \pi/3.4)$ is partially masked by the closer object at $(4, \pi/3)$. Despite the occlusion, the angular interval of $\theta \in \{(\pi/3, \pi/3.4)\}$ is deemed empty in the range of $r \in \{(4, 9)\}$ due to the characteristics of Radar sensors embedded into the ISM module. Additionally, the exponential increase in occupancy probability with a range in the free area between the two targets at $r = 9$ is evident.

Adaptive Height Augmentation

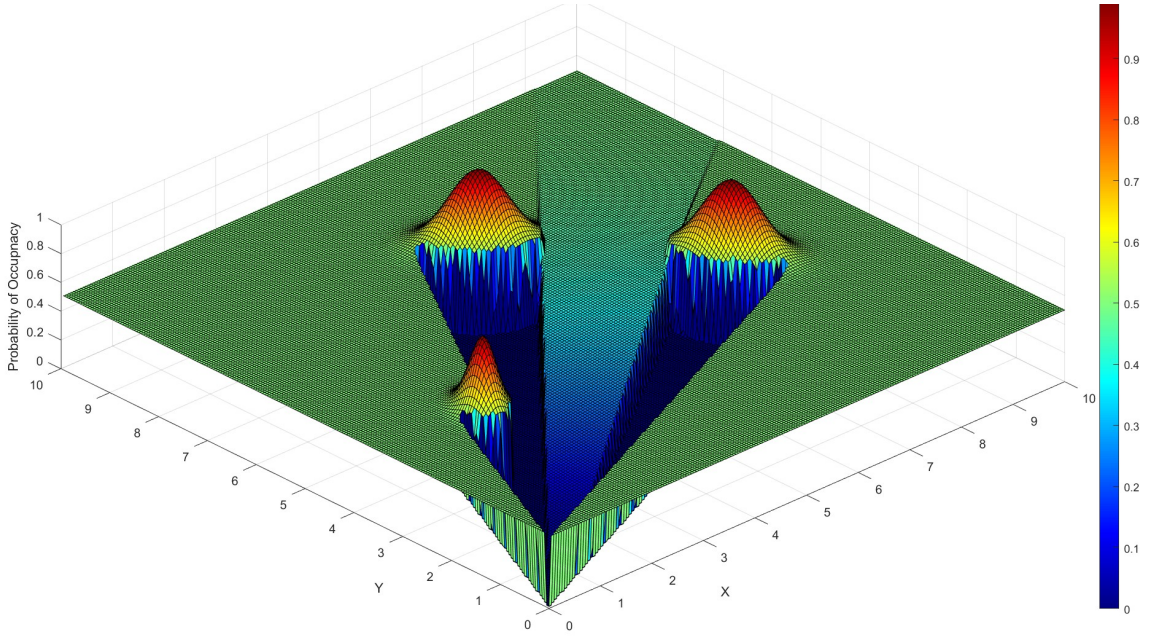
In our approach, we implement an adaptive height augmentation technique, extending the 2D map to a semi-3D representation. This involves generating a two-slice occupancy map where each grid cell is assigned a height value leveraging the distinction between the occupied and empty areas. For cells identified as occupied, we assign a height value corresponding to the reference point elevation from the ground, to give a realistic height dimension to objects. Conversely, cells marked as unoccupied are assigned the height of zero, at road level, indicating the absence of any obstacles or objects. This adaptive height assignment enhances the spatial understanding of the environment.

4.3.2 Static and Dynamic Objects Mapping

Traditional occupancy grid mapping algorithms assume that all entities in the environment are static and that no changes occur during the mapping process. However, in real-world driving scenarios, both static and dynamic objects are present and should be taken into account. Static objects are stationary and remain at the same location with respect to the global frame, whereas the location of dynamic targets changes over time. Therefore, a cell deemed occupied at one moment could be free at the next. Consequently, our system, first, separates static and dynamic objects and performs mapping on each set individually to deal with dynamic targets accordingly. One advantage of Radar sensors is the ability to simultaneously measure the location and the velocity of the targets. Therefore, based on the velocity of the detected objects and the speed of the ego vehicle, acquired from the odometry unit, the state of the target can be identified, thereby static and dynamic instances can be differentiated.



(a) Polar Coordinates



(b) Cartesian Coordinates

Figure 4.3: This figure shows an example of an occupancy map of three targets located at $(r, \theta) \in \{(4, \pi/3), (9, \pi/3.4), (9, \pi/6)\}$ in the polar and Cartesian coordinates. The reference view is located at the origin, $(r, \theta) = (0, 0)$.

Having separated the dynamic and static object points, the two frames undergo regular mapping procedures individually to build the local occupancy grids. At $t = 0$, both maps are stored in a memory to be used in future steps. Then, the dynamic map is superimposed on the static grid, generating the final output for the first iteration. For the subsequent interactions, $t > 0$, mapping is done for static and dynamic targets separately, at first. Then, following the filter update rule expressed in eq. 4.11, the static and dynamic stored from the previous iteration are recalled to be combined with current time results. On the dynamic objects pipeline, we embedded an exponential forgetting factor that enables the system to handle these objects and account for their impact on the occupancy state of the neighbor cells. This factor is formulated as eq.4.14 in which $\tau[s^{-1}]$ is the time constant and $\alpha[s]$ is the inverse of it that specifies the forgetting rate; the larger the α becomes the quicker the decay occurs.

$$\begin{aligned} & e^{-\frac{\Delta t}{\tau}} \\ & \alpha = \tfrac{1}{\tau} \end{aligned} \tag{4.14}$$

The forgetting factor applies to the dynamic occupancy map from the previous sample, shown in eq. 4.15, and gradually attenuates the probability of occupancy of the cells at each iteration until the initial uncertain state is regained, assuming no new detection occurs in that location. Consequently, the time constant needs to be adjusted accordingly. If the values are too small, the past impact of the moving entities will vanish quickly, whereas a large forgetting factor avoids appropriate temporal adaption to the new measurements, leaving an interfering mark of outdated samples.

$$l_{0:t}(c_{m,n}^{dyn}) = l_t(c_{m,n}^{dyn}) + e^{-\alpha \Delta t} l_{0:t-1}(c_{m,n}^{dyn}) \tag{4.15}$$

An illustration of dynamic object handling is provided in figure 4.4. The figure shows the same environment in four consecutive time samples where the blue, green, and red colors refer to empty space(lowest occupancy probability), uncertain areas, and the space occupied by objects, respectively. The second row shows magnified images of the areas of interest where a dynamic object, a car, is present. The current and initial location of the object of interest is highlighted by a cross mark and a rectangle. As observed, the object is moving forward toward the top of the image, which is specified with a red arrow. Additionally, it can be seen that the area between the ego vehicle and this object is colored dark blue, referring to the empty spaces, as no other objects were detected in the middle. Furthermore, the area behind the object of interest is deemed as uncertain, as no other objects were detected at the back. As the location of this object changes through time, a yellow-green mark is left at its previous location that vanishes completely at $t = 3$. In this scenario, the α is set to $ln(0.1)/3$ and $\Delta t = 1$ which attenuates the impact of the presence of the object at its initial place by a factor of 0.1 during three-time samples. In other words, the probability of occupancy at the initial location does not change to 50% instantly, but it decreases gradually through time to complete uncertainty, depending on the α parameter.

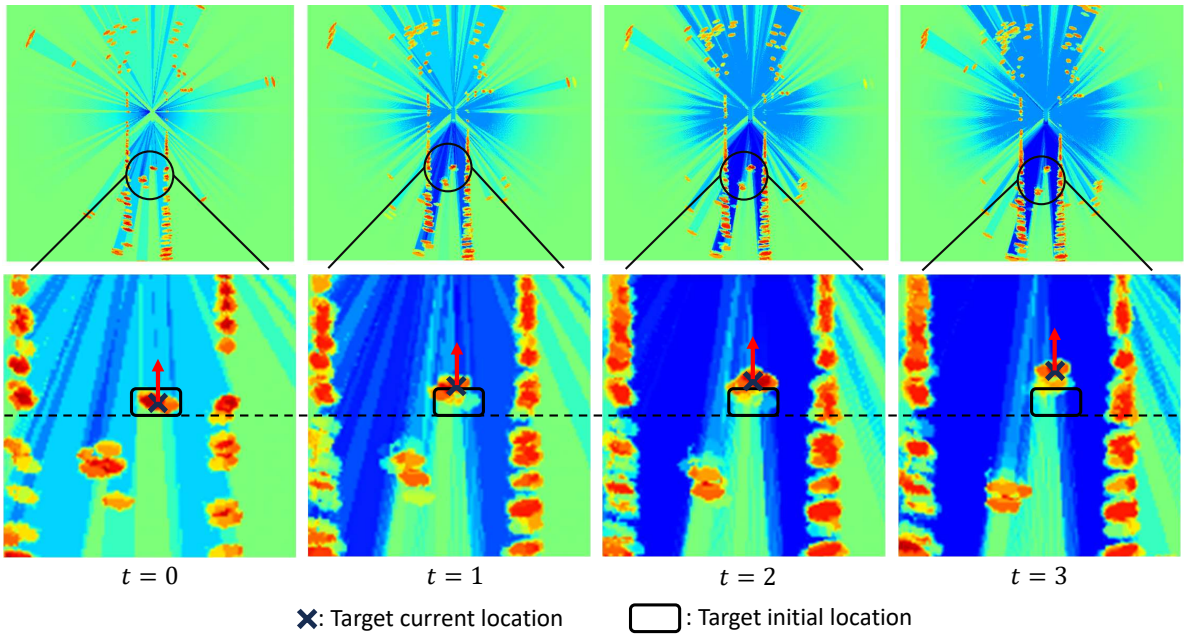


Figure 4.4: This figure shows four consecutive samples from a scene in the NuScenes dataset. The scene contains the ego vehicle and surrounding objects which are either in a stationary or dynamic state. The first row includes the complete view of the maps. The second row focuses on an object of interest, a car that is moving forward toward the ego vehicle. The changes in the location of this object are accounted for with the forgetting factor utilized to handle dynamic objects in mapping a driving scenario.

4.3.3 Multi-Sensor Fusion and Motion Compensation

Different vehicles and platforms within the research community use their own specific setup and sensor configuration. This variation is also reflected in the available datasets for perception purposes. For instance, the Pixset dataset [121] employs only one front-view Radar, while the NuScenes dataset [80], uses five Radars mounted on the front left, middle, and right, back left and back right of the platform vehicle. On the other hand, the grid mapping filter in eq. 4.11 was originally developed given measurements from a single sensor. We generalize our system adaptability to different configurations by implementing a multi-sensor fusion module that unifies the recorder data from different views, thereby making our system flexible in handling different sensor configurations at an arbitrary number.

Generally, each sensor tailors measurements in its local coordinates. Consequently, if the filtering algorithm were to naively process the local measurements, it would result in an incorrect mapping of the scene. To address this issue, we utilize the intrinsic matrices of each sensor and transform the local points into a unified view of the vehicle to generate a consistent representation of the surroundings. The top center of the vehicle is a suitable candidate for this purpose which enforces a symmetric view from all angles and enables

a consistent obstacle-free view of the surroundings. We call this point the *Reference Coordinates of View* or for short, the reference view.

To arrange the data acquired by all Radar channels into the reference view, we utilize the intrinsic parameters of each channel with respect to ego coordinates. Generally, ego coordinates are placed at the center of the vehicle, but if that is not the case, each point is transformed into ego coordinates and then to the reference view. This process is expressed in 4.16 where ${P_i}^{rch_k}$ is an arbitrary 3D point measured by k^{th} Radar channel (rch_k) which when transformed to reference view, is denoted by P_i^{ref}. Furthermore, $T_{rch_k}^{ego}$ is the transformation matrix from k_{th} Radar channel to ego view, which comprises of rotation matrix R and translation vector t of the Radar channel with respect to the ego coordinates. Similarly, the same transformation matrix can be derived for the ego to reference coordinates. The R and t matrices are called sensor intrinsic matrices typically reported in the dataset along with sensor configuration.

Having formed the two mentioned transformation matrices, points can be transformed to the reference view from arbitrary coordinates with known intrinsic following the process shown in eq. 4.17. After transformation, the ISM functions apply to the new points and identify the state of the cells in the local grid.

$$
\begin{aligned}
&{P_i}^{rch_k} = [x_{r_k,i}, y_{r_k,i}, z_{r_k,i}, 1]^T,\ P_i^{ref} = [x_{ref,i}, y_{ref,i}, z_{ref,i}, 1]^T \\
&T_{rch_k}^{ego} = \begin{bmatrix} R_{3\times3}^{rch_k} & t_{3\times1}^{rch_k} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}_{4\times4} \qquad (4.16) \\
&P_i^{ref} = T_{ego}^{ref} \,.\, T_{rch_k}^{ego} \,.\, {P_i}^{rch_k} \qquad (4.17)
\end{aligned}
$$

The filtering algorithm constructs a local grid map and updates the grid cell states from each input frame based on current-time measurements and the previous maps. Additionally, the ego vehicle has different states at different times, such as stationary, moving, making a turn, etc. Therefore, the vehicle motion and previous positions should be taken into account while building the present-time map. At time $t-1$ the local grid map is constructed relative to the vehicle's position (x_{t-1}, y_{t-1}). To ensure correct updating of the cell states at time t, the previous states must be correctly aligned with the cells corresponding to the vehicle's current position at (x_t, y_t). This process, known as motion compensation, follows a similar procedure mentioned previously to re-associate the previous cell states and reflect the viewpoint of the vehicle at its current location.

In this process, our system PARMG utilizes the rotation and translation matrices of the reference, the ego vehicle, and the global coordinates to build the relevant transformation matrices. The vehicle motion is captured by the changes in its position relative to the global frame, making the term T_{ego}^{global} a function of time. Therefore, to report the new location of the previous grid cell in the current-time view, they should be transformed to the global view, shown in eq. 4.18. In this transformation chain, each point P_i^{ref} in reference coordinates at time $t-1$ is first transformed to ego view, using ref-to-ego intrinsics, and then from ego to global view to find their true locations, using ego-to-global

intrinsics at time t denoted by $T_{ego}^{global}(t-1)$. The true location of the cells is the same in the global view and does not change over time, shown by eq. 4.19. Thus, the points are transformed back to the ego frame by utilizing the inverse of the ego-to-global transformation matrix and back to the reference view using the ref-to-ego matrix, expressed in eq. 4.20.

$$P_{t-1}^{global} = T_{ego}^{global}(t-1) \,.\, T_{ref}^{ego} \,.\, P_{t-1}^{ref} \tag{4.18}$$

$$P_{t}^{global} = P_{t-1}^{global} \;\; \forall \;\; cells \tag{4.19}$$

$$T_{ego}^{ref} = (T_{ref}^{ego})^{-1}, \; T_{global}^{ego} = (T_{ego}^{global})^{-1}$$

$$P_{t}^{ref} = T_{ego}^{ref} \,.\, T_{global}^{ego}(t) \,.\, P_{t}^{global} \tag{4.20}$$

Finally, at time t, the correct location of the cells derived in the previous time step is determined to ensure cells included in the new local grid map correctly represent the environment. Finally, the state of the grid cell is updated according to the ISM procedures, incorporating the latest sensor measurements and motion-compensated transformations.

4.4 Conclusion

In this chapter, we discussed our proposed system designed to enhance the quality and availability of Radar measurements in the perception datasets. By employing Bayesian filtering and a Radar-specific ISM, our system generates an occupancy grid representation of the environment around the ego vehicle mimicking the behavior of the analog azimuth-range Radar scans. Consequently, the OGMs contain fine structural details of the surroundings, as opposed to sparse point clouds, while accounting for both static and dynamic objects. The capability of our system to support multi-sensor configurations extends its accessibility to other datasets and different applications. In the next chapter, we present the qualitative evaluation of the capabilities of the framework in constructing high-quality maps and in runtime optimization. Additionally, we will assess its effectiveness in enhancing the robustness of a benchmark object detection model under adverse foggy weather conditions.

Chapter 5

Experiments

In this section, we analyze the performance of our proposed system, namely PARMG, from three different aspects. We, first, delve deep into the properties of the generated grid maps by (1) examining the effect of grid cell size on the resolution and the quality of the maps, (2) investigating the system capabilities from an information-theoretic aspect at the cell state level, (3) the effectiveness of our approach in tackling the extreme sparsity issue in the Radar point clouds followed by (4) a qualitative assessment through entropy metric on the amount of uncertainty eliminated in the occupancy state of the grid cell. Second, we evaluate the runtime optimization of PARMG by integrating a multi-processing scheme into the system, mainly aimed at enhancing the runtime of offline high-quality map construction. Finally, we demonstrate the Radar sensor's capabilities in tackling foggy weather conditions by evaluating the impact of adding the occupancy maps to the input of a baseline object detection on the model's robustness.

5.1 Overview of Experimental Dataset

For this research, we use the NuScenes dataset [80] that offers a large-scale and comprehensive collection of multi-modal recordings. The data is gathered in the urban environments of Boston and Singapore to capture their heavy and complicated traffic conditions. It features an all-encompassing sensor suite including 6 cameras, 5 Radar, and 1 LiDAR sensor providing 360-degree coverage in all modalities, as well as GPS and IMU data for accurate localization and positioning information. The platform vehicle used for data collection is depicted in figure 5.1 which illustrates the placement of sensors. Each sensor tailors measurements in its local coordinate which can be transformed to different views using their intrinsic matrices. The xy-axis of the local frame of each sensor is specified on the vehicle with respect to the global coordinates. The ego-vehicle view refers to the IMU sensor position where the kinematics of the vehicle such as ego positions and acceleration are measured.

This dataset offers around one thousand 20-second video snippets called scenes, each further broken into keyframe and intermediate-frame samples, also referred to as sweeps.

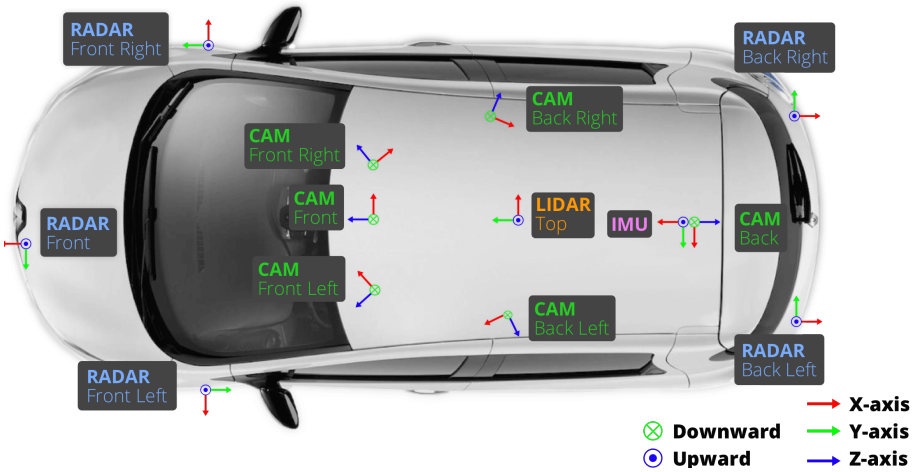


Figure 5.1: Overview of the Nuscene's platform vehicle. This image is taken from the original paper [80].

Keyframe samples are fully annotated over 23 different object categories whereas the sweeps, which are about ten times larger in volume, provide supporting data and enable a smooth transition from one keyframe to the next. The Radar portion of the dataset consists of recordings in the format of point clouds, describing a two-dimensional slice of the environment with up to 18 features appended to each point. For this research, we utilize ego-vehicle positioning and object dynamic information for occupancy mapping purposes and quality assessment entries for filtering invalid and noisy measurements. The technical specifications of the Radar sensor, provided by the publisher [278], are used to configure the hyperparameters of the ISM module in PARMG. A brief description of the Radar data entries used during the experiments is provided below.

- (x, y, z): 3D position of the measured target, where the height is always set to zero.
- RCS: Radar Cross Section of objects in dBsm (dB per squared meter), indicating their detectability. This parameter is affected by the material and size of the target as well as the wavelength and the angle of incident of the emitted beams.
- Dynamic properties: State of the target cluster, categorized as moving, stationary, oncoming, stationary candidate, unknown, crossing stationary, crossing moving, and stopped.
- State ambiguity: The uncertainty in determining the state of an object based on the Doppler response and radial velocity.
- False alarm: Probability of false alarm detection of a cluster, accounting for noise and artifact caused by multipath effect, etc.

5.2 The Properties of the Occupancy Grid Maps

In this section, we explore the quality and information-related characteristics of the Occupancy Grid Maps (OGMs) generated by PARMG. We assess the performance and the capabilities of our framework in producing suitable maps for perception-related tasks. In the first experiment, we evaluate the quality of the generated maps at different resolutions by two quality assessment metrics: Perceptual Image Quality Evaluator (PIQE) and Structural Similarity Index (SSIM). This experiment demonstrates the capability of our system to record and preserve fine structural details in a driving scenario that cannot be directly inferred from Radar point clouds.

Next, we assess PARMG's capabilities from an information-theoretic angle. Initially, we analyze the constructed maps at the cell level, to investigate the evolution of the state of the grid cells and how they contribute to creating more informative maps. Following this, we inspect the effectiveness of our approach in addressing the issue of Sparsity in the Radar point clouds. We show that the average number of Radar points per frame in the Nuscenes dataset is about 122.6, which is an extremely inadequate number for reliable perception. Therefore, we test the ability of our framework to reduce this sparsity and enhance data density. Lastly, we employ the Entropy metric to assess the capability of our system to generate a rich representation of the surroundings by reducing the ambiguity of the state of grid cells in the area and increasing the predictability of the situation for more reliable navigation.

5.2.1 Resolution and Map Quality Evaluation

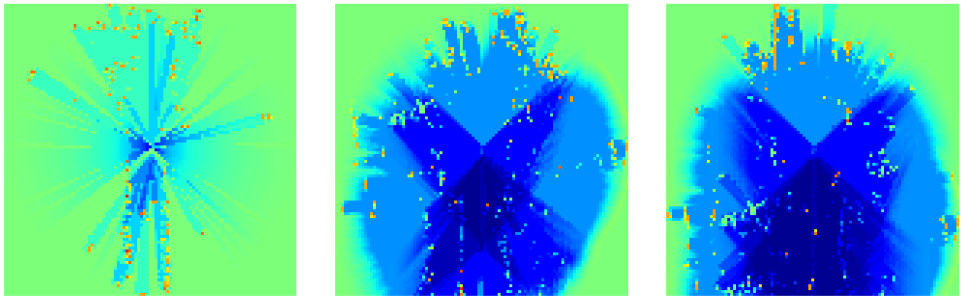
The purpose of our method was to construct OGMs that are capable of reconstructing and estimating the structural details of a driving scene lost by sparse sampling in the Radar point cloud. Prior research has shown that the quantization error, which occurs when sampling an environment with cells and representing it in a grid format, becomes significant if the cell edges exceed 0.25 meters [290]. This quantization error introduces noise and distortion during the process, degrading the quality and resulting in inaccurate maps.

In this section, we qualitatively assess the impact of cell size on the resolution and the quality of the OGMs. We employ two visual quality metrics, namely, Perception-based Image Quality Evaluator (PIQE) [311] and Structural Similarity Index (SSIM) [324] as the base of our analysis. These algorithms evaluate an input image and assign a numerical score based on its quality, structural consistency, and representation of information. Given the dynamic nature of local OGMs, where maps evolve over time, it is crucial to capture the evolution and flow of information across the time axis. Therefore, we perform our assessment on the quality of generated maps with different cell sizes and compare them at each time step. Specifically, first, we fix map edges to a 100m by 100m area. Then for the mapping scenario, we selected a sequence containing scenes from the ego vehicle passing over a bridge, navigating through populated areas, junctions, and

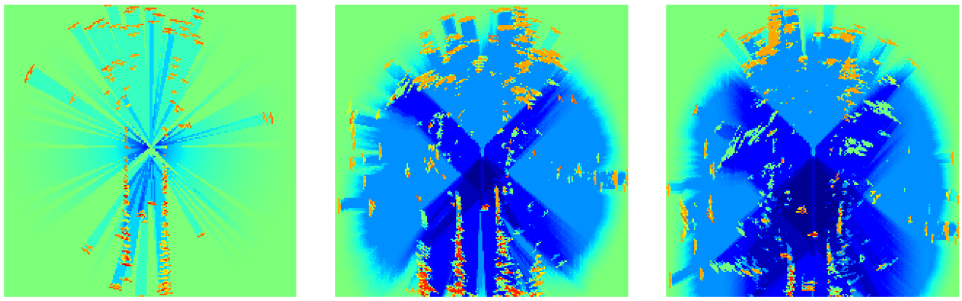
red lights with pedestrians, parked and moving parked cars. For this evaluation, a full sequence of maps is generated with four different resolutions by adjusting the cell edges to 1, 0.5, 0.25, and 0.125 meters. This approach allows us to systematically analyze the trade-offs between resolution and map quality, providing insights into the impact of cell size on the accuracy and details of grid maps.

Figure. 5.2 depicts sample frames of the sequence, selected for this experiment. The top row belongs to the largest cell size and the lowest resolution, and moving downwards to the bottom, the cell size decreases and the resolution improves. The improvement in the quality of the maps and the ability to preserve fine details become evident through visual inspection of the images from top to bottom. As an example, the second column tailors the vehicle navigating on a curved road. As shown, the image in the first row demonstrates the poorest representation of the road structure and the environment. Conversely, the second-row maps, with a cell edge of 0.5 meters, capture the structural context but fail to accurately derive the road's curvature, in contrast to the third and fourth images. On the other hand, the third and the fourth map show significant improvements. As the third-row maps are generated on cell edges of 0.25 meters, slightly above the quantization error threshold, the distortive impacts caused by the quantization process are still visible, compared to the fourth group. This distortion causes a less smooth mapping of the curvature of the road, similar to piece-wise linear approximations, and less detail consistency in the map. Meanwhile, the fourth map, with finer resolution, provides a more accurate and smoother depiction of the road curvature and overall map detail.

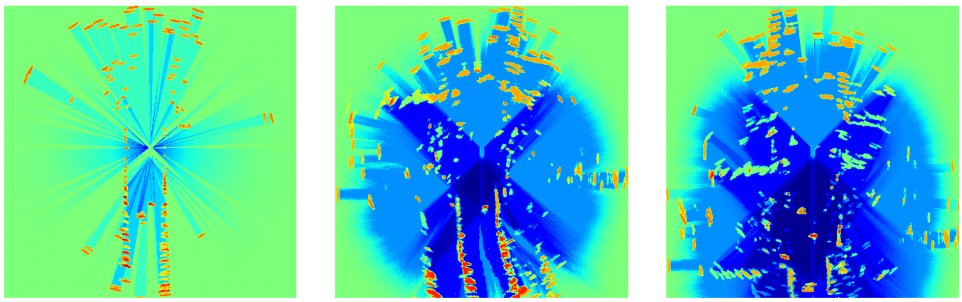
To quantitatively investigate the differences between these group maps, we adopt two quality assessment metrics, PIQE and SSIM, that assign quality-based scores to each image based on allowing us to draw time step-wise comparisons across different resolutions. The first metric, PIQE, is a no-reference method that makes assessments based on perceptual quality factors such as clarity and distortion. This algorithm provides local, block-by-block analysis of the input image and identifies the regions with quality issues. The final quality score is a weighted average of block scores, where a lower score indicates better quality.



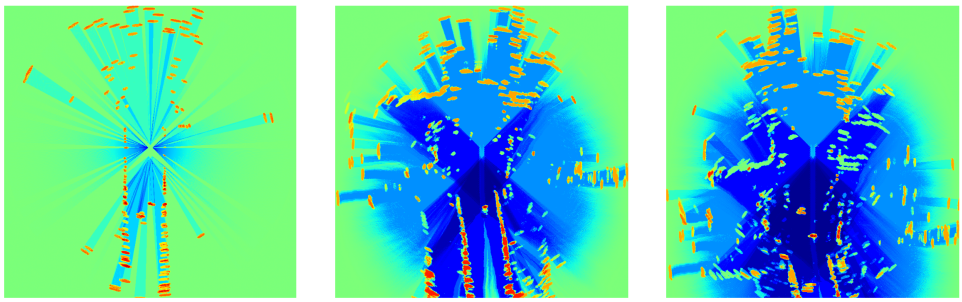
a) Cell edge = 1.0 m



b) Cell edge = 0.5 m



c) Cell edge = 0.25 m



d) Cell edge = 0.125 m

Figure 5.2: This figure shows three samples of a sequence recorded at four different resolutions. The top (a), second (b), third (c), and the last (d) row belong to the grids with the cell size of 1×1, 0.5×0.5, 0.25×0.25, $0.125 \times 0.125 m^2$, respectively, and the columns show three snapshots of the evolution of maps through time, from the first frame to the 70th.

In fig.5.3 the PIQE scores of the map sequences at four resolutions are illustrated. In the beginning, a transient time is evident during which little information is available for map generation and most cells are in an unknown state, showing minimal detail of the environment. As time progresses and the local maps accumulate richer information, they capture more events and contain more details, but differ in the presentation capabilities based on the number of total cells in each grid. Smaller cell sizes, which is equivalent to more cells per grid, are more capable of preserving fine details and therefore result in higher-quality maps, as observed from the figure. Following the transient period the blue curve, corresponding to the largest cell size $1 \times 1m^2$, maintains the highest PIQE score (lowest quality) while the purple curve, corresponding to the smallest cell size $0.125 \times 0.125m^2$, achieves the highest perceptual quality among all keeping the lowest score. According to [281], PIQE scores within the range of [0, 20] indicate images of excellent quality whereas the [81, 100] interval is for images with very poor quality. Consequently, as the cell size decreases, leading to better resolutions, the PIQE score improves from the poor to the higher quality zones. This highlights the flexibility and capability of PARMG to provide a proper mapping of the surroundings depending on the requirements of the applications.

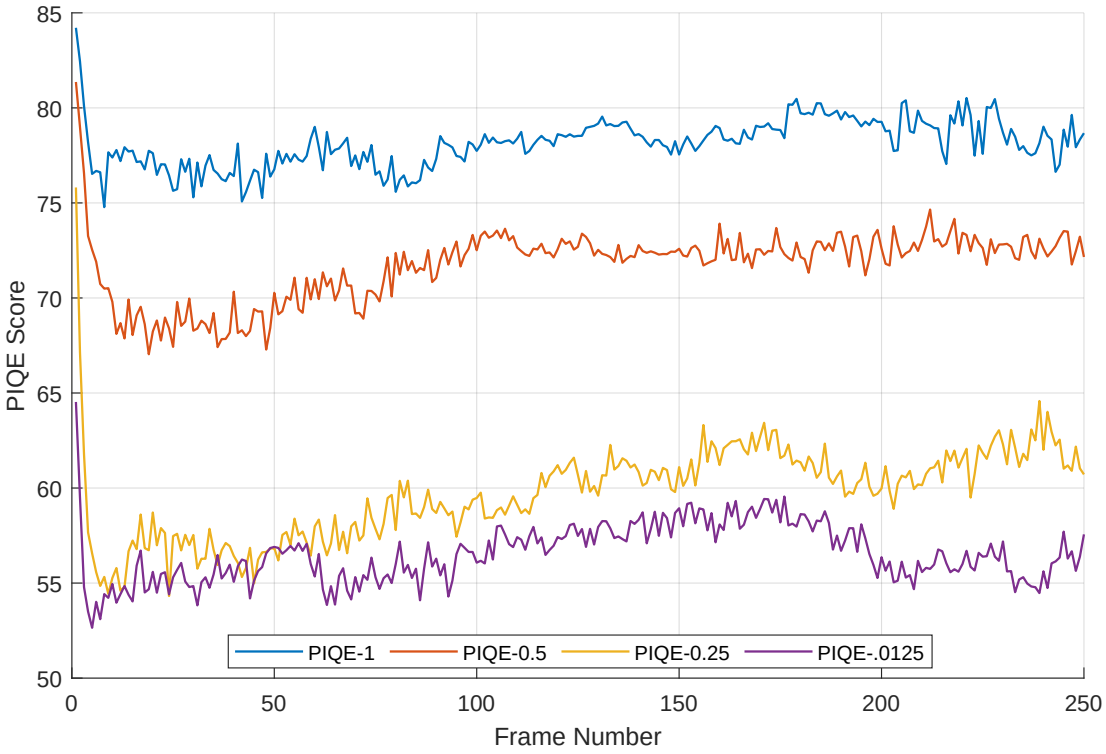


Figure 5.3: Image quality assessment with PIQE metric. The lower score is for images with better perceptual quality.

SSIM is a metric that measures the similarity between two images, one as an input and the other as a reference image. Its design is inspired by human vision's ability to extract structural information and assess similarity based on three different aspects such as luminance, contrast, and structure. The similarity is tailored as a number from zero to one where one indicates identical images and smaller values are for poor qualities. Following the previous assessment, in this experiment, we take the sequence generated at the highest quality as the reference and we compare lower-resolution maps to measure how successful they were in capturing the structure of the surroundings and preserving the details recorded in the high-resolution reference map.

Illustrated in fig. 5.4 is the SSIM scoring at each time sample of map evolution. The second highest resolution map sequence, corresponding to the yellow curve, mostly keeps a score of more than 85%, demonstrating a strong capability to preserve the structural information during mapping. As the cell edges grow larger, the ability to record fine details during the mapping process decreases which is demonstrated by the slump in the overall and temporal SSIM score.

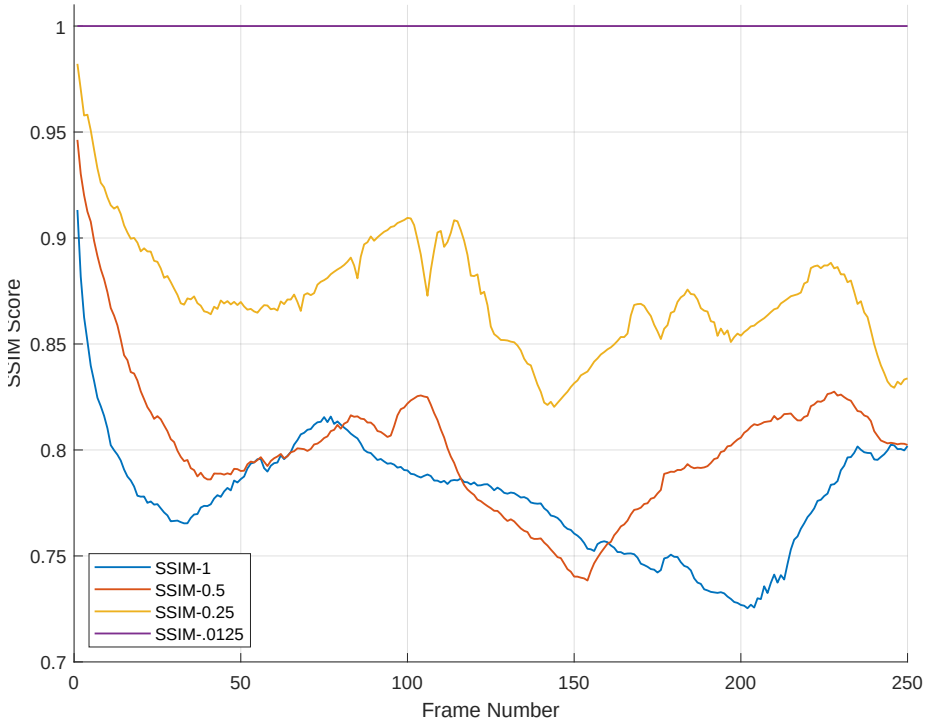


Figure 5.4: Image quality assessment with SSIM metric. The highest score corresponds to the value 1 which belongs to an input identical with the reference image. The lower score indicates less similarity in structural details and image properties.

Conclusively, our evaluation using PIQE and SSIM metrics across different resolutions

demonstrates that smaller cell sizes significantly enhance the quality and detail preservation capabilities of OGMs constructed by PARMG. Higher resolutions consistently yield lower PIQE scores indicating better quality and clarity of details in the images. When comparing lower resolution OGMs to the highest quality maps by SSIM scores, we show the effectiveness of smaller cell sizes in building maps with more preservation of structural information. These findings highlight PARMG's flexibility in adapting to various mapping requirements, ensuring accurate representation of dynamic environments.

5.2.2 Information-Theoretic Evaluation

In the preceding section, we aim to measure the effectiveness of the proposed system, PARMG, in generating enriched and more informative maps. Our study on the Nuscenes dataset indicates that the average number of points per Radar frame is approximately 191.6 with a standard deviation (SD) of 47.7 points for up to a 250-meter range which reduces to an average of 122.6 and SD of 39.9 points for the maximum range of 50 meters, considering our experimental setup. This number is extremely inadequate for the representation of a 100x100m^2 area. The effectiveness of our approach is measured from three different angles, assessing the flow of information and the evolution of the grid cell during the mapping process.

For this set of experiments, we perform occupancy grid mapping on the same scenario from the previous section. We employ PARMG to construct 100 by 100m^2 local OGMs of the surrounding environment from the Radar targets at four different resolutions with cell sizes of 1x1m^2, 0.5x0.5m^2, 0.25x0.25m^2 and 0.125x0.125m^2. First, we provide a cell-level analysis capturing the evolution of the state of cells as well as their temporal contribution in preparing the final map. Second, we examine our framework in addressing the sparsity issue by quantitatively comparing the frames before and after the utilization of PARMG, at each time step and resolution. Lastly, we assess the information gained from one-time sample to another by using the entropy metric and measure the effectiveness of PARMG in reducing the uncertainty in the occupancy state of the environment.

Grid Cells States Analysis

During the evolution of the maps, we conduct a cell-by-cell state analysis at each time step by measuring the number of total cells that fall into one of the three possible states: occupied, empty, and unknown (or uncertain) which are expressed in 5.1.

$$Cell\ states: \begin{cases} Occupied & p(c_{m,n}) > 0.55 \\ Empty & p(c_{m,n}) < 0.45 \\ Unknown & \text{otherwise} \end{cases} \tag{5.1}$$

In Fig. 5.5 the behavior of the states of all grid cells is depicted at four different resolutions. Initially, all the maps start with a high number of unknown cells due to the lack of prior information for the mapping procedure. As the algorithm injects new information

at each time step, the number of cells with uncertain states decreases rapidly and the grids make a quick transition from an overall uncertainty towards a more knowledgeable representation of the environment. Over time the process continues and the reduction in uncertainty settles at approximately one-third of the initial value.

Moreover, recursive processing of estimating the state of the cells and forming the occupancy probability distributions contributes to reducing uncertainty. As the filtering algorithm maps and derives occupied regions from Radar targets, the average number of cells deemed occupied becomes 220, 1900, 8700, and 38000, from the grid with the lowest to highest resolution. These amounts, compared to the average of 122.6 Radar targets per frame mentioned in 5.1, show an information gain of 1.8, 15.3, 70.1, and 306.5 about the occupied area. However, the identification of occupied regions is the second to the most contributing factor in OGM construction. The determination of empty zones, which is guided by the occupied areas, makes the most significant contribution to purging the uncertainty in the grid maps. As illustrated, the number of empty cells grows to meet and exchange places with the total number of uncertain cells, early during the transition period. This transition point characterizes a shift in that state of the map from high uncertainty to a more informative state. It happens at 0.38, 0.41, 0.61, and 1 second, for grids with cell edges of 1, 0.5, 0.25, and 0.125 meters, respectively.

It is notable that each decrease in cell size by half between resolutions results in a quadruple increase in the total number of cells within a fixed grid map. Despite this quadratic increase, our system prevents the transition point from growing proportion between two consecutive resolutions. For instance, moving from cell edges of 1 to 0.125 meters, the total number of cells grows by a factor of 64, however, the mentioned transition point is roughly delayed by a factor of 2.5 from 0.38 to 1 second. This highlights our system's robustness in maintaining efficiency despite the significant increase in cell count at finer resolutions.

Sparsity

One of the major problems with the Radar portion of the available datasets is the inadequate amount of available samples relative to the detection range environmental dimensions. Addressing this issue, our framework aims to reduce sparsity and generate more informative maps. In this experiment, an unknown cell is a representation of a non-existent measurement in that location whereas cells with a known state indicate the presence of information. Accordingly, we reformulate and define the sparsity metric to be the ratio of cells with unknown states over the number of all cells. From eq. 5.1 a cell has an unknown state if the probability of occupancy falls in the interval of $0.45 \leq p(c_{m,n}) \leq 0.55$. Therefore, the sparsity metric can be expressed as shown in eq. 5.2 where $< . >$ is the cardinality of the set, M and N are the height and width of the grid which are equal in our case where a rectangular-shaped grid is used.

$$Sparsity = \frac{\sum < c_{m,n}|_{0.45 \leq p(c_{m,n}) \leq 0.55} >}{M \ . \ N} \ ; \ M = N \tag{5.2}$$

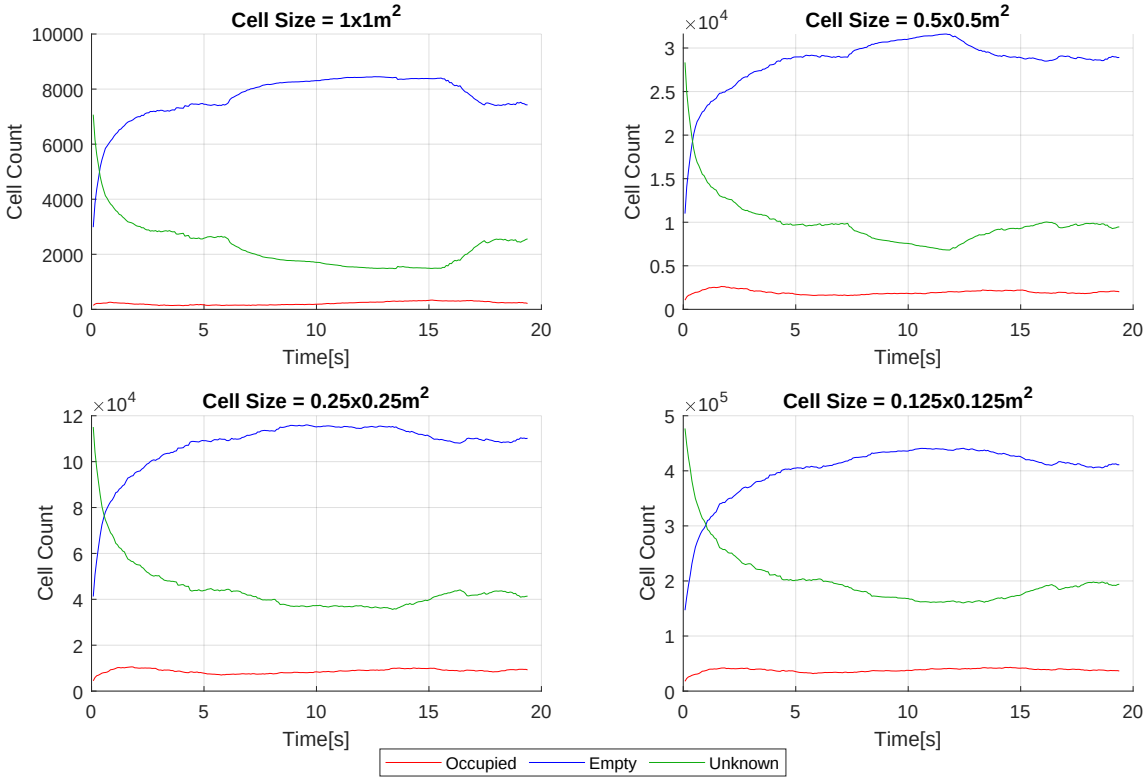


Figure 5.5: Cell-by-cell State Analysis. In this figure, the red, blue, and green curves belong to the number of cells with occupied, empty, and unknown states, respectively.

Figure 5.6 shows the measured sparsity of the grid cells at each time step across the four different resolutions. Prior to the mapping procedure ($t < 0$), with only 122.6 average Radar targets projected on the grid cells, a sparsity of more than 98.7% exists in our scenarios. In the first frame constructed by PARMG at $t = 0$, an instant drop of sparsity to less than 75% is evident in fig 5.6. Despite the initial uncertainty in the states of many cells, the first iteration identifies the initial occupied and empty zones contributing to this more than 25% reduction. The rapid increase in the identified zones continues to the $t = 2$ seconds where an overall of more than 60% reduction in sparsity is achieved. Beyond $t > 2$, as more cell states are identified, the sparsity gradually converges, experiencing fluctuation influenced by local adjustments made by the filtering algorithm.

Among the four resolutions, the lowest-resolution map exhibits the lowest percentage of sparsity whereas the highest-resolution map shows a higher percentage. It is noteworthy that higher grid cell counts require more processing to reduce sparsity. In this experiment, from a lower resolution to the next higher, the total number of cells increases by a factor of 4. Despite the highest sparsity occurring in the grid map built on 0.125x0.125 m^2 cells, with the highest total number of cells, this grid map successfully

maintains a maximum of 5% gap between itself and the adjacent-resolution map. For instance, at $t = 4$ seconds, the yellow curve, with 16×10^4 cells, stays at 30% sparsity whereas the purple curve, with 64×10^4 cells, stays nearly at 35%. This is the case where a 300% increase in the number of total cells resulted in a maximum of 5% increase in temporal sparsity. This highlights the effectiveness of our system in information mining and tackling the sparsity issue and its robustness for high-resolution maps.

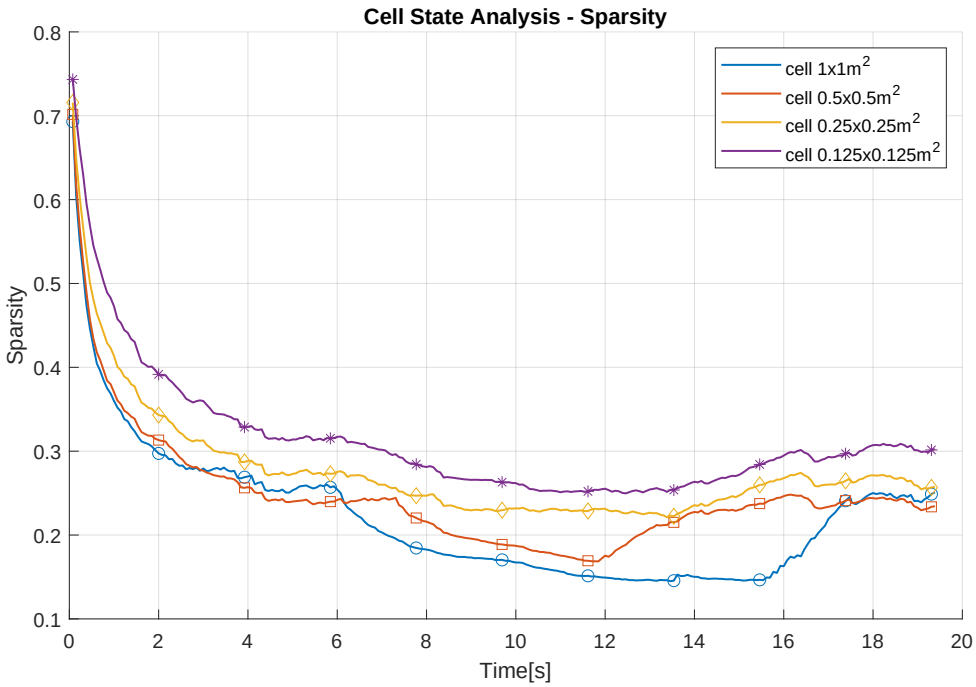


Figure 5.6: The sparsity of $100 \times 100m^2$ grid maps at the cell sizes of 1x1, 0.5x0.5, 0.25x0.25, and 0.125x$0.125m^2$ constructed by the PARMG framework on the average of 122.6 point measurements.

Entropy

Entropy, in information theory, is a metric for measuring the surprise or uncertainty of the occurrence of a probabilistic event. The higher the entropy, the more difficult it is to predict the outcome. For safe navigation of an AV through a driving scenario, the predictability of the environment is of high importance. Therefore, the metric is a suitable tool to measure the amount of uncertainty at each iteration and evaluate the information gained from one frame to the next. In this section, we will quantify what we previously called constructing more "informative" maps.

Given a discrete random variable X with N different states, the entropy is defined as

below.

$$H(X) = -\sum_{i=1}^{N} p(x_i) log_2(p(x_i)) \tag{5.3}$$

Knowing that the state of each grid cell is a binary random variable, where "1" represents occupancy and "0" represents emptiness, and abiding by the independence assumption from the previous chapter the entropy of each cell can be expressed as shown in eq. 5.4 in which $p(c_{m,n} = occ)$ is the probability of the cell $c_{m,n}$ being occupied.

$$\begin{aligned} H(C_{m,n}) = &-p(c_{m,n} = occ|z_t, x_t) \; . \; log_2(p(c_{m,n} = occ|z_t, x_t)) \\ &- [1 - p(c_{m,n} = occ|z_t, x_t)] \; . \; log_2(1 - p(c_{m,n} = occ|z_t, x_t)) \end{aligned} \tag{5.4}$$

Thus, the total entropy of the grid map can be measured by summing up the local entropy of each cell, shown in 5.5, where a division by the grid dimensions, $M \times N$, is performed to achieve a normalized range.

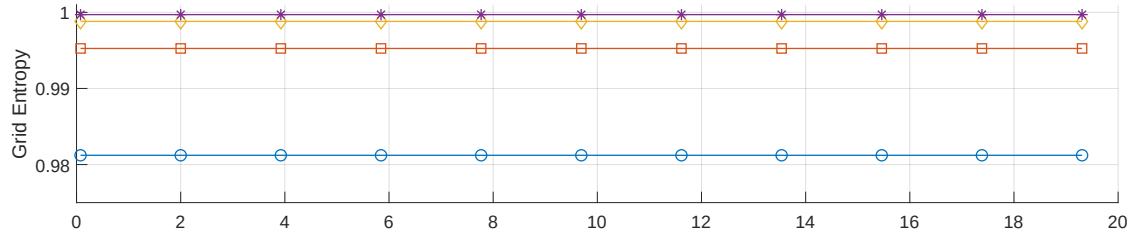
$$\bar{H}(C) = \frac{1}{M.N} \sum_{m,n \in grid} H(C_{m,n}) \; ; \; M = N \tag{5.5}$$

To avoid numerical instabilities during this experiment, we exclude the points with a probability of occupancy of zero and one. Similar to the previous experiment, we project the Radar targets onto the fixed grid maps and measure the entropy of the raw point clouds. Next. we filter out the points that do not belong to the probability interval mentioned above. Subsequently, all Radar measurements will be filtered, as these points represent 100% occupancy probability, and all the remaining cells of the grid will be included in entropy calculations. Given the states of all these cells are unknown and there is a 50% chance of both being occupied or empty, the entropy reaches values extremely close to one. Assuming that at each iteration an average of 122.6 point measurements are observed, the entropy of raw point clouds on the four grid cells are 0.9812, 0.9952, 0.9988, and 0.9997, corresponding to the lowest to the highest-resolution map, illustrated in Fig. 5.7a. It is evident that as the total cell count increases while maintaining the same number of Radar point measurements, the prediction of occupancy in the mapping area becomes more complex reflected by the entropy growing closer to one.

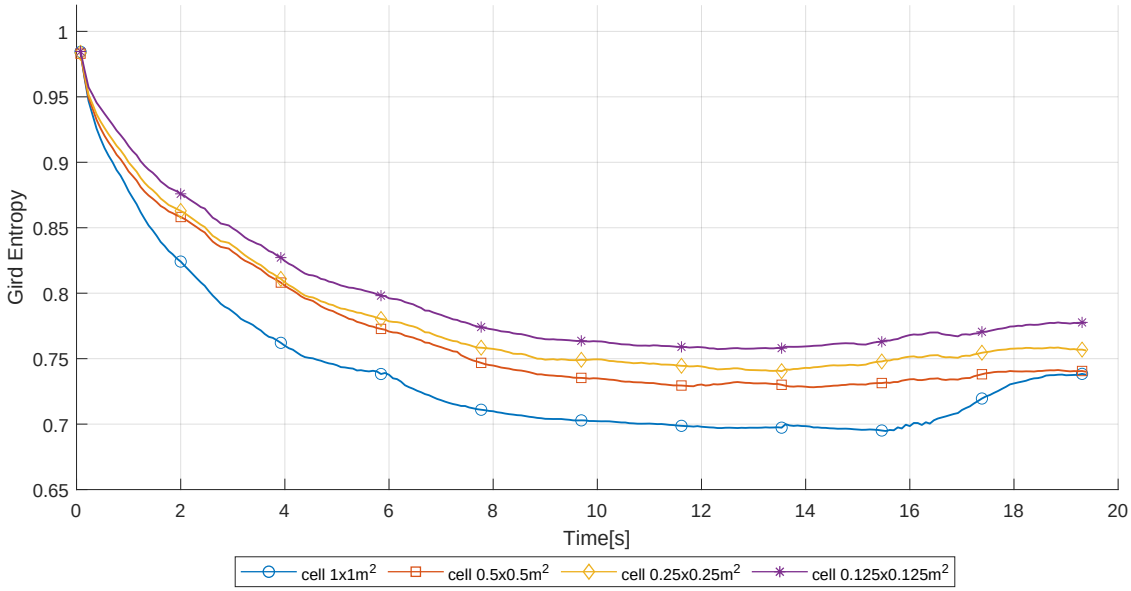
On the Other hand, our proposed framework (PARMG) benefits from the temporal redundancy of the data and generates the occupancy maps at each timestep utilizing the prior information from the previous frame along with the new measurement to update occupancy probabilities for all grid cells. Figure 5.7b depicts the entropy measurements of the four grid maps. It is observed that as the algorithm runs, our framework continuously decreases the grid entropy by more than 22% after convergence. The reduction in entropy is equal to the decrease in uncertainty, which indicates information gained at each time step.

Additionally, as the grid cell sizes shrink to generate maps at higher resolution, decreasing the uncertainty becomes quadratically more complicated. However, in our experiments, where the cell counts are quadrupled from the previous grid to achieve the

next higher resolution, only a maximum 3% increase in entropy can be observed among the highest-resolution maps. Conclusively, this experiment highlights the effectiveness of our framework in constructing informative representations of a local environment and its stability in managing uncertainty at higher resolutions.



(a) Grid entropy of Radar point cloud.



(b) Grid entropy of occupancy maps.

Figure 5.7: Illustration of grid entropy measurements (a) with and (b) without using the PARMG framework. The blue, orange, yellow, and purple curves belong to the grids with cell sizes of 1x1, 0.5x0.5, 0.25x0.25, and 0.125x0.125m^2.

5.3 Runtime Performance

In the previous section, we explored the impact of cell size and resolution on the quality and preservation of fine structural details in the grid maps. While smaller cell sizes are required for higher-quality maps, the mapping procedure is challenged by the cost

of processing time. To tackle this challenge, we implement a multi-processing scheme to optimize the runtime of our system for high-quality map generation by constructing maps of different scenarios in parallel instead of sequentially. This scheme is mostly beneficial for offline map generation use. In this section, we analyze the processing time of the PARMG system, with and without utilizing multi-processing, and investigate how changes in the cell count influence the runtime. Furthermore, we explore the contribution of our multi-processing scheme to address the production time challenges.

For this experiment, we generate five fixed-dimension 100 by 100 m^2 grid maps with cell edges of 1, 0.5, 0.25, 0.166, and 0.125 meters resulting in grids of various cell counts (101x101, 201x201, 401x401, 601x601 and 801x801). We measure the processing time required for generating 50 grids and take the average to account for the fluctuations in map production happening due to the time-varying nature of Radar point cloud measurements. We deploy the multi-processing scheme and compare the contribution of utilizing 2, 4, 8, 12, and 16 parallel processes on optimizing map construction. This experiment was conducted on Ubuntu 22.04.3 LTS using the hardware equipped with Intel(R) Core(TM) i9-10900X CPU, a clock speed of 3.70GHz, 10 cores, and 2 threads per core. The details results of the experiment are provided in table 5.1 where each column shows the processing time for different resolutions corresponding to the number of process instances specified in each row.

Cell edge [m]	1.0	0.5	0.25	0.166	0.125
Grid cells	101x101	201x201	401x401	601x601	801x801
1 process	4.93	13.72	45.38	98.33	166.54
2 processes	2.67 (46%)	7.28 (47%)	22.98 (49%)	49.09 (50%)	83.71 (50%)
4 processes	1.38 (72%)	3.81 (72%)	12.24 (73%)	27.16 (72%)	47.54 (71%)
8 processes	0.74 (85%)	2.20 (84%)	6.50 (86%)	14.39 (85%)	25.41 (85%)
12 processes	0.64 (87%)	1.98 (85%)	5.78 (87%)	12.26 (87%)	21.31 (87%)
16 processes	0.60 (88%)	1.89 (86%)	5.69 (87%)	11.84 (88%)	20.83 (87%)

Table 5.1: The impact of multiprocessing on the average time of a grid map at different resolutions.

Figure. 5.8 illustrates the average processing time of a grid map with the mentioned cell sizes, utilizing 1, 2, 5, 8, 12, and 16 process instances. It is observed that as the total cell numbers increase towards higher map resolutions, more processing time is required to construct a complete grid map. The longest processing time belongs to the 800 by 800 grid among all resolutions and across different number of process instances. In this figure, the blue curve belongs to the time required for the sequential construction of one full grid. The grid processing time exhibits exponential growth with the increase in cell counts, provided in the first row of table 5.1. Specifically, the processing time is, approximately, in linear relation with total cell counts in the grid. In a fixed grid, as the dimension of the cells is reduced by half, the total amount of cell on the edges

are doubled resulting in a squared total number of cells. Therefore, the grid processing time grows with the exponent of 2. This exponential growth poses a significant increase in processing time for high-resolution maps. To mitigate this, we implemented a multi-processing pipeline that distributes Radar measurement frames to an arbitrary number of processes to construct the OGMs in parallel.

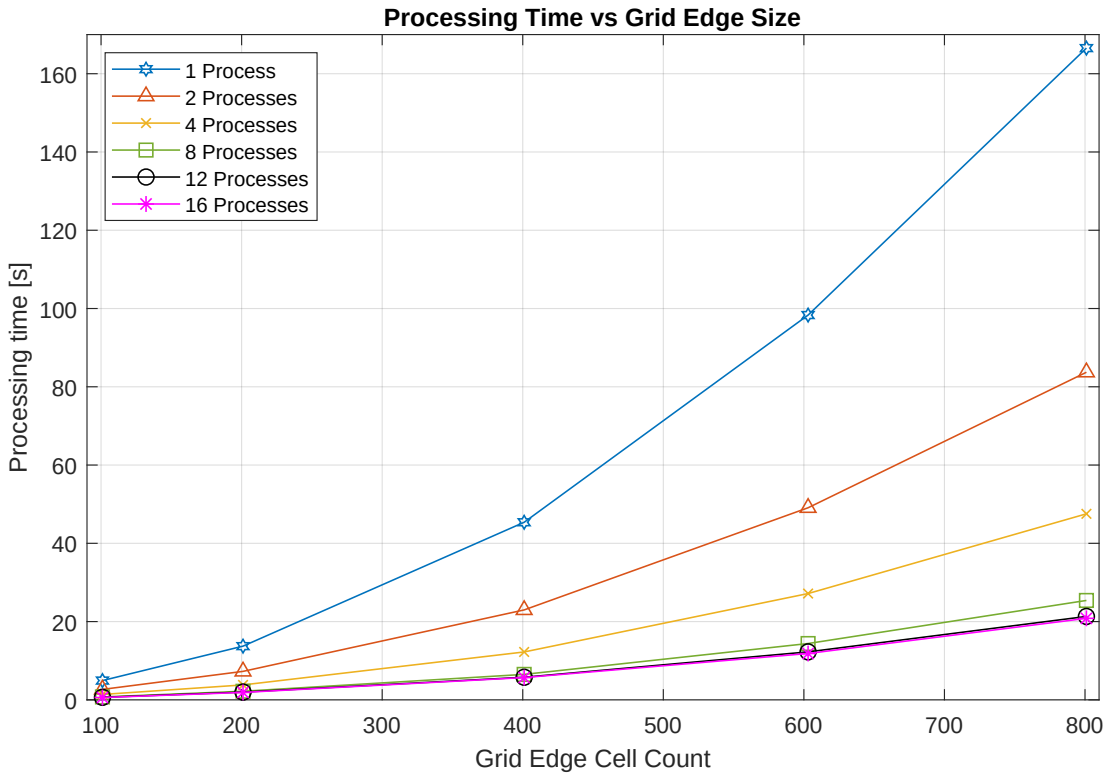


Figure 5.8: Grid processing time vs the number of cells on grid edges.

Utilizing the multi-processing scheme, the system generates multiple maps per iteration resulting in greater time efficiency. For instance, the last column of table 5.1 illustrates that about 166 seconds are required to generate an 800 by 800 map with one process. For the same scenario, employing two, four, and eight individual processes breaks the average production time by approximately a half, a quarter, and an eighth to almost 83, 47, and 25 seconds, respectively. An overview of the contribution of multi-processing is depicted in figure 5.9. It can be inferred that with up to 8 processes, the processing time can be decreased significantly. However, beyond 8 process instances, this optimization reaches a saturated point. Therefore, this indicates a point where an appropriate trade-off is made between time and memory efficiency. Taking the 800 by 800 grid as an example, approximately, 25 seconds is required in an 8-process setting which results in more 84% efficiency compared to sequential map construction. In contrast,

increasing the processes to 12 and 16 only adds up to a maximum of 4% improvement.

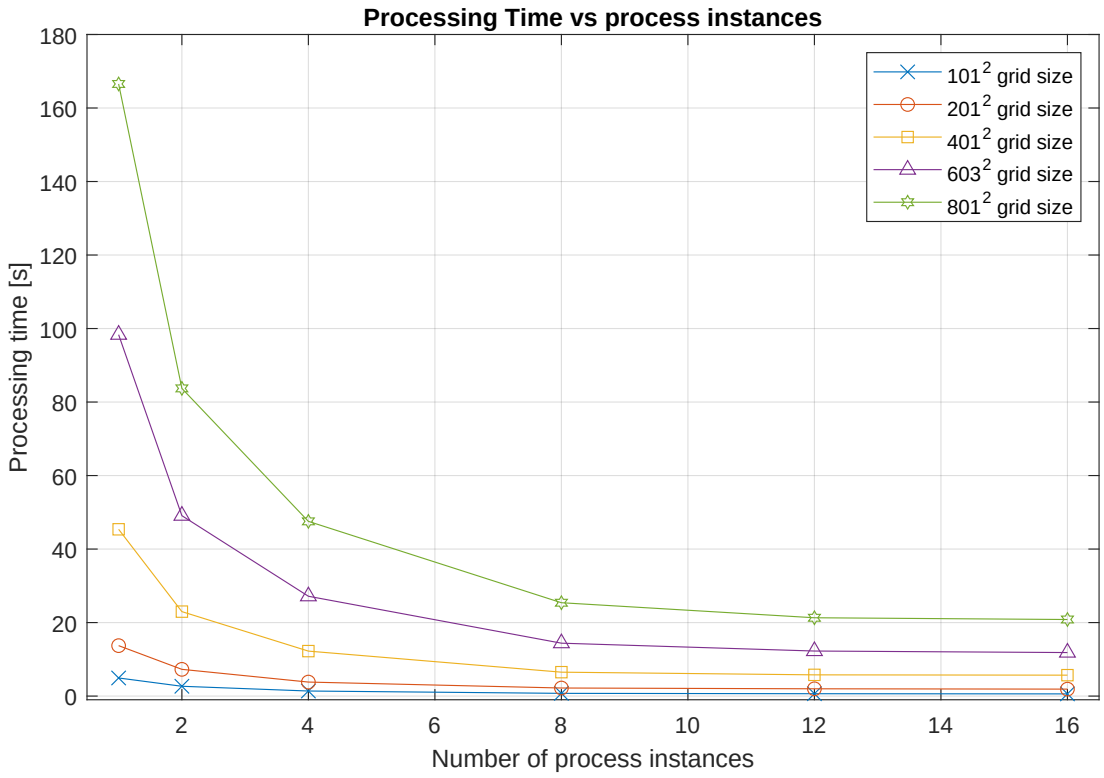


Figure 5.9: The response of average grid processing time to changes in the number of process instances.

In conclusion, our analysis of the PARMG system's runtime performance demonstrates that while higher resolutions enhance map quality and are essential for detail mapping of the environment, they also substantially increase processing time. However, the implementation of a multi-processing scheme enables our system to address this issue, becoming significantly efficient in generating high-resolution maps.

5.4 Robustness Against Adverse Weather Conditions

In this section, we discuss and evaluate the impact of integrating OGMs with LiDAR point clouds on the robustness of a benchmark object detection model against adverse foggy weather. Radar sensors offer several advantages when it comes to challenging weather conditions. Compared to LiDAR and camera sensors, Radar measurements experience less noise and degradation in these situations. The dataset used in this experiment is the same as the one used in previous sections, discussed in 5.1.

First, we discuss the metric used for evaluating the performance and robustness. Second, we provide insight into the experimental setting and the benchmark model. Next, a detailed explanation of the preprocessing steps is given which includes the fog synthesis and sensor synchronization procedures. Lastly, we evaluate the effectiveness of OGMs on the robustness of the aforementioned object detection framework against adverse foggy weather conditions. This experiment will help determine the extent of OGMs in improving performance robustness and reliability under challenging scenarios.

5.4.1 Evaluation Metrics

In the following, we introduce the metric used for evaluating the performance and robustness of an object detection model. The mean Average Precision (mAP) is a well-established performance evaluation metric in the object detection field that measures the average accuracy and precision of a model's predictions. This metric is based on the Average Precision (AP) metric formulated as follows, where TP and FP are for the number of True Positive and False Positive detections, respectively.

$$AP = \frac{TP}{TP + FP} \tag{5.6}$$

In the context of object detection, the NuScenes dataset defines a detection match where the center of the prediction bounding box distance to the center of the ground truth is lower than a threshold. A center-to-center of zero corresponds to a perfect match while greater distances indicate more detection error. The matching thresholds are 0.5, 1, 2, and 4 meters, and the overall mAP is calculated over the number of categories and the matching thresholds. The mAP is expressed in percentage and higher values correspond to better performance.

A model is said to be robust if it maintains high performance against a challenging situation with minimal decrease in its performance. A smaller performance degradation indicates greater robustness of the model. To assess robustness, we evaluate the drop in performance when the model is exposed to various challenging foggified scenarios compared to normal conditions.

5.4.2 Experimental Setting

In order to exploit the benefits of the Radar sensing modality, we test the impact of combining Radar information, OGMs, with the input of a baseline object detection model and analyze the impact of using this sensor on the robustness of the model. We implemented a concatenation scheme for the sensor-fusion module. After the preprocessing of the OGM and LiDAR point clouds, the OGMs are superimposed on top of the LiDAR detections to create the fused dataset. By this method, OGM occupied regions attract more focus on the present objects and obstacles in the environment. Consequently, this will enhance the model's ability to make accurate predictions even

under adverse weather conditions. To measure the effectiveness of the addition of Radar data we use the mAP discussed in 5.4.1.

To prepare a suitable dataset, we chose a set of 50 scenes from the NuScens dataset, totaling over 20,000 keyframes and sweeps of LiDAR and Radar recordings. These scenes include diverse driving scenarios and were recorded in various locations, such as junctions, highways, parking lots, and urban areas with construction sites. The scenes are divided into training and testing subsets following an 80-20% ratio. The LiDAR subset of the dataset is used directly for the foggification process. It is further fused with the OGM and both are used for performance evaluation of the baseline model under the foggy situation.

The baseline model, Pointpillars [184], is a well-known LiDAR-based object detection framework. The model groups the input points into infinite-height pillars and appends an aggregate of their positional and reflectance information to each pillar. This process transforms point clouds into pseudo-images for efficient processing and avoids the computational complexity of 3D convolutions by substituting them with 2D convolution operations on generated pseudo images. Thus the model is a fast, effective, and flexible choice for object detection and suitable for our study.

The hyperparameters of Pointpillar are adjusted by grid search and the official optimal checkpoint [99] is utilized for fine-tuning on both normal weather LiDAR-only and LR-fused sets so that the model adapts to the new sets of data and converges to new relevant optimal points. The training of the model on the two sets was performed using Tesla V100 GPUs with 16 GB of memory on the Google Colab platform with Adam optimizer, learning rate of $10^{-}4$, with scale ratio and rotation augmentations. The batch size of 2 was used to fully utilize the GPU capabilities and to obtain the best accuracy. Additionally, through consecutive validations after each epoch, we tracked the evolution of the model and saved the one with the highest performance based on the overall mAP.

5.4.3 Data Pre-processing

This section provides details about the necessary steps for preparing a compatible dataset for our experiment. The NuScenes dataset includes LiDAR and Radar point clouds, which are recorded under normal weather conditions with samples lacking proper synchronization. First, we propose our sensor synchronization module which up-samples the Radar OGMs and aligns them temporally to enable sensor fusion, creating a coherent and consistent dataset. Next, we discuss the mathematical modeling of the fog effect and its effect on the LiDAR point cloud toward building foggified driving scenarios.

Sensor Synchronization

One of the challenges in multi-modal fusion methods is the temporal misalignment of samples which generally stems from differences in sensor sampling frequencies. According to the authors of NuScenes, the sampling rate of the LiDAR sensor is 20Hz whereas the

Radar sampling frequency is 13Hz which results in about 400 LiDAR and 260 Radar frames per 20 seconds. The sampling time instances of both sensors and the misalignment between them over a one-second interval is depicted in fig. 5.10. The orange and the blue impulses are the timestamps at which LiDAR and Radar, respectively, present their measurement of the environment. It is only at the beginning of each second that the samples are timely aligned with each other, describing a scenario at the same time.

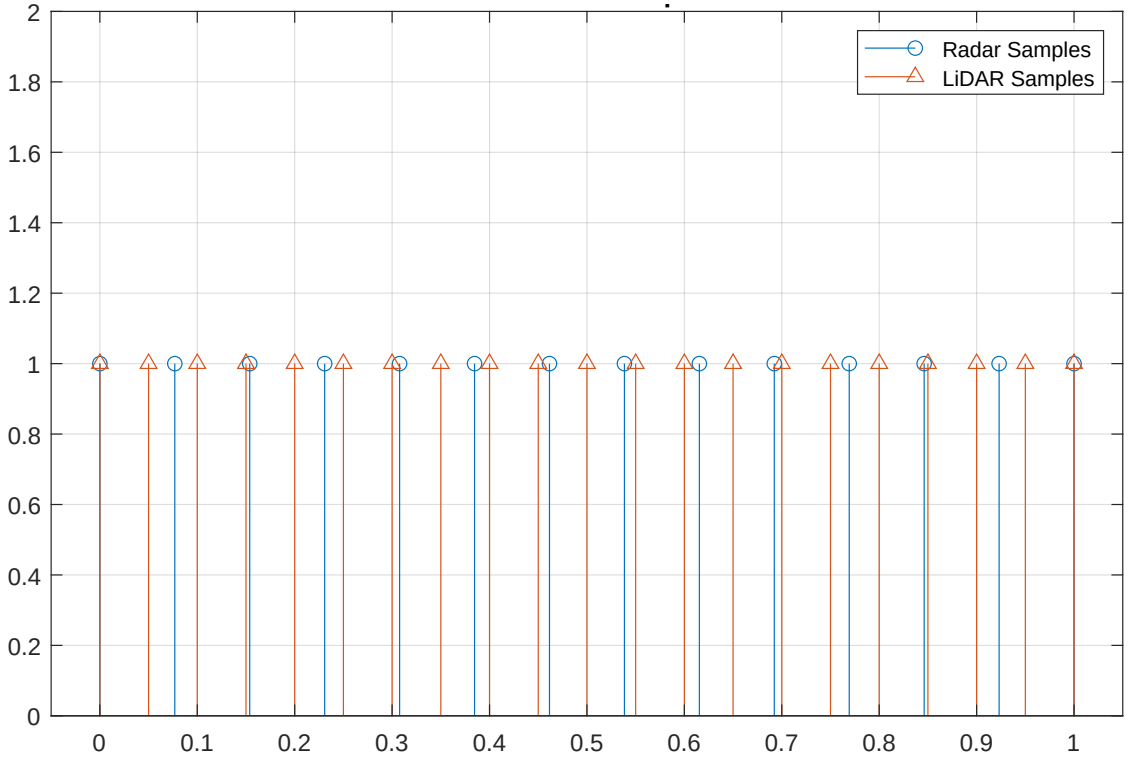


Figure 5.10: Illustration of sampling time points of LiDAR and Radar sensors in the NuScenes dataset and their temporal misalignment during an interval of one second.

To synchronize the Radar samples with LiDARs, we associate each LiDAR frame with the closest Radar sample. More specifically, the closest Radar samples to each LiDAR frame are identified, and then OGM is repeatedly generated with the same temporal tag as the sample from LiDAR. The time differences between each LiDAR sample and all the Radar samples are visualized in fig. 5.11 during this interval. The first row and column correspond to $t = T$ $[s]$ as the last row and column are for $t = T + 1$ $[s]$. Each row demonstrates the temporal distance of a specific LiDAR sample to all Radar samples where the negative sign is for samples of LiDAR preceding the ones from Radar. The minimum misalignment at each row is highlighted with red-colored values. The lowest time difference, zero seconds, appears on the first sample of each second where LiDAR and Radar are perfectly aligned. On the other hand, the largest time difference belongs to 11^{th} LiDAR which has a symmetrical temporal distance of 38 milliseconds from Radar

samples number 7 and 8. In this case, the past frame is used for association. The worst-case scenario of this pipeline does not exceed 0.64 meters in translation error, 1.2% of the detection range, for a vehicle driving at the speed of $60km/h$.

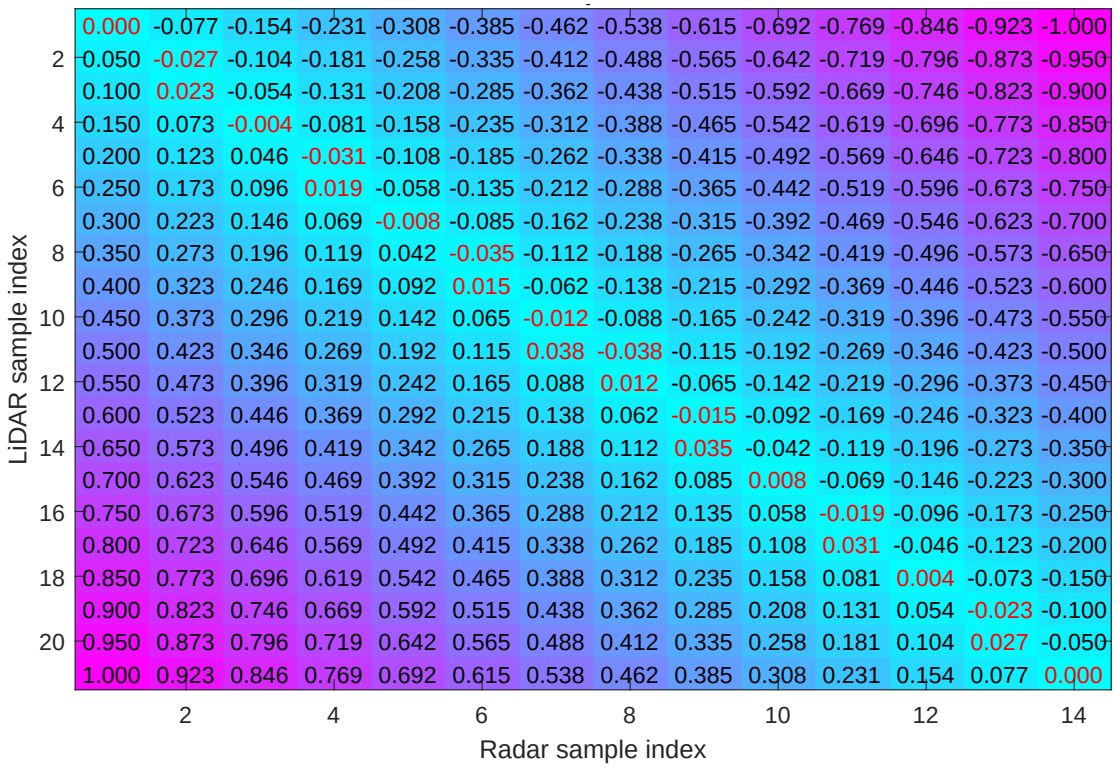


Figure 5.11: A heatmap of the temporal distance between each LiDAR sample to all Radar samples. Each row is associated with a specific LiDAR sample and each column belongs to a specific Radar sample. The cell contains a value of the temporal distance between each sample. The minimum misalignments are highlighted in red font which specifies temporal order.

Fog Effect Synthesis

To add the synthetic fog effect to the LiDAR point cloud we use the standard forward optical model formulated by *Koschmieder et al.* [176] and adjusted to LiDAR domain by *Bijelic et al.* [32] which is extensively used in the community. The mathematical fog model is expressed by the relation in eq. 5.7 where $\mathbf{I}_{clear}(\mathbf{x})$ is the emitted laser beam to the location $\mathbf{x}$, and $\mathbf{I}_{foggy}(\mathbf{x})$ is the attenuated reflected beam. The transmission function, $t(\mathbf{x})$, determines the visibility extent of the sensor.

$$\begin{aligned} \mathbf{I}_{foggy}(\mathbf{x}) &= \mathbf{I}_{clear}(\mathbf{x}).t(\mathbf{x}) \\ t(\mathbf{x}) &= exp(-\beta.l(\mathbf{x})) \end{aligned} \tag{5.7}$$

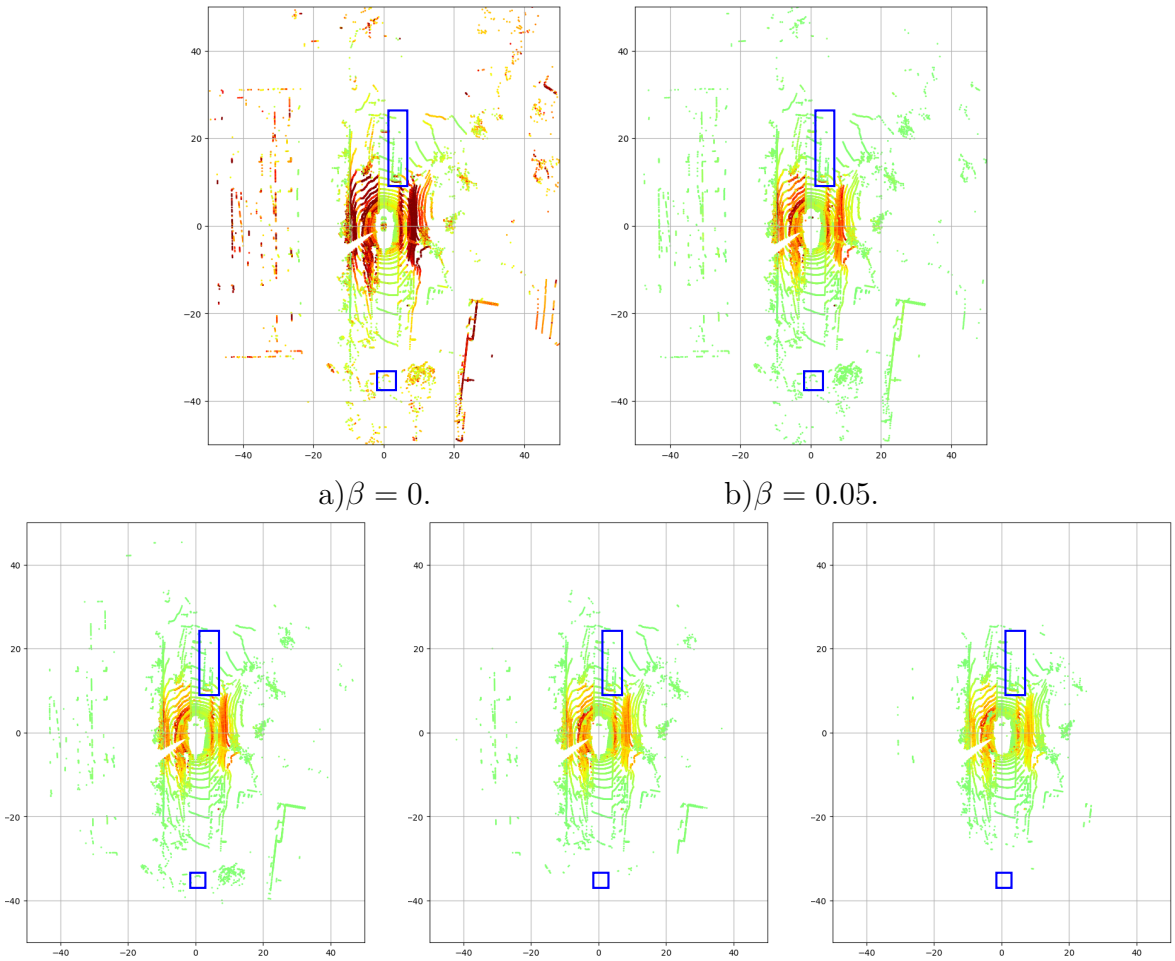
In this modeling, the density of the fog is inversely proportional to meteorological visibility and can be controlled by the attenuation coefficient β. The LiDAR sensor used by the authors of NuScenes is the Velodyne HDL32E model. The adaptive gain parameter g and the noise floor n of this sensor, shown in expression 5.8, have been reported to be 0.45 and 0.04, respectively, from the calibration procedures in [32].

$$r_{max} = -ln(\frac{n}{\mathbf{I}_{clear} + g}).\frac{1}{2\beta} \tag{5.8}$$

Ultimately, we prepare the final foggy LiADR point clouds by utilizing the algorithm developed by [32], accounting for laser beam attenuation and back-scattering effect. Four subsets are generated at different fog densities of 0.05, 0.067, 0.084, and $0.1m^{-1}$ which limit the visibility approximately to 50, 37.3, 29.7, and 25 meters, respectively. Figure 5.12 depicts the effect of synthesized fog with the densities of $\beta \in \{0.05, 0.067, 0.084, 0.1\}$ on the same input LiDAR frame. The points are assigned colors according to their reflectance parameter which is in an interval ranging from green, as the lowest, and ending with dark red, as the highest amount of reflectivity. The top left corner image shows the scenario under normal weather conditions in which the contextual information about the scene is clearly visible. For example, the blue box on the top and at the bottom highlights the presence of a bus and a vehicle in these frames. As the haze density increases, the maximum range of visibility reduces, preventing the sensor from sensing distant objects. The reduction in visibility stems from two phenomena: the attenuation of the laser beams which results in the reduction of average reflectance power, and the back-scattering effect adds noise to the frame by introducing false detections or obscuring some points. This gradual degradation of the details is evident from observing the frames in fig 5.12-a to e. While the fog becomes thicker, the average reflectance of the frame decreases and the loss of contextual details becomes greater. For example, the outline of the bus at 15 meters and the car at -40 meters in fig 5.12-b can be distinguished whereas in fig 5.12-d and e the car has completely disappeared and the detail of the sides and back of the bus is distorted considerably.

To evaluate the impact of integrating the Radar OGMs on the performance of the Pointpillars model, they need to be transformed into a readable format. To this end, we first generated the OGMs from all 50 scenes at the cell size of 0.15x0.15m^2. This resolution makes a trade-off between grid quality and the volume of the cells. Next, the grid is further down-sampled by a factor of 2 and then the cells in the uncertainty band are filtered out. The fusion algorithm has been experimentally found to work better when only the occupied cells are in the picture, therefore, we also filter the empty spaces after the down-sampling process. The occupancy probabilities are mapped to the median of the LiDAR reflectance distribution to approximately follow the shape of the distribution and act as the reflectance factor. The height of each point is augmented using the technique explained in 4.3.1, creating a semi-3D point cloud. Ultimately, each grid cell acts as a point carrying $(x, y, z, reflectance)$ information in the 3D space. The transformed Radar OGMs are aligned with LiDAR samples using the synchronization processes explained

in 5.4.3, then fused together following an early concatenation scheme. Finally, we have two distinct sets, called LiDAR-only and LiDAR-Radar fused (LR-fused), that are used for training and testing purposes under normal weather conditions. Furthermore, the validation subset of the LiDAR-only dataset undergoes the foggification process at four different fog densities that limit the maximum visibility and lower the detail quality of the point clouds. Then, this subset is fused with the Radar OGMs to produce the foggy LR-fused subset. Both foggified LiDAR-only and LR-fused sets are then used for robustness experiments.



a)$\beta = 0.$ b)$\beta = 0.05.$

c)$\beta = 0.067.$ d)$\beta = 0.084.$ e)$\beta = 0.1.$

Figure 5.12: Illustration of how different fog densities (β) reduce the maximum visibility of LiDAR point clouds from (a) Full visibility to (b) 50m, (c) 37.3m, (d) 29.7m, and (e) 25m. The blue rectangles at each frame indicate the presence of objects. The larger one belongs to a bus and the smaller one represents a car at the back of the ego vehicle.

5.4.4 Evaluation Results and Analysis

In this section, we provide a qualitative assessment of the effectiveness of OGMs on the robustness of the Pointpillars benchmark object detection model. Upon achieving the optimal models for each subset, we tested them on the foggified data to evaluate their performance and assess their robustness. It is noteworthy that while the NuScenes dataset offers ten object categories for detection tasks, we focused our analysis on six key categories that demonstrated reliable detection performance. These categories include "Car," "Pedestrian," "Truck," "Bus," "Barrier," and "Traffic Cone" which consist of object samples from various shapes and sizes, from small to large. This selection was based on their high and consistent mAP scores whereas the other four categories were not included in our analysis due to their significantly lower mAP scores. By concentrating on the consistent and relatively reliable detection results, we aimed to ensure a more accurate and reliable assessment of the model's performance.

To assess the robustness of the models against foggy weather, we measure the drop in mAP compared to the performance exhibited in the absence of fog particles. Greater degradation indicates less robustness capabilities against unfavorable conditions. To set the comparison base, we assessed the performance under normal weather conditions. A competitive result between the two models is observed with mAP of 47.3% from LR-fused and 48.5% from LiDAR-only datasets. This highlights a limitation of the fusion algorithm used in this experiment, in which minimal details of some objects are occluded by a number of huge occupancy clusters existing in the OGMs.

As the models are tested under various foggy weather conditions, a drop in the mAP occurs. The performance degradation of the LR-fused and LiDAR-only models are depicted in figure 5.13. The bar charts demonstrate the decrease in mAP under the fog densities of 0.05, 0.067, 0.084, and 0.1, which correspond to maximum visibility of 50, 37.3, 29.7, and 25 meters. As observed, at the highest fog density, 0.1 m^{-1}, the overall mAP on the LR-fused model drops by 18.2% whereas this drop on the LiDAR-only dataset results in a 20.1% mAP. The 1.9% improvement in robustness is caused by adding the Radar OGMs to the LiDAR point cloud. Furthermore, when comparing the performance of the model across different fog densities in the two datasets, it is evident that combining Radar information with LiDAR point clouds has been overall beneficial in addressing the performance degradation and improving the robustness of the model. Specifically, at medium to high fog densities of 0.067 and 0.084 the LiDAR-only dataset suffers by a reduction of 8.3% and 14.9%, whereas the inclusion of Radar data remedies this falloff to 7.2% and 12%. This signifies the improvement caused by the augmentation of information from Radar sensors to help with identifying objects with more precision.

Furthermore, we conducted a category-based assessment under the weather with the two most severe fog conditions, $\beta = 0.084$, $\beta = 0.1m^{-1}$, depicted in figure 5.15 and figure 5.14. The result demonstrates that the LR-fused data predominantly contributes to improving the overall robustness of final predictions. The "Car" category is the most sensitive class against foggy weather by exhibiting the highest increase among all classes.

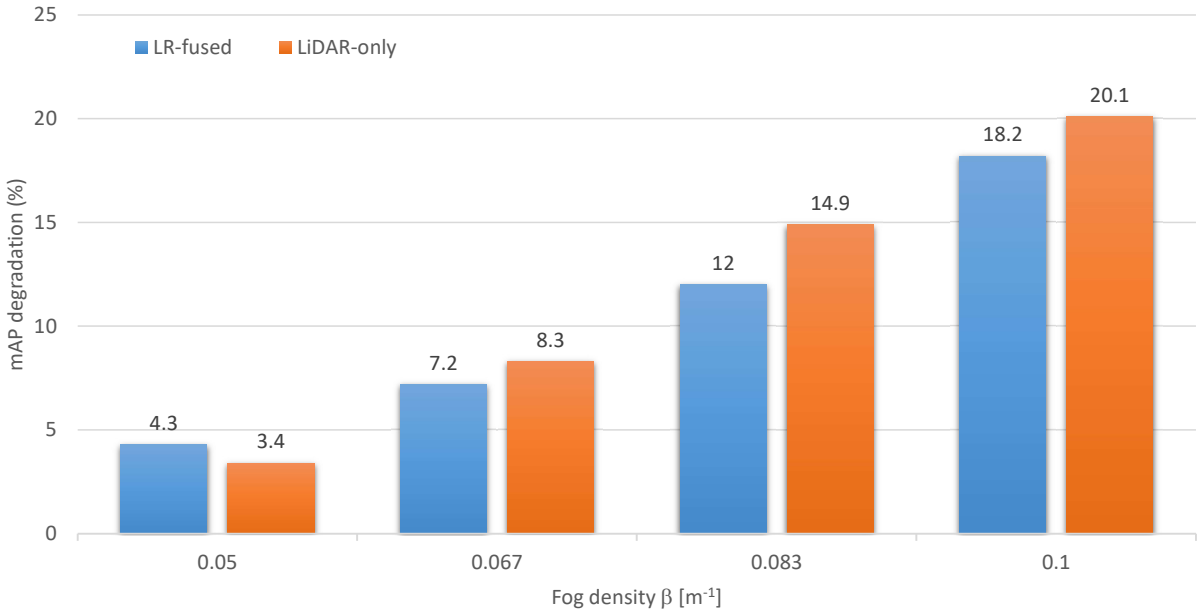


Figure 5.13: This figure shows the degradation in the performance of PointPillar on the LiDAR-only and LR-fused dataset under fog densities of 0.05, 0.067, 0.084 and 0.1 m^{-1} that results in the maximum visibility of 50, 37.3, 29.7 and 25 meters. The drop in performance at each fog density is measured with respect to the mAP of the model under normal weather conditions. Lower degradation represents higher robustness which is desired.

When fog density is $\beta = 0.084$, mAP of this category falls by 32.7% when detected by only LiDAR points. However, combined with Radar OGMs, the detection precision is increased by 2.4%. Under the highest fog density $\beta = 0.1$, the performance drop on LiDAR-only data is 43.8% whereas integrating the Radar OGMs mitigates this drop by 1.7%.

The performance of the model on the "Pedestrian" class, as an important category, exhibits robustness when using LR-fused data for detection. Under the medium-high fog density $\beta = 0.084$, a 21.6% performance drop is observed from LiDAR-only data whereas the LR-fused data remedies this by 1.6%. Moreover, the severe fog of $\beta = 0.1$ degrades the mAP on the LiDAR-only data by 21.6%, while the inclusion of Radar OGMs improves detection precision, reducing the performance drop by 1.8%. The OGM contribution toward the detection of small-sized objects extends to the "Traffic Cone" and "Barrier" classes by enhancing the robustness of the model by 7.7% and 0.4% under medium-high fog and 4.2% and 1.6% under sever fog density, respectively. These findings suggest that OGMs effectively pinpointed the present and the location of these small objects, even when their structural details were destructively affected or lost by the fog particles.

Even though the performance degradation for the "Bus" category is quite similar between the LR-fused and LiDAR-only datasets, the LR-fused point clouds enhance the

model's robustness by 4.4% and 4.1%, under medium-high and severe fog conditions, for the "Truck" instances. This indicates that while the benefits of integrating Radar data vary depending on the object type and size, it predominantly contributes to the robustness and detection accuracy under challenging foggy weather. Conclusively, this experiment highlights the capabilities of Radar-aided approaches in enhancing the robustness and reliability of the perception systems under adverse weather conditions.

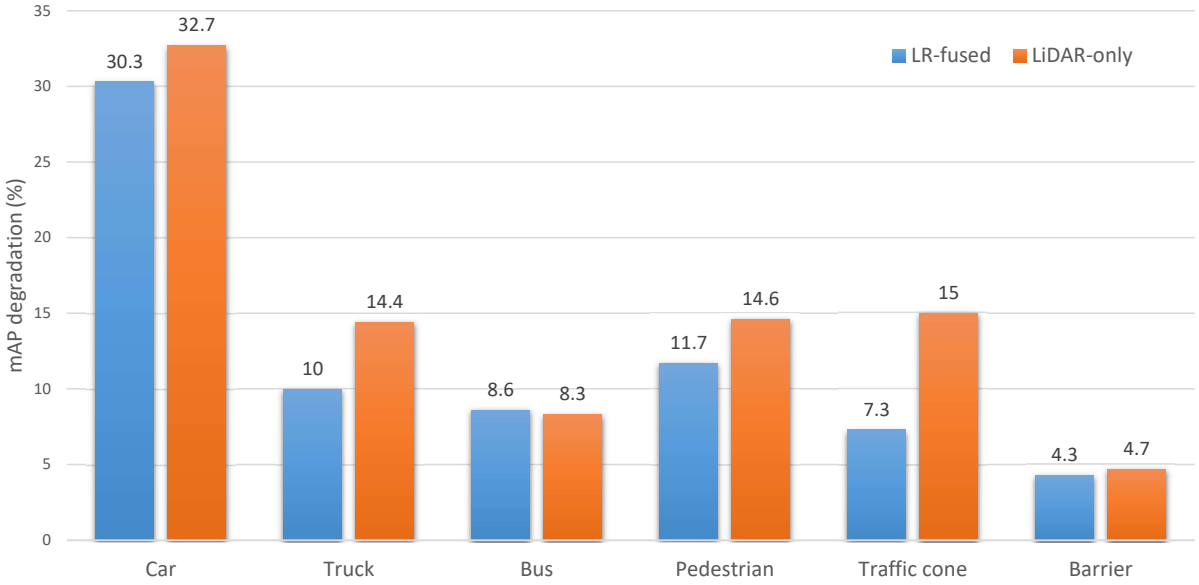


Figure 5.14: This figure shows the degradation in the performance under *medium-high* fog density of $\beta = 0.084m^{-1}$ of each object class. The blue and orange bars correspond to the mAP drop on the LR-fused and LiDAR-only datasets. The decrease in performance is measured with respect to the mAP achieved under normal weather conditions. Lower degradation represents higher robustness which is desired.

5.5 Conclusion

In this chapter, we thoroughly examined the characteristics of our proposed system, PARMG, from various angles. Firstly, we qualitatively analyzed the capabilities of PARMG in generating high-quality maps by using PIQE and SSIM metrics. Additionally, we assessed the effectiveness of our approach from an information-theoretic standpoint in tackling the sparsity issue existing in the Radar point clouds. We provided a cell-by-cell analysis accompanied by an entropy measurement to assess the capability of our system in constructing informative maps and purging uncertainty in the neighborhood of detected targets. Secondly, we highlighted the computational processing time required for generating high-resolution OGMs and addressed this challenge by employing a multi-processing scheme, showcasing the runtime optimization capabilities of our

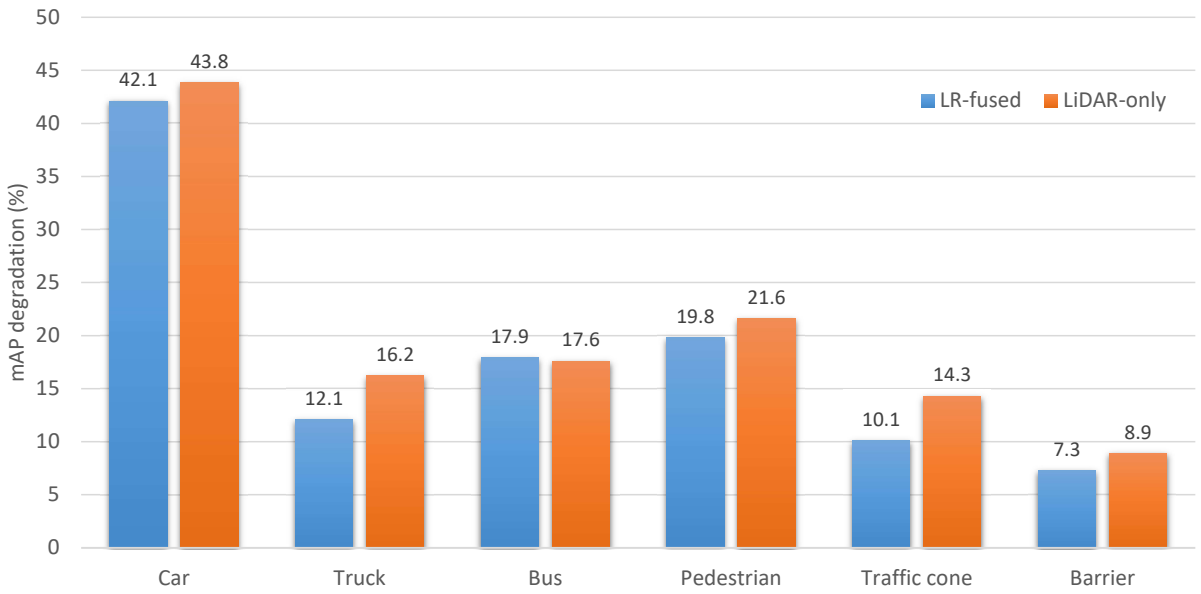


Figure 5.15: This figure shows the degradation in the performance under the *severe* fog density, $\beta = 0.1m^{-1}$, in each object class. The blue and orange bars correspond to the mAP drop on the LR-fused and LiDAR-only datasets. The decrease in performance is measured with respect to the mAP achieved under normal weather conditions. Lower degradation represents higher robustness which is desired.

system. Lastly, we demonstrated the effectiveness of the generated OGM in robustifying a LiDAR-based benchmark object detection model under adverse foggy weather conditions.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, we explored the critical role of perception systems in the realization of AVs and we focused on one of the major challenges these systems face, namely, adverse weather conditions. Traditional methods relying on camera and LiDAR sensors have suffered from performance degradation due to sensor limitations under these unfavorable conditions. While Radar sensors have shown potential advantages in robustifying the performance of perception systems against challenging weather, they are still under-explored compared to camera and LiDAR sensors. Furthermore, the lack of an adequate number of Radar-inclusive datasets poses another gap in the development of robust perception systems. Accordingly, we introduced the *Pseudo Analog Azimuth-Range Radar Maps Generator* (PARMG), our novel framework designed to enhance the quality of sparse Radar point cloud measurements by transforming them into high-quality Occupancy Grid Maps (OGMs). This approach aims to fill the knowledge gap in Radar-aided perception systems and facilitate comprehensive studies on the potential benefits of Radar sensors. Our experimental analysis demonstrated the effectiveness of the PARMG framework in generating informative representations of environments and enhancing detection performance under challenging weather conditions. The framework's ability to provide accessible high-quality Radar maps across different datasets signifies a notable contribution to the field.

In Chapter 2 a background and detailed review of the key elements of the perception system was given. The realization of advanced perception systems has been enabled by the Deep Learning frameworks designed to perform various perception tasks such as object detection or segmentation through the training procedure.

In Chapter 3 basic concepts and characteristics of the perception system were discussed through a comparative analysis of a number of different object detection models, highlighting their challenges against adverse weather conditions.

Chapter 4 explains the design and the mechanism of our proposed framework, PARGM. Our system enhances the quality of the sparse Radar recordings by constructing detailed

occupancy grid maps from the surroundings. PARGM leverages the temporal redundancy between consecutive frames and uses the Bayesian Filtering algorithm accompanied by a Radar-specific inverse sensor modeling to mine fine structural details and build an informative reconstruction of the local environment as a grid map over time.

In Chapter 5 we investigated the key features and characteristics of our system from quality, information-theoretic, and runtime-related aspects and showed its effectiveness in enhancing detection robustness under foggy weather.

6.2 Futuer Work

While our work has made several advancements and highlighted the potential benefits of integrating Radar sensors into perception systems for AVs, several areas require further development. These future directions can be studied further and contribute to the development of a more mature and more robust Radar-aided perception model in autonomous vehicles:

- Real-time Online OGM Mapping with High Resolution: In this study, we showed that resolution has a direct impact on the quality of the grid maps. To create a rich representation of the environment that contains fine-grained details, higher-resolution maps are required. Consequently, there is a need for more efficient implementations and advancements in real-time online occupancy grid mapping, considering the computational load of high-resolution maps. This would significantly enhance the ability of AVs to perceive the happenings in the driving scenarios more efficiently and perform decision-making and navigation accurately and in real-time.

- Use of More Complex Fusion Algorithms: Owing it the limitations in time and resources, the sensor fusion that was employed in this research had some shortcomings in which minimal degradation in quality would occur to the LiDAR point cloud after data fusion. This was primarily caused by some large clusters of occupied cells occluding parts of the object visible in the LiDAR point cloud. Employing more sophisticated and complex fusion algorithms, such as attention-based mechanisms, can improve the integration of data from multiple sensors and build a more comprehensive representation of the environment that would lead to more robust perception systems capable of better handling adverse weather conditions and other challenging scenarios.

- Integration of Radar OGMs into Multi-modal Models: Currently, the state-of-the-art perception models have achieved significant advancements in terms of runtime and accuracy under ideal conditions. However, many are still designed primarily using cameras and LiDARs in their sensor suite, which struggle to operate under challenging illumination and weather conditions. Further studies are required to

address the under-explored state of Radar-aided perception systems and to harness the advantages of this sensing modality. This will be facilitated by integrating Radar OGMs into various multi-modal models, complementing the data from cameras and LiDAR. This will demonstrate the capabilities of Radar sensors in realistic challenging situations. Eventually, this will be a pathway to develop solutions capable of handling more generalized, realistic, and challenging scenarios, such as adverse weather conditions. This is crucial for achieving reliable and effective autonomous driving in all environments.

The insights and contributions of this thesis serve as a foundation for ongoing research and development. By addressing these areas, the research community can make significant advancements toward developing perception systems that are not only accurate and efficient but also resilient and versatile enough to operate in a wide range of real-world conditions.

References

[1] Chapter 7 - miniature inertial measurement unit. In Zheng You, editor, *Space Microsystems and Micro/nano Satellites*, Micro and Nano Technologies, page 288. Butterworth-Heinemann, 2018.

[2] Fahad Jibrin Abdu, Yixiong Zhang, Maozhong Fu, Yuhan Li, and Zhenmiao Deng. Application of deep learning on millimeter-wave radar signals: A review. *Sensors*, 21(6):1951, 2021.

[3] Osama Abumansoor and Azzedine Boukerche. A secure cooperative approach for nonline-of-sight location verification in vanet. *IEEE Transactions on Vehicular Technology*, 61(1):275–285, 2011.

[4] National Aeronautics and Science Mission Directorate Space Administration.

[5] National Aeronautics and Science Mission Directorate Space Administration, 2016.

[6] United States Environmental Protection Agency. Electric vehicle myths, Jun 2024.

[7] Giancarlo Alessandretti, Alberto Broggi, and Pietro Cerri. Vehicle and guard rail detection using radar and vision data fusion. *IEEE transactions on intelligent transportation systems*, 8(1):95–105, 2007.

[8] Noura Aljeri and Azzedine Boukerche. A dynamic map discovery and selection scheme for predictive hierarchical mipv6 in vehicular networks. *IEEE Transactions on Vehicular Technology*, 69(1):793–806, 2019.

[9] Noura Aljeri and Azzedine Boukerche. Mobility management in 5g-enabled vehicular networks: Models, protocols, and classification. *ACM Computing Surveys (CSUR)*, 53(5):1–35, 2020.

[10] Noura Aljeri and Azzedine Boukerche. Smart and green mobility management for 5g-enabled vehicular networks. *Transactions on Emerging Telecommunications Technologies*, 33(3):e4054, 2022.

[11] Moayad Aloqaily, Ouns Bouachir, Azzedine Boukerche, and Ismaeel Al Ridhawi. Design guidelines for blockchain-assisted 5g-uav networks. *IEEE network*, 35(1):64–71, 2021.

[12] Moayad Aloqaily, Azzedine Boukerche, Ouns Bouachir, Fariea Khalid, and Sobia Jangsher. An energy trade framework using smart contracts: Overview and challenges. *IEEE Network*, 34(4):119–125, 2020.

[13] Codruta O Ancuti, Cosmin Ancuti, Mateu Sbert, and Radu Timofte. Dense-haze: A benchmark for image dehazing with dense-haze and haze-free images. In *2019 IEEE international conference on image processing (ICIP)*, pages 1014–1018. IEEE, 2019.

[14] Codruta O Ancuti, Cosmin Ancuti, and Radu Timofte. Nh-haze: An image dehazing benchmark with non-homogeneous hazy and haze-free images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 444–445, 2020.

[15] Codruta O Ancuti, Cosmin Ancuti, Radu Timofte, and Christophe De Vleeschouwer. O-haze: a dehazing benchmark with real hazy and haze-free outdoor images. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 754–762, 2018.

[16] Cosmin Ancuti, Codruta O Ancuti, and Christophe De Vleeschouwer. D-hazy: A dataset to evaluate quantitatively dehazing algorithms. In *2016 IEEE international conference on image processing (ICIP)*, pages 2226–2230. IEEE, 2016.

[17] Dave Anthony. Fog advisory issued for the southeast. 2023.

[18] Anurag Arnab and Philip HS Torr. Pixelwise instance segmentation with a dynamically instantiated network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 441–450, 2017.

[19] Eduardo Arnold, Omar Y Al-Jarrah, Mehrdad Dianati, Saber Fallah, David Oxtoby, and Alex Mouzakitis. A survey on 3d object detection methods for autonomous driving applications. *IEEE T-ITS*, 20(10):3782–3795, 2019.

[20] Mozhgan Nasr Azadani and Azzedine Boukerche. Driving behavior analysis guidelines for intelligent transportation systems. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):6027–6045, 2021.

[21] Mozhgan Nasr Azadani and Azzedine Boukerche. Stag: A novel interaction-aware path prediction method based on spatio-temporal attention graphs for connected automated vehicles. *Ad Hoc Networks*, 138:103021, 2023.

[22] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[23] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1090–1099, 2022.

[24] Athanasios Bamis, Azzedine Boukerche, Ioannis Chatzigiannakis, and Sotiris Nikoletseas. A mobility aware protocol synthesis for efficient routing in ad hoc mobile networks. *Computer Networks*, 52(1):130–154, 2008.

[25] Kshitiz Bansal, Keshav Rungta, Siyuan Zhu, and Dinesh Bharadia. Pointillism: accurate 3d bounding box estimation with multi-radars. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, pages 340–353, 2020.

[26] Dan Barnes, Matthew Gadd, Paul Murcutt, Paul Newman, and Ingmar Posner. The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6433–6438. IEEE, 2020.

[27] Rodolfo Bezerra Batista, Azzedine Boukerche, and Alba Cristina Magalhaes Alves de Melo. A parallel strategy for biological sequence alignment in restricted memory space. *Journal of Parallel and Distributed Computing*, 68(4):548–561, 2008.

[28] Lukas Beer, Thorsten Luettel, and Hans-Joachim Wuensche. Genpa-slam: Using a general panoptic segmentation for a real-time semantic landmark slam. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 873–879. IEEE, 2022.

[29] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9297–9307, 2019.

[30] Abhay Singh Bhadoriya, Vamsi Vegamoor, and Sivakumar Rathinam. Vehicle detection and tracking using thermal cameras in adverse visibility conditions. *Sensors*, 22(12):4567, 2022.

[31] Deblina Bhattacharjee, Seungryong Kim, Guillaume Vizier, and Mathieu Salzmann. Dunit: Detection-based unsupervised image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4787–4796, 2020.

[32] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11682–11692, 2020.

[33] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[34] Mario Bijelic, Tobias Gruber, and Werner Ritter. A benchmark for lidar sensors in fog: Is detection breaking down? In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 760–767. IEEE, 2018.

[35] Mario Bijelic, Tobias Gruber, and Werner Ritter. Benchmarking image sensors under adverse weather conditions for autonomous driving. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1773–1779. IEEE, 2018.

[36] Mario Bijelic, Paula Kysela, Tobias Gruber, Werner Ritter, and Klaus Dietmayer. Recovering the unseen: Benchmarking the generalization of enhancement methods to real world data in heavy fog. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[37] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

[38] A Boukerch, Li Xu, and Khalil El-Khatib. Trust-based security for wireless ad hoc and sensor networks. *Computer Communications*, 30(11-12):2413–2427, 2007.

[39] Azzedine Boukerche. Performance evaluation of routing protocols for ad hoc wireless networks. *Mobile networks and applications*, 9:333–342, 2004.

[40] Azzedine Boukerche. *Handbook of algorithms for wireless networking and mobile computing*. CRC Press, 2005.

[41] Azzedine Boukerche. *Algorithms and protocols for wireless sensor networks*. Wiley Online Library, 2009.

[42] Azzedine Boukerche and Luciano Bononi. Simulation and modeling of wireless, mobile, and ad hoc networks. *Mobile ad hoc networking*, pages 373–409, 2004.

[43] Azzedine Boukerche, Xiuzhen Cheng, and Joseph Linus. Energy-aware data-centric routing in microsensor networks. In *Proceedings of the 6th ACM international workshop on Modeling analysis and simulation of wireless and mobile systems*, pages 42–49, 2003.

[44] Azzedine Boukerche, Xuzhen Cheng, and Joseph Linus. A performance evaluation of a novel energy-aware data-centric routing algorithm in wireless sensor networks. *Wireless Networks*, 11(5):619–635, 2005.

[45] Azzedine Boukerche and Amir Darehshoorzadeh. Opportunistic routing in wireless networks: Models, algorithms, and classifications. *ACM Computing Surveys (CSUR)*, 47(2):1–36, 2014.

[46] Azzedine Boukerche and Sajal K Das. Dynamic load balancing strategies for conservative parallel simulations. In *Proceedings of the eleventh workshop on Parallel and distributed simulation*, pages 20–28, 1997.

[47] Azzedine Boukerche, Sajal K Das, and Alessandro Fabbri. Analysis of a randomized congestion control scheme with dsdv routing in ad hoc wireless networks. *Journal of Parallel and Distributed Computing*, 61(7):967–995, 2001.

[48] Azzedine Boukerche, Khalil El-Khatib, Li Xu, and Larry Korba. Sdar: a secure distributed anonymous routing protocol for wireless and mobile ad hoc networks. In *29th Annual IEEE International Conference on Local Computer Networks*, pages 618–624. IEEE, 2004.

[49] Azzedine Boukerche, Khalil El-Khatib, Li Xu, and Larry Korba. An efficient secure distributed anonymous routing protocol for mobile and wireless ad hoc networks. *computer communications*, 28(10):1193–1203, 2005.

[50] Azzedine Boukerche and Xin Fei. A coverage-preserving scheme for wireless sensor network with irregular sensing range. *Ad hoc networks*, 5(8):1303–1316, 2007.

[51] Azzedine Boukerche and Xin Fei. A voronoi approach for coverage protocols in wireless sensor networks. In *IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference*, pages 5190–5194. IEEE, 2007.

[52] Azzedine Boukerche, Sungbum Hong, and Tom Jacob. An efficient synchronization scheme of multimedia streams in wireless and mobile systems. *IEEE transactions on Parallel and Distributed Systems*, 13(9):911–923, 2002.

[53] Azzedine Boukerche and Zhijun Hou. Object detection using deep learning methods in traffic scenarios. *ACM Computing Surveys (CSUR)*, 54(2):1–35, 2021.

[54] Azzedine Boukerche, Kathia Regina Lemos Jucá, Joao Bosco Sobral, and Mirela Sechi Moretti Annoni Notare. An artificial immune based intrusion detection model for computer and telecommunication systems. *Parallel Computing*, 30(5-6):629–646, 2004.

[55] Azzedine Boukerche and Xu Li. An agent-based trust and reputation management scheme for wireless sensor networks. In *GLOBECOM'05. IEEE Global Telecommunications Conference, 2005.*, volume 3, pages 5–pp. IEEE, 2005.

[56] Azzedine Boukerche, Renato B Machado, Kathia RL Jucá, João Bosco M Sobral, and Mirela SMA Notare. An agent based and biological inspired real-time intrusion detection and security model for computer network operations. *Computer Communications*, 30(13):2649–2660, 2007.

[57] Azzedine Boukerche, Alexander Magnano, and Noura Aljeri. Mobile ip handover for vehicular networks: Methods, models, and classifications. *ACM Computing Surveys (CSUR)*, 49(4):1–34, 2017.

[58] Azzedine Boukerche, Anahit Martirosyan, and Richard Pazzi. An inter-cluster communication based energy aware and fault tolerant protocol for wireless sensor networks. *Mobile Networks and Applications*, 13:614–626, 2008.

[59] Azzedine Boukerche and Mirela Sechi M Annoni Notare. Behavior-based intrusion detection in mobile phone systems. *Journal of Parallel and Distributed Computing*, 62(9):1476–1490, 2002.

[60] Azzedine Boukerche, Horacio ABF Oliveira, Eduardo F Nakamura, and Antonio AF Loureiro. Localization systems for wireless sensor networks. *IEEE wireless Communications*, 14(6):6–12, 2007.

[61] Azzedine Boukerche, Horacio ABF Oliveira, Eduardo F Nakamura, and Antonio AF Loureiro. Secure localization algorithms for wireless sensor networks. *IEEE Communications Magazine*, 46(4):96–101, 2008.

[62] Azzedine Boukerche, Horacio ABF Oliveira, Eduardo F Nakamura, and Antonio AF Loureiro. Vehicular ad hoc networks: A new challenge for localization-based systems. *Computer communications*, 31(12):2838–2849, 2008.

[63] Azzedine Boukerche, Horacio ABF Oliveira, Eduardo Freire Nakamura, and Antonio AF Loureiro. Dv-loc: a scalable localization protocol using voronoi diagrams for wireless sensor networks. *IEEE Wireless Communications*, 16(2):50–55, 2009.

[64] Azzedine Boukerche, Richard Werner Nelem Pazzi, and Regina B Araujo. Hpeq a hierarchical periodic, event-driven and query-based wireless sensor network protocol. In *The IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05) l*, pages 560–567. IEEE, 2005.

[65] Azzedine Boukerche, Richard Werner Nelem Pazzi, and Regina Borges Araujo. A fast and reliable protocol for wireless sensor networks in critical conditions monitoring applications. In *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pages 157–164, 2004.

[66] Azzedine Boukerche, Richard Werner Nelem Pazzi, and Regina Borges Araujo. Fault-tolerant wireless sensor network routing protocols for the supervision of context-aware physical environments. *Journal of Parallel and Distributed Computing*, 66(4):586–599, 2006.

[67] Azzedine Boukerche and Yonglin Ren. A trust-based security system for ubiquitous and pervasive computing environments. *Computer communications*, 31(18):4343–4351, 2008.

[68] Azzedine Boukerche and Yonglin Ren. A secure mobile healthcare system using trust-based multicast scheme. *IEEE Journal on Selected Areas in Communications*, 27(4):387–399, 2009.

[69] Azzedine Boukerche, Cristiano Rezende, and Richard W Pazzi. Improving neighbor localization in vehicular ad hoc networks to avoid overhead from periodic messages. In *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*, pages 1–6. IEEE, 2009.

[70] Azzedine Boukerche and E Robson. Vehicular cloud computing: Architectures, applications, and mobility. *Computer networks*, 135:171–189, 2018.

[71] Azzedine Boukerche and Samer Samarah. A novel algorithm for mining association rules in wireless ad hoc sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 19(7):865–877, 2008.

[72] Azzedine Boukerche and Peng Sun. Connectivity and coverage based protocols for wireless sensor networks. *Ad Hoc Networks*, 80:54–69, 2018.

[73] Azzedine Boukerche, Yanjie Tao, and Peng Sun. Artificial intelligence-based vehicular traffic flow prediction methods for supporting intelligent transportation systems. *Computer networks*, 182:107484, 2020.

[74] Azzedine Boukerche, Begumhan Turgut, Nevin Aydin, Mohammad Z Ahmad, Ladislau Bölöni, and Damla Turgut. Routing protocols in ad hoc networks: A survey. *Computer networks*, 55(13):3032–3080, 2011.

[75] Azzedine Boukerche and Jiahao Wang. Machine learning-based traffic prediction models for intelligent transportation systems. *Computer Networks*, 181:107530, 2020.

[76] Azzedine Boukerche, Lining Zheng, and Omar Alfandi. Outlier detection: Methods, models, and classification. *ACM Computing Surveys (CSUR)*, 53(3):1–37, 2020.

[77] Guillaume Bresson, Zayed Alsayed, Li Yu, and Sébastien Glaser. Simultaneous localization and mapping: A survey of current trends in autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 2(3):194–220, 2017.

[78] Tim Brophy, Darragh Mullins, Ashkan Parsi, Jonathan Horgan, Enda Ward, Patrick Denny, Ciarán Eising, Brian Deegan, Martin Glavin, and Edward Jones. A review of the impact of rain on camera-based perception in automated driving systems. *IEEE Access*, 2023.

[79] Gabriel J Brostow, Jamie Shotton, Julien Fauqueur, and Roberto Cipolla. Segmentation and recognition using structure from motion point clouds. In *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I 10*, pages 44–57. Springer, 2008.

[80] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[81] Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, and Dacheng Tao. Dehazenet: An end-to-end system for single image haze removal. *IEEE transactions on image processing*, 25(11):5187–5198, 2016.

[82] Hongxiang Cai, Zeyuan Zhang, Zhenyu Zhou, Ziyin Li, Wenbo Ding, and Jiuhua Zhao. Bevfusion4d: Learning lidar-camera fusion under bird's-eye-view via cross-modality guidance and temporal aggregation. *arXiv preprint arXiv:2303.17099*, 2023.

[83] Luca Caltagirone, Samuel Scheidegger, Lennart Svensson, and Mattias Wahde. Fast lidar-based road detection using fully convolutional neural networks. In *2017 ieee intelligent vehicles symposium (iv)*, pages 1019–1024. IEEE, 2017.

[84] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986.

[85] Clayson Celes, Fabricio A Silva, Azzedine Boukerche, Rossana Maria de Castro Andrade, and Antonio AF Loureiro. Improving vanet simulation with calibrated vehicular mobility traces. *IEEE Transactions on Mobile Computing*, 16(12):3376–3389, 2017.

[86] Simon Chadwick, Will Maddern, and Paul Newman. Distant vehicle detection using radar and vision. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8311–8317. IEEE, 2019.

[87] Tony F Chan and Luminita A Vese. Active contours without edges. *IEEE Transactions on image processing*, 10(2):266–277, 2001.

[88] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, et al. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8748–8757, 2019.

[89] Nicholas Charron, Stephen Phillips, and Steven L Waslander. De-noising of lidar point clouds corrupted by snowfall. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 254–261. IEEE, 2018.

[90] Bo-Hao Chen, Shih-Chia Huang, Chian-Ying Li, and Sy-Yen Kuo. Haze removal using radial basis function networks for visibility restoration applications. *IEEE transactions on neural networks and learning systems*, 29(8):3828–3838, 2017.

[91] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4013–4022, 2018.

[92] Liang-Chieh Chen, Raphael Gontijo Lopes, Bowen Cheng, Maxwell D Collins, Ekin D Cubuk, Barret Zoph, Hartwig Adam, and Jonathon Shlens. Naive-student: Leveraging semi-supervised learning in video sequences for urban scene segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 695–714. Springer, 2020.

[93] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[94] Xiaozhi Chen, Kaustav Kundu, Ziyu Zhang, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2147–2156, 2016.

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

[96] Yilun Chen, Zhiding Yu, Yukang Chen, Shiyi Lan, Anima Anandkumar, Jiaya Jia, and Jose M Alvarez. Focalformer3d: focusing on hard instance for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8394–8405, 2023.

[97] Jun Cheng, Liyan Zhang, Qihong Chen, Xinrong Hu, and Jingcao Cai. A review of visual slam methods for autonomous driving vehicles. *Engineering Applications of Artificial Intelligence*, 114:104992, 2022.

[98] Ji Dong Choi and Min Young Kim. A sensor fusion system with thermal infrared camera and lidar for autonomous vehicles and deep learning based object detection. *ICT Express*, 2022.

[99] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. https://github.com/open-mmlab/mmdetection3d, 2020.

[100] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[101] Sergio Correia, Azzedine Boukerche, and Rodolfo I Meneguette. An architecture for hierarchical software-defined vehicular networks. *IEEE Communications Magazine*, 55(7):80–86, 2017.

[102] Rodolfo WL Coutinho, Azzedine Boukerche, Luiz FM Vieira, and Antonio AF Loureiro. Gedar: Geographic and opportunistic routing protocol with depth adjustment for mobile underwater sensor networks. In *2014 IEEE International Conference on communications (ICC)*, pages 251–256. IEEE, 2014.

[103] Rodolfo WL Coutinho, Azzedine Boukerche, Luiz FM Vieira, and Antonio AF Loureiro. Geographic and opportunistic routing for underwater sensor networks. *IEEE Transactions on Computers*, 65(2):548–561, 2015.

[104] Rodolfo WL Coutinho, Azzedine Boukerche, Luiz FM Vieira, and Antonio AF Loureiro. A novel void node recovery paradigm for long-term underwater sensor networks. *Ad Hoc Networks*, 34:144–156, 2015.

[105] Rodolfo WL Coutinho, Azzedine Boukerche, Luiz FM Vieira, and Antonio AF Loureiro. Design guidelines for opportunistic routing in underwater networks. *IEEE Communications Magazine*, 54(2):40–48, 2016.

[106] Rodolfo WL Coutinho, Azzedine Boukerche, Luiz FM Vieira, and Antonio AF Loureiro. Underwater wireless sensor networks: A new challenge for topology control–based systems. *ACM Computing Surveys (CSUR)*, 51(1):1–36, 2018.

[107] Felipe Cunha, Leandro Villas, Azzedine Boukerche, Guilherme Maia, Aline Viana, Raquel AF Mini, and Antonio AF Loureiro. Data communication in vanets: Protocols, applications and challenges. *Ad hoc networks*, 44:90–103, 2016.

[108] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3150–3158, 2016.

[109] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1, 2005.

[110] Yahia Dalbah, Jean Lahoud, and Hisham Cholakkal. Radarformer: Lightweight and accurate real-time radar object detection model. In *Scandinavian Conference on Image Analysis*, pages 341–358. Springer, 2023.

[111] Amir Darehshoorzadeh and Azzedine Boukerche. Underwater sensor networks: A new challenge for opportunistic routing protocols. *IEEE Communications Magazine*, 53(11):98–107, 2015.

[112] J Davis and M Keck. A two-stage approach to person detectionin thermal imagery. In *Proceeding of Workshop on Applications of Computer Vision (WACV)*, 2005.

[113] Bert De Brabandere, Davy Neven, and Luc Van Gool. Semantic instance segmentation for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 7–9, 2017.

[114] Daan de Geus, Panagiotis Meletis, and Gijs Dubbelman. Fast panoptic segmentation network. *IEEE Robotics and Automation Letters*, 5(2):1742–1749, 2020.

[115] Horacio Antonio Braga Fernandes De Oliveira, Azzedine Boukerche, Eduardo Freire Nakamura, and Antonio Alfredo Ferreira Loureiro. An efficient directed localization recursion protocol for wireless sensor networks. *IEEE Transactions on Computers*, 58(5):677–691, 2008.

[116] Johan Degerman, Thomas Pernstål, and Klas Alenljung. 3d occupancy grid mapping using statistical radar models. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 902–908. IEEE, 2016.

[117] Omid Dehzangi, Mojtaba Taherisadr, and Raghvendar ChangalVala. Imu-based gait recognition using convolutional neural networks and multi-sensor fusion. *Sensors*, 17(12):2735, 2017.

[118] Hanieh Deilamsalehy and Timothy C Havens. Sensor fused three-dimensional localization using imu, camera and lidar. In *2016 IEEE SENSORS*, pages 1–3. IEEE, 2016.

[119] Zijun Deng, Lei Zhu, Xiaowei Hu, Chi-Wing Fu, Xuemiao Xu, Qing Zhang, Jing Qin, and Pheng-Ann Heng. Deep multi-model fusion for single-image dehazing. In

Proceedings of the IEEE/CVF international conference on computer vision, pages 2453–2462, 2019.

[120] Ayush Dewan and Wolfram Burgard. Deeptemporalseg: Temporally consistent semantic segmentation of 3d lidar scans. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2624–2630. IEEE, 2020.

[121] Jean-Luc Déziel, Pierre Merriaux, Francis Tremblay, Dave Lessard, Dominique Plourde, Julien Stanguennec, Pierre Goulet, and Pierre Olivier. Pixset: An opportunity for 3d computer vision to go beyond point clouds with a full-waveform lidar dataset. In *2021 ieee international intelligent transportation systems conference (itsc)*, pages 2987–2993. IEEE, 2021.

[122] Bertrand Douillard, James Underwood, Noah Kuntz, Vsevolod Vlaskine, Alastair Quadros, Peter Morton, and Alon Frenkel. On the segmentation of 3d lidar point clouds. In *2011 IEEE International Conference on Robotics and Automation*, pages 2798–2805. IEEE, 2011.

[123] Florian Drews, Di Feng, Florian Faion, Lars Rosenbaum, Michael Ulrich, and Claudius Gläser. Deepfusion: A robust and modular 3d object detector for lidars, cameras and radars. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 560–567. IEEE, 2022.

[124] Kyle Edward. Goodbye gridlock: How autonomous vehicles can revolutionize city living, Jul 2022.

[125] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE international conference on computer vision*, pages 2650–2658, 2015.

[126] Omar Elharrouss, Somaya Al-Maadeed, Nandhini Subramanian, Najmath Ottakath, Noor Almaadeed, and Yassine Himeur. Panoptic segmentation: A review. *arXiv preprint arXiv:2111.10250*, 2021.

[127] Martin Engelcke, Dushyant Rao, Dominic Zeng Wang, Chi Hay Tong, and Ingmar Posner. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1355–1361. IEEE, 2017.

[128] EUSPA.

[129] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88:303–338, 2010.

[130] Lue Fan, Ziqi Pang, Tianyuan Zhang, Yu-Xiong Wang, Hang Zhao, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Embracing single stride 3d object detector with sparse transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8458–8468, 2022.

[131] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1915–1929, 2012.

[132] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[133] Di Feng, Christian Haase-Schütz, Lars Rosenbaum, Heinz Hertlein, Claudius Glaeser, Fabian Timm, Werner Wiesbeck, and Klaus Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360, 2020.

[134] Thomas Fersch, Alexander Buhmann, Alexander Koelpin, and Robert Weigel. The influence of rain on small aperture lidar sensors. In *2016 German Microwave Conference (GeMiC)*, pages 84–87. IEEE, 2016.

[135] Naiyu Gao, Fei He, Jian Jia, Yanhu Shan, Haoyang Zhang, Xin Zhao, and Kaiqi Huang. Panopticdepth: A unified framework for depth-aware panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1632–1642, 2022.

[136] Kyle Genova, Xiaoqi Yin, Abhijit Kundu, Caroline Pantofaru, Forrester Cole, Avneesh Sud, Brian Brewington, Brian Shucker, and Thomas Funkhouser. Learning 3d semantic segmentation with only 2d image supervision. In *2021 International Conference on 3D Vision (3DV)*, pages 361–372. IEEE, 2021.

[137] Prasenjit Ghorai, Azim Eskandarian, Young-Keun Kim, and Goodarz Mehr. State estimation and motion prediction of vehicles and vulnerable road users for cooperative autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):16983–17002, 2022.

[138] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[139] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):142–158, 2015.

[140] Benjamin Graham. Spatially-sparse convolutional neural networks. *arXiv preprint arXiv:1409.6070*, 2014.

[141] David Grangier, Léon Bottou, and Ronan Collobert. Deep convolutional networks for scene parsing. In *ICML 2009 deep learning workshop*, volume 3, page 109, 2009.

[142] Tobias Gruber, Mario Bijelic, Felix Heide, Werner Ritter, and Klaus Dietmayer. Pixel-accurate depth evaluation in realistic driving scenarios. In *2019 International Conference on 3D Vision (3DV)*, pages 95–105. IEEE, 2019.

[143] Tobias Gruber, Frank Julca-Aguilar, Mario Bijelic, and Felix Heide. Gated2depth: Real-time dense lidar from gated images. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019.

[144] Junyao Guo, Unmesh Kurup, and Mohak Shah. Is it safe to drive? an overview of factors, metrics, and datasets for driveability assessment in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 21(8):3135–3151, 2019.

[145] Yulan Guo, Hanyun Wang, Qingyong Hu, Hao Liu, Li Liu, and Mohammed Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(12):4338–4364, 2020.

[146] Zhiyang Guo, Yingping Huang, Hongjian Wei, Chong Zhang, Baigan Zhao, and Zheqin Shao. Dalanenet: A dual attention instance segmentation network for real-time lane detection. *IEEE sensors journal*, 21(19):21730–21739, 2021.

[147] Hadi Habibzadeh, Cem Kaptan, Tolga Soyata, Burak Kantarci, and Azzedine Boukerche. Smart city system design: A comprehensive study of the application and data planes. *ACM Computing Surveys (CSUR)*, 52(2):1–38, 2019.

[148] Martin Hahner, Christos Sakaridis, Mario Bijelic, Felix Heide, Fisher Yu, Dengxin Dai, and Luc Van Gool. Lidar snowfall simulation for robust 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16364–16374, 2022.

[149] Chenhang He, Hui Zeng, Jianqiang Huang, Xian-Sheng Hua, and Lei Zhang. Structure aware single-stage 3d object detection from point cloud. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11873–11882, 2020.

[150] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[151] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[152] Robin Heinzler, Florian Piewak, Philipp Schindler, and Wilhelm Stork. Cnn-based lidar point cloud de-noising in adverse weather. *IEEE Robotics and Automation Letters*, 5(2):2514–2521, 2020.

[153] Robin Heinzler, Philipp Schindler, Jürgen Seekircher, Werner Ritter, and Wilhelm Stork. Weather influence and classification with automotive lidar sensors. In *2019 IEEE intelligent vehicles symposium (IV)*, pages 1527–1534. IEEE, 2019.

[154] Noureldin Hendy, Cooper Sloan, Feng Tian, Pengfei Duan, Nick Charchut, Yuesong Xie, Chuang Wang, and James Philbin. Fishing net: Future inference of semantic heatmaps in grids. *arXiv preprint arXiv:2006.09917*, 2020.

[155] Michael Himmelsbach, Felix V Hundelshausen, and H-J Wuensche. Fast segmentation of 3d point clouds for ground vehicles. In *2010 IEEE Intelligent Vehicles Symposium*, pages 560–565. IEEE, 2010.

[156] Namdar Homayounfar, Yuwen Xiong, Justin Liang, Wei-Chiu Ma, and Raquel Urtasun. Levelset r-cnn: A deep variational method for instance segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*, pages 555–571. Springer, 2020.

[157] Weixiang Hong, Qingpei Guo, Wei Zhang, Jingdong Chen, and Wei Chu. Lpsnet: A lightweight solution for fast panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16746–16754, 2021.

[158] Armin Hornung, Kai M. Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 2013. Software available at https://octomap.github.io.

[159] Ji Hou, Angela Dai, and Matthias Nießner. 3d-sis: 3d semantic instance segmentation of rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4421–4430, 2019.

[160] Rui Hou, Jie Li, Arjun Bhargava, Allan Raventos, Vitor Guizilini, Chao Fang, Jerome Lynch, and Adrien Gaidon. Real-time panoptic segmentation from dense detections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8523–8532, 2020.

[161] Chunyong Hu, Hang Zheng, Kun Li, Jianyun Xu, Weibo Mao, Maochun Luo, Lingxuan Wang, Mingxia Chen, Kaixuan Liu, Yiru Zhao, et al. Fusionformer: A multi-sensory fusion in bird's-eye-view and temporal consistent transformer for 3d objection. *arXiv preprint arXiv:2309.05257*, 2023.

[162] Qiangui Huang, Weiyue Wang, and Ulrich Neumann. Recurrent slice networks for 3d segmentation of point clouds. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2626–2635, 2018.

[163] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscape dataset for autonomous driving. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 954–960, 2018.

[164] Soonmin Hwang, Jaesik Park, Namil Kim, Yukyung Choi, and In So Kweon. Multispectral pedestrian detection: Benchmark dataset and baseline. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1037–1045, 2015.

[165] Alexander Jaus, Kailun Yang, and Rainer Stiefelhagen. Panoramic panoptic segmentation: Insights into surrounding parsing for mobile agents via unsupervised contrastive learning. *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[166] Zicong Jiang, Liquan Zhao, Shuaiyang Li, and Yanfei Jia. Real-time object detection method based on improved yolov4-tiny. *arXiv preprint arXiv:2011.04244*, 2020.

[167] Tianzhe Jiao, Chaopeng Guo, Xiaoyue Feng, Yuming Chen, and Jie Song. A comprehensive survey on deep learning multi-modal fusion: Methods, technologies and applications. *Computers, Materials & Continua*, 80(1), 2024.

[168] Narsimlu Kemsaram, Anweshan Das, and Gijs Dubbelman. A stereo perception framework for autonomous vehicles. In *2020 IEEE 91st vehicular technology conference (VTC2020-Spring)*, pages 1–6. IEEE, 2020.

[169] Atmane Khellal, Hongbin Ma, and Qing Fei. Pedestrian classification and detection in far infrared images. In *Intelligent Robotics and Applications: 8th International Conference, ICIRA 2015, Portsmouth, UK, August 24-27, 2015, Proceedings, Part I 8*, pages 511–522. Springer, 2015.

[170] Jisong Kim, Minjae Seong, Geonho Bang, Dongsuk Kum, and Jun Won Choi. Rcmfusion: Radar-camera multi-level fusion for 3d object detection. *arXiv preprint arXiv:2307.10249*, 2023.

[171] Taek-Lim Kim and Tae-Hyoung Park. Camera-lidar fusion method with feature switch layer for object detection networks. *Sensors*, 22(19):7163, 2022.

[172] Youngseok Kim, Sanmin Kim, Jun Won Choi, and Dongsuk Kum. Craft: Camera-radar 3d object detection with spatio-contextual fusion transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1160–1168, 2023.

[173] Youngseok Kim, Juyeb Shin, Sanmin Kim, In-Jae Lee, Jun Won Choi, and Dongsuk Kum. Crn: Camera radar net for accurate, robust, efficient 3d perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17615–17626, 2023.

[174] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6399–6408, 2019.

[175] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019.

[176] Harald Koschmieder. Theorie der horizontalen sichtweite. *Beitrage zur Physik der freien Atmosphare*, pages 33–53, 1924.

[177] Robert Krajewski, Julian Bock, Laurent Kloeker, and Lutz Eckstein. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2118–2125. IEEE, 2018.

[178] Florian Kraus, Nicolas Scheiner, Werner Ritter, and Klaus Dietmayer. Using machine learning to detect ghost images in automotive radar. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–7. IEEE, 2020.

[179] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d proposal generation and object detection from view aggregation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–8. IEEE, 2018.

[180] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d proposal generation and object detection from view aggregation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–8. IEEE, 2018.

[181] Jason Ku, Alex D Pon, and Steven L Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11867–11876, 2019.

[182] Irfan Tito Kurniawan and Bambang Riyanto Trilaksono. Clusterfusion: Leveraging radar spatial features for radar-camera 3d object detection in autonomous vehicles. *IEEE Access*, 2023.

[183] L'ubor Ladickỳ, Paul Sturgess, Karteek Alahari, Chris Russell, and Philip HS Torr. What, where and how many? combining object detectors and crfs. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pages 424–437. Springer, 2010.

[184] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12697–12705, 2019.

[185] Kai Lei, Zhan Chen, Shuman Jia, and Xiaoteng Zhang. Hvdetfusion: A simple and robust camera-radar fusion framework. *arXiv preprint arXiv:2307.11323*, 2023.

[186] Boyi Li, Xiulian Peng, Zhangyang Wang, Jizheng Xu, and Dan Feng. Aod-net: All-in-one dehazing network. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4780–4788, 2017.

[187] Boyi Li, Wenqi Ren, Dengpan Fu, Dacheng Tao, Dan Feng, Wenjun Zeng, and Zhangyang Wang. Benchmarking single-image dehazing and beyond. *IEEE Transactions on Image Processing*, 28(1):492–505, 2018.

[188] Jiale Li, Hang Dai, Hao Han, and Yong Ding. Mseg3d: Multi-modal 3d semantic segmentation for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21694–21704, 2023.

[189] Yang Li, Zhuang Li, Yanping Wang, Guangda Xie, Yun Lin, Wenjie Shen, and Wen Jiang. Improving the performance of rodnet for mmw radar target detection in dense pedestrian scene. *Mathematics*, 11(2):361, 2023.

[190] Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Fully convolutional networks for panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 214–223, 2021.

[191] Yu-Jhe Li, Jinhyung Park, Matthew O'Toole, and Kris Kitani. Modality-agnostic learning for radar-lidar fusion in vehicle detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 918–927, 2022.

[192] Zhen Li, Yukang Gan, Xiaodan Liang, Yizhou Yu, Hui Cheng, and Liang Lin. Lstm-cf: Unifying context modeling and fusion with lstms for rgb-d scene labeling. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 541–557. Springer, 2016.

[193] Zhiqi Li, Wenhai Wang, Enze Xie, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, Ping Luo, and Tong Lu. Panoptic segformer: Delving deeper into panoptic segmentation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1280–1289, 2022.

[194] Justin Liang, Namdar Homayounfar, Wei-Chiu Ma, Yuwen Xiong, Rui Hu, and Raquel Urtasun. Polytransform: Deep polygon transformer for instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9131–9140, 2020.

[195] Ming Liang, Bin Yang, Yun Chen, Rui Hu, and Raquel Urtasun. Multi-task multi-sensor fusion for 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7345–7353, 2019.

[196] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 641–656, 2018.

[197] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 641–656, 2018.

[198] Velodyn LIDAR.

[199] Guosheng Lin, Chunhua Shen, Anton Van Den Hengel, and Ian Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3194–3203, 2016.

[200] Hao Lin, Ashkan Parsi, Darragh Mullins, Jonathan Horgan, Enda Ward, Ciaran Eising, Patrick Denny, Brian Deegan, Martin Glavin, and Edward Jones. A study on data selection for object detection in various lighting conditions for autonomous vehicles. *Journal of Imaging*, 10(7):153, 2024.

[201] Chen Liu and Yasutaka Furukawa. Masc: Multi-scale affinity with sparse convolution for 3d instance segmentation. *arXiv preprint arXiv:1902.04478*, 2019.

[202] Ryan Wen Liu, Yu Guo, Yuxu Lu, Kwok Tai Chui, and Brij B. Gupta. Deep network-enabled haze visibility enhancement for visual iot-driven intelligent transportation systems. *IEEE Transactions on Industrial Informatics*, 19(2):1581–1591, 2023.

[203] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 21–37. Springer, 2016.

[204] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548. Springer, 2022.

[205] Ze Liu, Yingfeng Cai, Hai Wang, Long Chen, Hongbo Gao, Yunyi Jia, and Yicheng Li. Robust target recognition and tracking of self-driving cars with radar and camera information fusion under severe weather conditions. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):6640–6653, 2021.

[206] Zechen Liu, Zizhang Wu, and Roland Tóth. Smoke: Single-stage monocular 3d object detection via keypoint estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 996–997, 2020.

[207] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, pages 2774–2781. IEEE, 2023.

[208] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[209] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004.

[210] Tao Lu, Xiang Ding, Haisong Liu, Gangshan Wu, and Limin Wang. Link: Linear kernel for lidar-based 3d perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1105–1115, 2023.

[211] Wenjie Luo, Bin Yang, and Raquel Urtasun. Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3569–3577, 2018.

[212] Jiayi Ma, Linfeng Tang, Meilong Xu, Hao Zhang, and Guobao Xiao. Stdfusionnet: An infrared and visible image fusion network based on salient target detection. *IEEE Transactions on Instrumentation and Measurement*, 70:1–13, 2021.

[213] Andréa Macario Barros, Maugan Michel, Yoann Moline, Gwenolé Corre, and Frédérick Carrel. A comprehensive survey of visual slam algorithms. *Robotics*, 11(1):24, 2022.

[214] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.

[215] Abdelhamid Mammeri, Azzedine Boukerche, and Zongzhi Tang. A real-time lane marking localization, tracking and communication system. *Computer Communications*, 73:132–143, 2016.

[216] Jiageng Mao, Yujing Xue, Minzhe Niu, Haoyue Bai, Jiashi Feng, Xiaodan Liang, Hang Xu, and Chunjing Xu. Voxel transformer for 3d object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3164–3173, 2021.

[217] Tamás Matuszka, Iván Barton, Ádám Butykai, Péter Hajas, Dávid Kiss, Domonkos Kovács, Sándor Kunsági-Máté, Péter Lengyel, Gábor Németh, Levente Pető, et al. aimotive dataset: A multimodal dataset for robust autonomous driving with long-range perception. *arXiv preprint arXiv:2211.09445*, 2022.

[218] Jilin Mei, Biao Gao, Donghao Xu, Wen Yao, Xijun Zhao, and Huijing Zhao. Semantic segmentation of 3d lidar data in dynamic scene using semi-supervised learning. *IEEE Transactions on Intelligent Transportation Systems*, 21(6):2496–2509, 2019.

[219] Michael Meyer and Georg Kuschk. Automotive radar dataset for deep learning based 3d object detection. In *2019 16th european radar conference (EuRAD)*, pages 129–132. IEEE, 2019.

[220] Khan Muhammad, Tanveer Hussain, Hayat Ullah, Javier Del Ser, Mahdi Rezaei, Neeraj Kumar, Mohammad Hijji, Paolo Bellavista, and Victor Hugo C de Albuquerque. Vision-based semantic segmentation in scene understanding for autonomous driving: Recent achievements, challenges, and outlooks. *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[221] Ramin Nabati and Hairong Qi. Rrpn: Radar region proposal network for object detection in autonomous vehicles. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 3093–3097. IEEE, 2019.

[222] Ramin Nabati and Hairong Qi. Rrpn: Radar region proposal network for object detection in autonomous vehicles. In *2019 IEEE ICIP*, pages 3093–3097, 2019.

[223] Ramin Nabati and Hairong Qi. Centerfusion: Center-based radar and camera fusion for 3d object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1527–1536, 2021.

[224] Mozhgan Nasr Azadani and Azzedine Boukerche. A novel multimodal vehicle path prediction method based on temporal convolutional networks. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):25384–25395, 2022.

[225] Navigation National Coordination Office for Space-Based Positioning and Timing. What is gps?

[226] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE international conference on computer vision*, pages 4990–4999, 2017.

[227] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.

[228] Pedro O O Pinheiro, Ronan Collobert, and Piotr Dollár. Learning to segment object candidates. *Advances in neural information processing systems*, 28, 2015.

[229] U.S. Department of Transportation NHSTA. Canadian motor vehicle traffic collision statistics: 2022, 2023.

[230] Renê Oliveira, Carlos Montez, Azzedine Boukerche, and Michelle S Wangham. Reliable data dissemination protocol for vanet traffic safety applications. *Ad Hoc Networks*, 63:30–44, 2017.

[231] D Orth. Personalization and human-machine-interaction. 2017.

[232] Su Pang, Daniel Morris, and Hayder Radha. Clocs: Camera-lidar object candidates fusion for 3d object detection. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10386–10393. IEEE, 2020.

[233] Su Pang, Daniel Morris, and Hayder Radha. Transcar: Transformer-based camera-and-radar fusion for 3d object detection. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10902–10909. IEEE, 2023.

[234] Richard W Pazzi and Azzedine Boukerche. Propane: A progressive panorama streaming protocol to support interactive 3d virtual environment exploration on graphics-constrained devices. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 11(1):1–22, 2014.

[235] Richard WN Pazzi and Azzedine Boukerche. Mobile data collector strategy for delay-sensitive applications over wireless sensor networks. *Computer Communications*, 31(5):1028–1039, 2008.

[236] Andra Petrovai and Sergiu Nedevschi. Fast panoptic segmentation with soft attention embeddings. *Sensors*, 22(3):783, 2022.

[237] Thierry Peynot, James Underwood, and Steven Scheding. Towards reliable perception for unmanned ground vehicles in challenging conditions. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1170–1176. IEEE, 2009.

[238] Viet-Quoc Pham, Satoshi Ito, and Tatsuo Kozakaya. Biseg: Simultaneous instance segmentation and semantic segmentation with fully convolutional networks. *arXiv preprint arXiv:1706.02135*, 2017.

[239] Florian Piewak, Peter Pinggera, Manuel Schafer, David Peter, Beate Schwarz, Nick Schneider, Markus Enzweiler, David Pfeiffer, and Marius Zollner. Boosting lidar-based semantic labeling by cross-modal training data generation. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[240] Pedro O Pinheiro, Tsung-Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 75–91. Springer, 2016.

[241] Lorenzo Porzi, Samuel Rota Bulo, and Peter Kontschieder. Improving panoptic segmentation at all scales. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7302–7311, 2021.

[242] W.K. Pratt. *Laser Communication Systems*. Wiley series in pure and applied optics. Wiley, 1969.

[243] Robert Prophet, Anastasios Deligiannis, Juan-Carlos Fuentes-Michel, Ingo Weber, and Martin Vossiek. Semantic segmentation on 3d occupancy grids for automotive radar. *IEEE Access*, 8:197917–197930, 2020.

[244] Robert Prophet, Gang Li, Christian Sturm, and Martin Vossiek. Semantic segmentation on automotive radar maps. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 756–763. IEEE, 2019.

[245] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[246] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[247] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas 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*, pages 652–660, 2017.

[248] Charles R Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas J Guibas. Volumetric and multi-view cnns for object classification on 3d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5648–5656, 2016.

[249] Charles R Qi, Yin Zhou, Mahyar Najibi, Pei Sun, Khoa Vo, Boyang Deng, and Dragomir Anguelov. Offboard 3d object detection from point cloud sequences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6134–6144, 2021.

[250] Kun Qian, Shilin Zhu, Xinyu Zhang, and Li Erran Li. Robust multimodal vehicle detection in foggy weather using complementary lidar and radar signals. In *Proceedings of the IEEE/CVF CVPR*, pages 444–453, 2021.

[251] Navtech Radar.

[252] M. A. Rahim, J. Liu, Z. Zhang, L. Zhu, X. Li, and S. Khan. Reviewnet: A fast and resource optimized network for enabling safe autonomous driving in hazy weather conditions. *IEEE Sens. J*, pages 1–1, 2021.

[253] M. A. Rahim, J. Liu, Z. Zhang, L. Zhu, X. Li, and S. Khan. Robust intensity-based localization method for autonomous driving on snow–wet road surface. *IEEE Sens. J*, pages 1–1, 2021.

[254] M. A. Rahim, J. Liu, Z. Zhang, L. Zhu, X. Li, and S. Khan. Robust multimodal vehicle detection in foggyweather using complementary lidar and radar signals. *IEEE Sens. J*, pages 1–1, 2021.

[255] M. A. Rahim, J. Liu, Z. Zhang, L. Zhu, X. Li, and S. Khan. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles. *IEEE Sens. J*, pages 1–1, 2021.

[256] Ralph H Rasshofer, Martin Spies, and Hans Spies. Influences of weather phenomena on automotive laser radar systems. *Advances in radio science*, 9:49–60, 2011.

[257] Ratheesh Ravindran, Michael J Santora, and Mohsin M Jamali. Camera, lidar, and radar sensor fusion based on bayesian neural network (clr-bnn). *IEEE Sensors Journal*, 22(7):6964–6974, 2022.

[258] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[259] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[260] Mengye Ren and Richard S Zemel. End-to-end instance segmentation with recurrent attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6656–6664, 2017.

[261] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[262] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[263] Yonglin Ren, Richard Werner, Nelem Pazzi, and Azzedine Boukerche. Monitoring patients via a secure and mobile healthcare system. *IEEE Wireless Communications*, 17(1):59–65, 2010.

[264] Cristiano Rezende, Azzedine Boukerche, Heitor S Ramos, and Antonio AF Loureiro. A reactive and scalable unicast solution for video streaming over vanets. *IEEE Transactions on Computers*, 64(3):614–626, 2014.

[265] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3d representations at high resolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3577–3586, 2017.

[266] Jamie Roche, Varuna De-Silva, and Ahmet Kondoz. A multimodal perception-driven self evolving autonomous ground vehicle. *IEEE Transactions on Cybernetics*, 52(9):9279–9289, 2021.

[267] Hermann Rohling and M-M Meinecke. Waveform design principles for automotive radar systems. In *2001 CIE International Conference on Radar Proceedings (Cat No. 01TH8559)*, pages 1–4. IEEE, 2001.

[268] Eduardo Romera, Luis M Bergasa, Kailun Yang, Jose M Alvarez, and Rafael Barea. Bridging the day and night domain gap for semantic segmentation. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1312–1318. IEEE, 2019.

[269] Bernardino Romera-Paredes and Philip Hilaire Sean Torr. Recurrent instance segmentation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14*, pages 312–329. Springer, 2016.

[270] Ricardo Roriz, André Campos, Sandro Pinto, and Tiago Gomes. Dior: A hardware-assisted weather denoising solution for lidar point clouds. *IEEE Sensors Journal*, 22(2):1621–1628, 2021.

[271] Danila Rukhovich, Anna Vorontsova, and Anton Konushin. Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2397–2406, 2022.

[272] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7374–7383, 2019.

[273] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6):3139–3153, 2020.

[274] Nicolas Scheiner, Florian Kraus, Fangyin Wei, Buu Phan, Fahim Mannan, Nils Appenrodt, Werner Ritter, Jurgen Dickmann, Klaus Dietmayer, Bernhard Sick, et al. Seeing around street corners: Non-line-of-sight detection and tracking in-the-wild using doppler radar. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2068–2077, 2020.

[275] Markus Schön, Michael Buchholz, and Klaus Dietmayer. Mgnet: Monocular geometric scene understanding for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15804–15815, 2021.

[276] Ole Schumann, Markus Hahn, Nicolas Scheiner, Fabio Weishaupt, Julius F Tilly, Jürgen Dickmann, and Christian Wöhler. Radarscenes: A real-world radar point cloud data set for automotive applications. In *2021 IEEE 24th International Conference on Information Fusion (FUSION)*, pages 1–8. IEEE, 2021.

[277] Qusay Sellat and Kanagachidambaresan Ramasubramanian. Advanced techniques for perception and localization in autonomous driving systems: A survey. *Optical Memory and Neural Networks*, 31(2):123–144, 2022.

[278] Continental Engineering Services. Ars 408-21 long range radar sensor 77 ghz - data sheet.

[279] Abu Ubaidah Shamsudin, Kazunori Ohno, Thomas Westfechtel, Suzuki Takahiro, Yoshito Okada, and Satoshi Tadokoro. Fog removal using laser beam penetration, laser intensity, and geometrical features for 3d measurements in fog-filled room. *Advanced robotics*, 30(11-12):729–743, 2016.

[280] Marcel Sheeny, Emanuele De Pellegrin, Saptarshi Mukherjee, Alireza Ahrabian, Sen Wang, and Andrew Wallace. Radiate: A radar dataset for automotive perception in bad weather. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7. IEEE, 2021.

[281] H Sheikh. Live image quality assessment database release 2. *http://live. ece. utexas. edu/research/quality*, 2005.

[282] Ju Shen and Sen-Ching S Cheung. Layer depth denoising and completion for structured-light rgb-d cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1187–1194, 2013.

[283] Jianbo Shi et al. Good features to track. In *1994 Proceedings of IEEE conference on computer vision and pattern recognition*, pages 593–600. IEEE, 1994.

[284] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 770–779, 2019.

[285] Rathin Chandra Shit. Precise localization for achieving next-generation autonomous navigation: State-of-the-art, taxonomy and future prospects. *Computer Communications*, 160:351–374, 2020.

[286] Jamie Shotton, Matthew Johnson, and Roberto Cipolla. Semantic texton forests for image categorization and segmentation. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008.

[287] Mennatullah Siam, Mostafa Gamal, Moemen Abdel-Razek, Senthil Yogamani, Martin Jagersand, and Hong Zhang. A comparative study of real-time semantic segmentation for autonomous driving. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 587–597, 2018.

[288] Vishwanath A Sindagi, Yin Zhou, and Oncel Tuzel. Mvx-net: Multimodal voxelnet for 3d object detection. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7276–7282. IEEE, 2019.

[289] Liat Sless, Bat El Shlomo, Gilad Cohen, and Shaul Oron. Road scene understanding by occupancy grid learning from sparse radar clusters using semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[290] Michael Slutsky and Daniel Dobkin. Dual inverse sensor model for radar occupancy grids. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1760–1767. IEEE, 2019.

[291] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015.

[292] Shuran Song and Jianxiong Xiao. Deep sliding shapes for amodal 3d object detection in rgb-d images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 808–816, 2016.

[293] Roniel S De Sousa, Azzedine Boukerche, and Antonio AF Loureiro. Vehicle trajectory similarity: Models, methods, and applications. *ACM Computing Surveys (CSUR)*, 53(5):1–32, 2020.

[294] AG Stove. Modern fmcw radar-techniques and applications. In *First European Radar Conference, 2004. EURAD.*, pages 149–152. IEEE, 2004.

[295] Andrew G Stove. Linear fmcw radar techniques. In *IEE Proceedings F (Radar and Signal Processing)*, volume 139, pages 343–350. IET, 1992.

[296] Paul Sturgess, Karteek Alahari, Lubor Ladicky, and Philip HS Torr. Combining appearance and structure from motion features for road scene understanding. In *BMVC-British Machine Vision Conference*. BMVA, 2009.

[297] Bo Sun, Jason Kuen, Zhe Lin, Philippos Mordohai, and Simon Chen. Prn: Panoptic refinement network. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3963–3973, 2023.

[298] Lei Sun, Kaiwei Wang, Kailun Yang, and Kaite Xiang. See clearer at night: towards robust nighttime semantic segmentation through day-night image conversion. In *Artificial Intelligence and Machine Learning in Defense Applications*, volume 11169, pages 77–89. SPIE, 2019.

[299] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020.

[300] Peng Sun, Noura Aljeri, and Azzedine Boukerche. Machine learning-based models for real-time traffic flow prediction in vehicular networks. *IEEE Network*, 34(3):178–185, 2020.

[301] Peng Sun, Azzedine Boukerche, and Yanjie Tao. Ssgru: A novel hybrid stacked gru-based traffic volume prediction approach in a road network. *Computer Communications*, 160:502–511, 2020.

[302] Yuxiang Sun, Weixun Zuo, and Ming Liu. Rtfnet: Rgb-thermal fusion network for semantic segmentation of urban scenes. *IEEE Robotics and Automation Letters*, 4(3):2576–2583, 2019.

[303] Karasawa Takumi, Kohei Watanabe, Qishen Ha, Antonio Tejero-De-Pablos, Yoshitaka Ushiku, and Tatsuya Harada. Multispectral object detection for autonomous vehicles. In *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*, pages 35–43, 2017.

[304] Maxim Tatarchenko and Kilian Rambach. Histogram-based deep learning for automotive radar. In *2023 IEEE Radar Conference (RadarConf23)*, pages 1–6. IEEE, 2023.

[305] Lyne Tchapmi, Christopher Choy, Iro Armeni, JunYoung Gwak, and Silvio Savarese. Segcloud: Semantic segmentation of 3d point clouds. In *2017 international conference on 3D vision (3DV)*, pages 537–547. IEEE, 2017.

[306] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *IEEE/CVF ICCV*, pages 9627–9636, 2019.

[307] Alexander Toet. The tno multiband image data collection. *Data in brief*, 15:249–251, 2017.

[308] Patrik Tosteberg. Semantic segmentation of point clouds using deep learning, 2017.

[309] Jonas Uhrig, Marius Cordts, Uwe Franke, and Thomas Brox. Pixel-level encoding and depth layering for instance-level semantic labeling. In *Pattern Recognition: 38th German Conference, GCPR 2016, Hannover, Germany, September 12-15, 2016, Proceedings 38*, pages 14–25. Springer, 2016.

[310] Jorge Vargas, Suleiman Alsweiss, Onur Toker, Rahul Razdan, and Joshua Santos. An overview of autonomous vehicles sensors and their vulnerability to weather conditions. *Sensors*, 21(16):5397, 2021.

[311] N Venkatanath, D Praneeth, Maruthi Chandrasekhar Bh, Sumohana S Channappayya, and Swarup S Medasani. Blind image quality evaluation using perception based features. In *2015 twenty first national conference on communications (NCC)*, pages 1–6. IEEE, 2015.

[312] Leandro A Villas, Azzedine Boukerche, Horacio ABF De Oliveira, Regina B De Araujo, and Antonio AF Loureiro. A spatial correlation aware algorithm to perform efficient data collection in wireless sensor networks. *Ad Hoc Networks*, 12:69–85, 2014.

[313] Leandro A Villas, Azzedine Boukerche, Daniel L Guidoni, Horacio ABF De Oliveira, Regina Borges De Araujo, and Antonio AF Loureiro. An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks. *Computer Communications*, 36(9):1054–1066, 2013.

[314] Leandro Aparecido Villas, Azzedine Boukerche, Guilherme Maia, Richard Werner Pazzi, and Antonio AF Loureiro. Drive: An efficient and robust data dissemination protocol for highway and urban vehicular ad hoc networks. *Computer Networks*, 75:381–394, 2014.

[315] Leandro Aparecido Villas, Azzedine Boukerche, Heitor Soares Ramos, Horacio AB Fernandes De Oliveira, Regina Borges De Araujo, and Antonio Alfredo Ferreira Loureiro. Drina: A lightweight and reliable routing approach for in-network aggregation in wireless sensor networks. *IEEE Transactions on Computers*, 62(4):676–689, 2012.

[316] Amanpreet Walia, Stefanie Walz, Mario Bijelic, Fahim Mannan, Frank Julca-Aguilar, Michael Langer, Werner Ritter, and Felix Heide. Gated2gated: Self-supervised depth estimation from gated images. *arXiv preprint arXiv:2112.02416*, 2021.

[317] Dominic Zeng Wang and Ingmar Posner. Voting for voting in online point cloud object detection. In *Robotics: science and systems*, volume 1, pages 10–15. Rome, Italy, 2015.

[318] Hai Wang, Yanyan Chen, Yingfeng Cai, Long Chen, Yicheng Li, Miguel Angel Sotelo, and Zhixiong Li. Sfnet-n: An improved sfnet algorithm for semantic segmentation of low-light autonomous driving road scenes. *IEEE Transactions on Intelligent Transportation Systems*, 23(11):21405–21417, 2022.

[319] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 913–922, 2021.

[320] Wei Wang, Kaicheng Yu, Joachim Hugonot, Pascal Fua, and Mathieu Salzmann. Recurrent u-net for resource-constrained segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2142–2151, 2019.

[321] Weiyue Wang, Ronald Yu, Qiangui Huang, and Ulrich Neumann. Sgpn: Similarity group proposal network for 3d point cloud instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2569–2578, 2018.

[322] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Solo: A simple framework for instance segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 44(11):8587–8601, 2021.

[323] Zhangjing Wang, Yu Wu, and Qingqing Niu. Multi-sensor fusion in automated driving: A survey. *Ieee Access*, 8:2847–2868, 2019.

[324] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[325] Zining Wang, Wei Zhan, and Masayoshi Tomizuka. Fusing bird's eye view lidar point cloud and front view camera image for 3d object detection. In *2018 IEEE intelligent vehicles symposium (IV)*, pages 1–6. IEEE, 2018.

[326] Klaudius Werber, Matthias Rapp, Jens Klappstein, Markus Hahn, Jürgen Dickmann, Klaus Dietmayer, and Christian Waldschmidt. Automotive radar gridmap representations. In *2015 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*, pages 1–4. IEEE, 2015.

[327] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting. *arXiv preprint arXiv:2301.00493*, 2023.

[328] Philipp Wolters, Johannes Gilg, Torben Teepe, Fabian Herzog, Anouar Laouichi, Martin Hofmann, and Gerhard Rigoll. Unleashing hydra: Hybrid fusion, depth consistency and radar for unified 3d perception. *arXiv preprint arXiv:2403.07746*, 2024.

[329] Uland Wong, Aaron Morris, Colin Lea, James Lee, Chuck Whittaker, Ben Garney, and Red Whittaker. Comparative evaluation of range sensing technologies for underground void modeling. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3816–3823. IEEE, 2011.

[330] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 1887–1893. IEEE, 2018.

[331] Xinyi Wu, Zhenyao Wu, Hao Guo, Lili Ju, and Song Wang. Dannet: A one-stage domain adaptation network for unsupervised nighttime semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15769–15778, 2021.

[332] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015.

[333] Yuxing Xie, Jiaojiao Tian, and Xiao Xiang Zhu. Linking points with labels in 3d: A review of point cloud semantic segmentation. *IEEE Geoscience and remote sensing magazine*, 8(4):38–59, 2020.

[334] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8818–8826, 2019.

[335] Danfei Xu, Dragomir Anguelov, and Ashesh Jain. Pointfusion: Deep sensor fusion for 3d bounding box estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 244–253, 2018.

[336] Han Xu, Jiayi Ma, Junjun Jiang, Xiaojie Guo, and Haibin Ling. U2fusion: A unified unsupervised image fusion network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):502–518, 2020.

[337] Shaoqing Xu, Dingfu Zhou, Jin Fang, Junbo Yin, Zhou Bin, and Liangjun Zhang. Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 3047–3054. IEEE, 2021.

[338] Shaoqing Xu, Dingfu Zhou, Jin Fang, Junbo Yin, Zhou Bin, and Liangjun Zhang. Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 3047–3054. IEEE, 2021.

[339] Junjie Yan, Yingfei Liu, Jianjian Sun, Fan Jia, Shuailin Li, Tiancai Wang, and Xiangyu Zhang. Cross modal transformer: Towards fast and robust 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18268–18278, 2023.

[340] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

[341] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018.

[342] Bin Yang, Runsheng Guo, Ming Liang, Sergio Casas, and Raquel Urtasun. Radarnet: Exploiting radar for robust perception of dynamic objects. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 496–512, Cham, 2020. Springer International Publishing.

[343] Bin Yang, Runsheng Guo, Ming Liang, Sergio Casas, and Raquel Urtasun. Radarnet: Exploiting radar for robust perception of dynamic objects. In *Computer*

Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16, pages 496–512. Springer, 2020.

[344] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17830–17839, 2023.

[345] Xitong Yang, Zheng Xu, and Jiebo Luo. Towards perceptual image dehazing by physics-based disentanglement and adversarial training. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[346] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11040–11048, 2020.

[347] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. *Advances in Neural Information Processing Systems*, 35:1992–2005, 2022.

[348] Cang Ye and Johann Borenstein. Characterization of a 2d laser scanner for mobile robot obstacle negotiation. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, volume 3, pages 2512–2518. IEEE, 2002.

[349] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021.

[350] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021.

[351] Keisuke Yoneda, Naoya Hashimoto, Ryo Yanase, Mohammad Aldibaja, and Naoki Suganuma. Vehicle localization using 76ghz omnidirectional millimeter-wave radar for winter automated driving. In *2018 IEEE intelligent vehicles symposium (IV)*, pages 971–977. IEEE, 2018.

[352] Maram Bani Younes and Azzedine Boukerche. Intelligent traffic light controlling algorithms using vehicular networks. *IEEE transactions on vehicular technology*, 65(8):5887–5899, 2015.

[353] Maram Bani Younes and Azzedine Boukerche. A performance evaluation of an efficient traffic congestion detection protocol (ecode) for intelligent transportation systems. *Ad Hoc Networks*, 24:317–336, 2015.

[354] Maram Bani Younes and Azzedine Boukerche. An efficient dynamic traffic light scheduling algorithm considering emergency vehicles for intelligent transportation systems. *Wireless Networks*, 24:2451–2463, 2018.

[355] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020.

[356] Jun Yu, Xinlong Hao, Xinjian Gao, Qiang Sun, Yuyu Liu, Peng Chang, Zhong Zhang, Fang Gao, and Feng Shuang. Radar object detection using data merging, enhancement and fusion. In *Proceedings of the 2021 International Conference on Multimedia Retrieval*, pages 566–572, 2021.

[357] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. A survey of autonomous driving: Common practices and emerging technologies. *IEEE access*, 8:58443–58469, 2020.

[358] Shizhe Zang, Ming Ding, David Smith, Paul Tyler, Thierry Rakotoarivelo, and Mohamed Ali Kaafar. The impact of adverse weather conditions on autonomous vehicles: How rain, snow, fog, and hail affect the performance of a self-driving car. *IEEE vehicular technology magazine*, 14(2):103–111, 2019.

[359] Diankun Zhang, Zhijie Zheng, Haoyu Niu, Xueqing Wang, and Xiaojun Liu. Fully sparse transformer 3d detector for lidar point cloud. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

[360] Feihu Zhang, Chenye Guan, Jin Fang, Song Bai, Ruigang Yang, Philip HS Torr, and Victor Prisacariu. Instance segmentation of lidar point clouds. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9448–9455. IEEE, 2020.

[361] Shengdong Zhang, Wenqi Ren, Xin Tan, Zhi-Jie Wang, Yong Liu, Jingang Zhang, Xiaoqin Zhang, and Xiaochun Cao. Semantic-aware dehazing network with adaptive feature fusion. *IEEE Transactions on Cybernetics*, 53(1):454–467, 2021.

[362] Shengdong Zhang, Wenqi Ren, Xin Tan, Zhi-Jie Wang, Yong Liu, Jingang Zhang, Xiaoqin Zhang, and Xiaochun Cao. Semantic-aware dehazing network with adaptive feature fusion. *IEEE Transactions on Cybernetics*, 53(1):454–467, 2021.

[363] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z Li. Single-shot refinement neural network for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4203–4212, 2018.

[364] Tao Zhang, Shiqing Wei, and Shunping Ji. E2ec: An end-to-end contour-based method for high-quality high-speed instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4443–4452, 2022.

[365] Yanfu Zhang, Li Ding, and Gaurav Sharma. Hazerd: an outdoor scene dataset and benchmark for single image dehazing. In *2017 IEEE international conference on image processing (ICIP)*, pages 3205–3209. IEEE, 2017.

[366] Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3289–3298, 2021.

[367] Yuxiao Zhang, Alexander Carballo, Hanting Yang, and Kazuya Takeda. Perception and sensing for autonomous vehicles under adverse weather conditions: A survey. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:146–177, 2023.

[368] Zhenxia Zhang, Azzedine Boukerche, and Richard Pazzi. A novel multi-hop clustering scheme for vehicular ad-hoc networks. In *Proceedings of the 9th ACM international symposium on Mobility management and wireless access*, pages 19–26, 2011.

[369] Ziyu Zhang, Alexander G Schwing, Sanja Fidler, and Raquel Urtasun. Monocular object instance segmentation and depth ordering with cnns. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2614–2622, 2015.

[370] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[371] Dingfu Zhou, Jin Fang, Xibin Song, Liu Liu, Junbo Yin, Yuchao Dai, Hongdong Li, and Ruigang Yang. Joint 3d instance segmentation and object detection for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1839–1849, 2020.

[372] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.

[373] Yi Zhou, Lulu Liu, Haocheng Zhao, Miguel López-Benítez, Limin Yu, and Yutao Yue. Towards deep radar perception for autonomous driving: Datasets, methods, and challenges. *Sensors*, 22(11):4208, 2022.

[374] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *IEEE/CVF CVPR*, pages 4490–4499, 2018.

[375] Zixiang Zhou, Xiangchen Zhao, Yu Wang, Panqu Wang, and Hassan Foroosh. Centerformer: Center-based transformer for 3d object detection. In *European Conference on Computer Vision*, pages 496–513. Springer, 2022.

[376] Zhuangwei Zhuang, Rong Li, Kui Jia, Qicheng Wang, Yuanqing Li, and Mingkui Tan. Perception-aware multi-sensor fusion for 3d lidar semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16280–16290, 2021.

[377] Jian Zou, Tianyu Huang, Guanglei Yang, Zhenhua Guo, and Wangmeng Zuo. Unim2ae: Multi-modal masked autoencoders with unified 3d representation for 3d perception in autonomous driving. *arXiv preprint arXiv:2308.10421*, 2023.