Towards Zero Bottlenecks for Scaling Autonomous Driving

Tonderski, Adam

2025

– CENTRUM SCIENTIARUM MATHEMATICARUM –

Towards Zero Bottlenecks for Scaling Autonomous Driving

ADAM TONDERSKI

Lund University
Faculty of Engineering
Centre for Mathematical Sciences
Mathematics

Towards Zero Bottlenecks for Scaling Autonomous Driving

by Adam Tonderski

LUND
UNIVERSITY

Thesis for the degree of Doctor of Philosophy in Engineering

Thesis advisors:
Prof. Kalle Åström, Assoc. Prof. Christoffer Petersson

Faculty opponent:
Prof. Felix Heide, Princeton University, US

To be presented, with the permission of the Faculty of Engineering of Lund University, for public criticism in the lecture hall (MH:G) at the Centre of Mathematical Sciences on Friday, the 28th of February 2025 at 13:15.

Organization	Document name
LUND UNIVERSITY Centre of Mathematical Sciences Box 118 SE–221 00 LUND Sweden	**DOCTORAL DISSERTATION** Date of disputation 2025-02-28 Sponsoring organization
Author(s) Adam Tonderski	

Title and subtitle
Towards Zero Bottlenecks for Scaling Autonomous Driving

Abstract
In this dissertation I examine the main scaling challenges in autonomous driving development, discussing recent advances in the field while contributing specific solutions to key bottlenecks. The first challenge is the reliance on human labor, particularly for annotations. Here we make two key contributions: new techniques to extract additional value from existing annotations through future prediction (I), and an adaptation of vision-language learning to 3D automotive sensors that reduces dependence on explicit labels while maintaining interpretability (II). The second challenge concerns access to training data covering the full spectrum of driving scenarios. We address this data bottleneck through complementary approaches: releasing a diverse European driving dataset collected across multiple years and conditions (III), and developing a neural rendering method that enables scalable generation of realistic synthetic data (IV). Finally, to enable scalable safety testing, we introduce a closed-loop neural simulator that transforms ordinary driving scenarios into challenging near-collision cases (V). Together with broader advances in the field, our contributions suggest a promising path toward scaling autonomous vehicle development.

Key words
autonomous driving, perception, vision-language, neural rendering, simulation

Classification system and/or index terms (if any)

Supplementary bibliographical information	Language
	English
ISSN and key title 1404-0034. Doctoral Theses in Mathematical Sciences	ISBN 978-91-8104-298-6 (print) 978-91-8104-299-3 (pdf)
Recipient's notes	Number of pages 193 / Price
	Security classification

Towards Zero Bottlenecks for Scaling Autonomous Driving

by Adam Tonderski

LUND
UNIVERSITY

Cover illustration: A drawing symbolizing how the world of self-driving escapes from the (scaling) bottle-neck. The image is an iterative collaboration between Linnea and I, using various generative AI systems (Flux, ChatGPT+Dall-E, FreePik).

Funding information: This thesis work was supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation, and by Zenseact AB through their industrial PhD program.

Faculty of Engineering, Centre of Mathematical Sciences

Doctoral Theses in Mathematical Sciences 2025:1
ISSN: 1404-0034
ISBN: 978-91-8104-298-6 (print)
ISBN: 978-91-8104-299-3 (pdf)
LUTFTM-1001-2025

Printed in Sweden by Media-Tryck, Lund University, Lund 2025

And that's all I have to say about that.
– Forrest Gump

Abstract

In this dissertation I examine the main scaling challenges in autonomous driving development, discussing recent advances in the field while contributing specific solutions to key bottlenecks. The first challenge is the reliance on human labor, particularly for annotations. Here we make two key contributions: new techniques to extract additional value from existing annotations through future prediction (I), and an adaptation of vision-language learning to 3D automotive sensors that reduces dependence on explicit labels while maintaining interpretability (II). The second challenge concerns access to training data covering the full spectrum of driving scenarios. We address this data bottleneck through complementary approaches: releasing a diverse European driving dataset collected across multiple years and conditions (III), and developing a neural rendering method that enables scalable generation of realistic synthetic data (IV). Finally, to enable scalable safety testing, we introduce a closed-loop neural simulator that transforms ordinary driving scenarios into challenging near-collision cases (V). Together with broader advances in the field, our contributions suggest a promising path toward scaling autonomous vehicle development.

Popular Science Summary

The dream of self-driving cars is swiftly transitioning from science fiction to reality, largely due to rapid advancements in deep neural networks. However, the performance of these networks is inherently constrained by the quality and quantity of their training data and supervision. Achieving each seemingly incremental improvement in safety often demands an order-of-magnitude scaling of data and computational resources. In this thesis, I identify key bottlenecks that hinder this scaling – often boiling down to the human-in-the-loop – and propose novel solutions to overcome them.

Typical autonomous driving stacks consist of multiple specialized modules, like object detection and road model estimation, that heavily depend on human-annotated data. Such manual labeling is extremely laborious and expensive, so naturally we want to extract maximum value from each annotation. Therefore the first question we address in this thesis is how to increase the value of existing human annotations. We find that by training networks to predict future annotations from past sensor data, we force the model to reason about dynamics and depth and even implicitly perform 3D object tracking. This simple approach enables the model to unlock new capabilities, extracting insights beyond the original annotations without any additional human input.

But what if we can dispense with human-generated labels entirely? An intriguing approach involves moving away from predefined categories such as 'car' or 'pedestrian,' and instead using unconstrained natural language. This shift has been well-explored in the domain of image-language models, but our work extends this to lidar — a critical sensor in self-driving vehicles. Lidar devices measure the distance to surrounding objects by emitting laser pulses and recording the light's return time. Our model allows these resulting point clouds to be queried with natural language, facilitating the exploration and understanding of complex 3D environments without relying on detailed manual annotations or predefined label categories. With the recent advancements in large language models, which now include image understanding, we are optimistic that some version of our approach will enable these models to also reason about lidar data, and possibly other automotive sensors as well.

The next bottleneck in developing autonomous driving systems is, of course, the data itself. Given that neural networks are notoriously slow learners, needing huge amounts of data to learn seemingly simple concepts, we need to ensure that the training set covers every imaginable scenario – from snowstorms in northern sweden to nightime driving in busy street in downtown paris. Since existing open datasets often fall short in this regard, we release a comprehensive dataset collected across Europe over several years, curated to contain a diverse set of driving conditions and environments.

However, certain scenarios, such as near-miss accidents and encounters with rare obstacles, are too infrequent or hazardous to capture through traditional data collection methods.

Typically, training on such cases would involve generating synthetic data using a mix of human creativity and procedural generation within a game engine. Yet, this method requires extensive human effort and often fails to represent the diversity of the real world accurately. To address this, we develop a neural renderer capable of transforming real-world driving logs into sensor-realistic, interactive 3D environments. This allows us to significantly expand the coverage of our dataset, by generating a suite of rare and diverse scenarios from a single boring driving log – in a safe and scalable way.

By integrating these strategies – gathering high-quality datasets, maximizing the value of human supervision, leveraging self-supervised learning (potentially with language models), and employing neural simulation to enrich our data with rare scenarios – we can significantly enhance autonomous vehicle development efficiency. This holistic approach reduces dependence on manual annotation while broadening training data scope and depth, enabling systems that can better navigate real-world driving complexities.

A crucial question remains: how can we verify the safety of the models we develop? In response, we have introduced NeuroNCAP, a closed-loop simulator that employs our neural renderer to transform ordinary driving scenes into potential near-crash scenarios. This tool allows for rigorous stress testing of autonomous systems, uncovering potential safety issues that conventional testing might miss, and allowing developers to assess and refine system safety before public road deployment.

While many open questions and challenges remain in autonomous vehicle technology, the bitter lesson suggests that success ultimately depends on scaling. By painting a broad picture of the necessary pieces and proposing novel solutions to the most critical bottlenecks, we hope to move one step closer to realizing the dream of fully autonomous vehicles safely navigating every street in every country.

List of Publications

This thesis is based on the following publications, referred to by their Roman numerals:

I **Future Object Detection with Spatiotemporal Transformers**

A. Tonderski, J. Johnander, C. Petersson, K. Åström
European Conference on Computer Vision Workshop on "What is Motion For?" (ECCV Workshop) 2022
AT and JJ developed the method, performed experiments (mostly AT), and wrote most of the paper. CP and KÅ provided advice and feedback on the project.

II **LidarCLIP or: How I Learned to Talk to Point Clouds**

G. Hess*, **A. Tonderski***, K. Åström, L. Svensson, C. Petersson
Winter Conference on Applications of Computer Vision (WACV) 2024

AT and GH (shared first authorship) conceived the idea, developed the method, performed experiments, and wrote the paper. KÅ, LS, and CP provided feedback.

III **Zenseact Open Dataset: A large-scale and diverse multimodal dataset for autonomous driving**

M. Alibeigi*, W. Ljungbergh*, **A. Tonderski***, G. Hess, A. Lilja, C. Lindström, D. Motorniuk, J. Fu, J. Widahl, C. Petersson
International Conference on Computer Vision (ICCV) 2023

MA, WL, and AT share first authorship. MA organized the project. AT and WL performed most data mining and extraction. AT, WL, GH, CL worked on devkit, analysis and experiments. All co-authors contributed to the writing.

IV **NeuRAD: Neural Rendering for Autonomous Driving**

A. Tonderski*, C. Lindström*, G. Hess*, W. Ljungbergh, L. Svensson, C. Petersson
Conference on Computer Vision and Pattern Recognition (CVPR) 2024

AT, GH, and CL share first authorship. They developed the method and performed experiments, with assistance from WL. AT, CL, and GH wrote the manuscript, with input from all co-authors. CP and LS provided key feedback.

V **NeuroNCAP: Photorealistic Closed-loop Safety Testing for Autonomous Driving**

W. Ljungbergh*, **A. Tonderski***, J. Johnander, H. Caesar, K. Åström, M. Felsberg, C. Petersson
European Conference on Computer Vision (ECCV) 2024

AT and WL share first authorship, developed the framework and executed experiments. AT focused on rendering and WL on motion models and E2E models. AT, WL and CP conceived the idea. Writing was done by AT, WL and JJ, with contributions from everyone. CP, JJ, MF, KÅ and HC provided invaluable advice.

Acknowledgements

This journey started a long time ago (pre covid) in a dark meeting room far, far away (Zenseact offices)... when Christoffer randomly mentioned that he was involved in a WASP application for industrial PhD students. One thing led to another and in the span of only a week or two we had met Kalle, submitted an application and off we went! Now, almost 5 years later, this adventure is coming to an end and I want to thank you, Kalle and Christoffer, for a truly awesome time. You have provided excellent advice, been extremely supportive, always encouraging me to follow my passion and interests[1], which while admittedly not at all following the original project plan has worked out quite all right in the end, I think.

I want to thank all the wonderful people I have met over the years – at conferences, study trips, or simply in the hallways at Zenseact or Lund. Many of these encounters were thanks to the amazing WASP program, which has fostered an excellent community here in Sweden through its courses, international study trips, and other fun activities. I particularly want to thank everyone in my fellow AGP gang (the Zenseact PhD student group), as well as Carl, Mats, and Jonas for their support, coaching, leadership, and for sending me on so many incredible opportunities around the world. To the senior students (now PhDs) who showed me the way -— especially Jocke J, who taught me the research ropes, and Magnus, who taught me the art of the patent. And of course, Georg, William, Carl, and Adam – deadline all-nighters, countless bug hunts, frantic whiteboard brainstorming, and traveling the world – it has simply been a blast!

I also want to express my gratitude to all my coauthors – your contributions and collaborations have been essential to this work. Among them, a special thank you to Lennart, who often felt like an unofficial co-supervisor due to our close collaboration on several papers. To my colleagues at Zenseact, thank you for the engaging discussions, for providing practical perspectives, and for showing genuine interest in my research. A particular thanks to Jocke B – our Master's thesis gave me a taste for more and set me on this path. While we didn't get the paper accepted, in hindsight we were simply too far ahead of our time! To my fellow academics at the Centre of Mathematics, thank you for always welcoming me with open arms and providing a balancing perspective on things. I wish I had been around more. In the wise words of Bilbo: "I don't know half of you half as well as I should like, and I like less than half of you half as well as you deserve."

Last but not least I want to thank my family and friends who, as always, supported me and made these years a true pleasure. Especially in the first year of my thesis, when covid was putting the world somewhat upside down – never forget lunch counter-strike and friday afternoon-padel! Special thanks to my mama and papa, who didn't raise no quitter

[1] *cough* LidarCLIP *cough*

- and to mom especially for providing excellent feedback on this thesis. To my brother David for always showing interest and engaging in great discussions. Last but definitely not least, thank you to my better half, Linnea. For your support since before day one, always providing your design expertize, putting up with me during deadline weeks, and most of all for making these years full of joy and great memories.

Funding

This thesis work was supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation, and by Zenseact AB through their industrial PhD program.

Contents

Chapter 1

Introduction

The development of autonomous driving systems stands as one of the most complex and impactful challenges in the fields of artificial intelligence and robotics, requiring solutions to problems in perception, decision-making, and control in dynamic, real-world environments. These systems heavily rely on machine learning techniques, particularly supervised learning, to navigate the intricate and dynamic environments encountered on roads. However, the traditional supervised learning paradigm faces two significant bottlenecks: the requirement for vast quantities of human-annotated data (the supervision bottleneck) [1] and the need for diverse, high-quality data representing a wide range of driving scenarios (the data bottleneck). This thesis explores both established and emerging approaches to these fundamental bottlenecks, while providing several key contributions that advance the state of the art in scalable autonomous driving development.

One avenue that has been explored in recent research is to increase the efficiency of human supervision by selectively applying it where it provides most value. The autonomous driving community has made significant progress in this direction through active learning and automatic annotation techniques [2, 3]. Active learning methods identify the most informative or uncertain samples for human annotation, ensuring that manual labeling efforts are focused on the most challenging examples. Complementing this, automatic annotation techniques, often powered by large offline models, can process vast amounts of data to generate initial labels [4, 5].

This combination of targeted human supervision and large-scale automatic labeling creates a powerful synergy. Human annotators provide high-quality labels for critical or ambiguous cases, while automatic systems handle the bulk of more straightforward annotations. The resulting dataset strikes a balance between the precision of human expertise and the scale of machine-generated labels. Moreover, these techniques can be integrated into an iterative process, where the model's performance continuously improves as it learns from

both human-verified and automatically generated labels, ultimately accelerating the development of robust autonomous driving systems.

Another approach to increase efficiency focuses on extracting more value from each annotation through novel training methodologies. We contribute to this direction through the development of a method that repurposes 2D annotations to train neural networks about world dynamics (Paper I). This approach leverages pre-existing annotations to extract additional value from data that has already been labeled, demonstrating how the field can progress without always requiring fresh annotation efforts. By finding new ways to utilize existing annotations, this method effectively multiplies the impact of human labeling effort, complementing the advances in active learning and automatic annotation.

While these approaches optimize the use of human annotations, the field has increasingly sought methods that can learn from raw data without any human supervision at all. Self-supervised learning has emerged as a promising direction for addressing the annotation bottleneck [6–9]. This approach extracts knowledge directly from raw data without human input, enabling the development of foundation models, i.e. large generalist models that can be fine-tuned for specific tasks with minimal labeled data. A particularly powerful form of self-supervision has emerged through the joint learning of vision and language from web-scale image-text pairs [10], leading to models with rich understanding of both modalities. This approach represents an intriguing middle ground, moving away from rigid predefined categories toward more flexible and expressive descriptions while still maintaining a form of human guidance and interpretability through natural language supervision.

Building on these advances, we introduce LidarCLIP (Paper II), which adapts the CLIP (Contrastive Language-Image Pre-training) [10] model to work with lidar data. LidarCLIP learns to associate point cloud data with natural language descriptions, enabling zero-shot transfer to various 3D understanding tasks. By combining language supervision with rich sensor data, LidarCLIP demonstrates the potential of creating powerful foundation models for autonomous driving, enhancing capabilities in scene understanding, anomaly detection, and human-AI interaction.

These complementary approaches to the supervision bottleneck – optimizing human annotation through active learning and automatic labeling, extracting additional value from existing annotations, and developing self-supervised learning methods – each contribute to reducing the dependency on extensive manual labeling. Together, they suggest a path toward more scalable development of autonomous driving systems. However, despite these techniques – or rather because of them – we encounter a more fundamental bottleneck: the scarcity of available high-quality autonomous driving data.

To address this challenge, we introduce a novel dataset (Paper III) that complements existing autonomous driving datasets with a focus on data diversity. Collected across multiple years and spanning all of Europe, our dataset captures a wide range of driving scenarios under

diverse environmental conditions, seasons, and cultural contexts. This comprehensive approach significantly enhances the robustness and generalization capabilities of autonomous driving models, providing crucial benefits for real-world applications.

As data collection for autonomous driving expands, it also encounters inherent limitations. Critical scenarios in real-world driving are often too rare to gather at a meaningful scale, and achieving broader operational design domains requires an ever-increasing amount of data to cover increasingly rare edge cases. Traditional approaches to address these challenges rely on synthetic data generation through game engines and computer graphics-based simulators, with modular systems typically undergoing closed-loop testing in simplified object-level environments. However, modern end-to-end driving models require realistic sensor simulation for effective evaluation, making traditional approaches costly and time-consuming due to their reliance on substantial input from artists and designers.

Recent advancements in neural rendering offer a compelling alternative: learning a simulator directly from data using differentiable rendering techniques [11]. Building upon these advancements, we introduce NeuRAD (Paper IV) – a neural simulator that demonstrates state-of-the-art performance in both camera and lidar reconstruction across major autonomous driving datasets. This data-driven approach enables more efficient and scalable generation of realistic synthetic data, addressing the limitations of traditional simulation methods.

The final frontier we tackle is the generation of data that cannot be safely or practically collected by humans at scale. This includes rare scenarios, unexpected behaviours, and near-crash situations that are too dangerous to recreate in the real world. To address this critical gap, we introduce NeuroNCAP (Paper V), a closed-loop safety-testing evaluation framework that transforms ordinary driving scenes into near-crash scenarios using neural rendering techniques. By subjecting state-of-the-art end-to-end driving models [12, 13] to these challenging conditions, we uncover alarming safety concerns that underscore the critical importance of robust evaluation methods in autonomous driving development. This approach not only pushes the boundaries of synthetic data generation but also provides invaluable insights into the real-world performance and safety of autonomous driving systems in extreme situations.

In conclusion, this thesis presents my perspective on how the autonomous driving field can address its fundamental scaling challenges through multiple complementary strategies. For the supervision bottleneck, approaches range from optimizing human annotation through active learning to eliminating the need for supervision entirely through self-supervised learning. The data bottleneck is addressed through both the collection of diverse real-world data and the generation of synthetic data, particularly for rare and safety-critical scenarios. Together with this broad picture, our specific contributions help make autonomous driving development more scalable and robust.

Chapter 2

Research Questions

As the field of autonomous driving progresses toward vehicles capable of operating without human intervention in any condition, it faces critical scaling challenges that threaten to bottleneck development. The core assumption underlying this thesis is that to achieve the required robustness and reliability, we must dramatically scale up the training and testing of autonomous systems.

This scaling challenge can be decomposed into several dimensions: computational resources, algorithms, raw data, and supervision[1]. Computational scaling, though critical, is primarily driven by hardware advances and industry investment. Similarly, algorithmic improvements such as advanced network architectures or alternatives to gradient-based learning remain tantalizing due to their unknown potential. However, empirical studies across various domains consistently demonstrate that current approaches can achieve remarkable performance when scaled far enough [1, 14].

Given this, we focus our efforts on the data and supervision bottlenecks, which present both unique challenges and the most promising path towards advancing autonomous driving capabilities. We structure our investigation around two main research questions, one for each bottleneck:

RQ1: *What strategies can be employed to overcome the supervision bottleneck in autonomous driving development?*

The supervision bottleneck stems from the extensive human annotation required by traditional supervised learning approaches. Modern perception systems require millions of

[1]While data could be considered to encompass both raw data and its annotations, we deliberately separate these dimensions as they present distinct scaling challenges with different solution spaces.

precisely labeled objects across diverse scenarios, with each object potentially requiring multiple annotations (e.g., 2D bounding boxes, 3D boxes, instance segmentation). As dataset sizes grow exponentially to handle more complex scenarios, the cost and time requirements for manual labeling become prohibitive. This challenge demands both immediate solutions to maximize the value of existing supervision and approaches to reduce our dependence on human annotations entirely.

The autonomous driving industry invests substantial resources in creating high-quality human annotations, ranging from basic bounding boxes to complex semantic segmentation masks. While these annotations are primarily used for their immediate purpose of training specific perception models, they contain valuable implicit information about scene dynamics, object relationships, and real-world physics that could be leveraged for additional learning objectives. This untapped potential leads to our first subquestion:

RQ1a: *How can we extract more value from existing human annotations, beyond their directly intended use? (Paper I)*

However, maximizing the value of existing annotations alone cannot solve the scaling challenge. A more sustainable approach requires reducing our dependence on explicit human supervision entirely. Pure self-supervised learning techniques have shown remarkable success in learning representations from raw data, but they often lack semantic grounding and interpretability - crucial requirements for safety-critical systems. Vision-language models offer a promising direction by providing both semantic understanding through natural language associations and human interpretability without requiring explicit labels [10]. However, autonomous driving presents unique challenges for such approaches due to the lack of freely available text pairs, unlike general text-image datasets which can be scraped from the internet. While general image understanding transfers somewhat well to autonomous driving, this problem is particularly pronounced for non-image sensors like lidar. This leads us to ask:

RQ1b: *How can we achieve semantic understanding of multi-modal autonomous driving data without explicit human supervision? (Paper II)*

Having addressed the supervision challenges, we turn our attention to the other bottleneck:

RQ2: *What strategies can be employed to overcome the data bottleneck in autonomous driving development?*

The data bottleneck centers on the essential challenge of collecting sufficiently diverse and comprehensive real-world driving data. Achieving robust performance requires exposure to

varied weather conditions, cultural contexts, driving styles, and infrastructure variations - a breadth of experience difficult to obtain through limited fleet operations. Moreover, safety-critical scenarios present a particular challenge, being either too dangerous to deliberately encounter or too rare to collect at meaningful scale through normal driving - a limitation that affects both training and systematic safety validation.

While simulation offers a potential solution to these challenges, conventional approaches either lack the necessary fidelity for interfacing with modern perception systems, suffer from a lack of diversity, or require too much human artistry to be scalable. To address these challenges systematically, we explore two complementary research directions.

The fundamental challenge in autonomous driving data collection lies in its inherent long-tail nature: while standard driving scenarios can be collected at scale, many critical situations occur extremely rarely in practice. This creates a crucial gap between the data needed for robust deployment and what can be feasibly collected through conventional means. This leads us to ask:

RQ2a: *How can we ensure comprehensive data coverage of real-world driving scenarios? (Papers* III, IV*)*

The complexity of modern autonomous driving systems, with their deep neural networks and intricate software stacks, suggests the need for joint evaluation of the full system – from raw sensor inputs to final control outputs. Even in traditional modular architectures, simulating realistic error modes in perception components remains challenging, and conventional sensor simulation approaches struggle to generate the photorealistic data needed by modern perception systems at scale. The increasing adoption of end-to-end driving models further complicates this as there are no explicit interfaces between perception and planning for simulation injection. Real-world validation approaches like test track evaluation offer controlled conditions but limited scale, while public road testing cannot safely explore important edge cases. This fundamental tension between comprehensive safety validation and real-world feasibility leads us to ask:

RQ2b: *How can we effectively verify the safety of the full AD system, from pixel to torque? (Paper* V*)*

Together, these research questions address two critical bottlenecks in scaling autonomous driving development: the need for scalable supervision strategies and comprehensive, diverse, and even interactive datasets for training and validation. By addressing these challenges through novel approaches and techniques, this thesis aims to advance the robustness and reliability of autonomous systems, paving the way for safe and effective deployment.

Chapter 3

The Supervision Bottleneck

The performance of deep learning systems is fundamentally constrained by what we can teach them. Traditionally this teaching has come through supervised learning from human-annotated data. As established in our research questions, this creates a critical supervision bottleneck that limits the scalability of autonomous driving development.

This chapter explores these challenges and their potential solutions through increasingly scalable approaches: from methods that make human annotation more efficient through active learning and automatic annotation, to techniques that extract additional value from existing labels, and finally to approaches that reduce or eliminate the need for explicit human supervision entirely.

Our work addresses two key research questions in this progression: how to maximize value from existing annotations (RQ1a), which we tackle through a novel method for extracting additional training signals from standard 2D annotations through future prediction (Paper I), and how to move beyond explicit supervision while maintaining interpretable 3D understanding (RQ1b), which we address through LidarCLIP (Paper II), demonstrating how language supervision can provide semantic understanding in lidar point clouds without explicit labels or even text pairs.

Throughout this chapter, we will examine how these various approaches complement each other and how they contribute to the broader goal of developing scalable autonomous driving systems.

3.1 Learning from Human Supervision

To understand the challenges of human supervision in autonomous driving, let's begin with the basic principles of supervised learning before examining how these principles scale in complexity for real-world applications. Consider the classical supervised learning setup: we have a dataset $D = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$, where each x_i represents an input (such as an image) and y_i its corresponding human-assigned label. Our goal is to train a neural network $f(x; \theta)$ with parameters θ that can accurately map inputs to their corresponding labels.

The training process follows three basic steps:

- **Forward pass**: For each input x_i, compute the network's prediction

$$\hat{y}_i = f(x_i; \theta). \tag{3.1}$$

- **Loss calculation**: Compare the prediction to the true label using a task-appropriate loss function (l)

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} l(y_i, \hat{y}_i). \tag{3.2}$$

- **Backwards pass**: Perform back-propagation to calculate the gradient of the loss for each parameter θ_l using the chain rule and adjust the network parameters using gradient descent:

$$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}(\theta). \tag{3.3}$$

This process, while conceptually straightforward for simple tasks like image classification, becomes increasingly complex when applied to autonomous driving. Let us examine how this complexity manifests through a progression of increasingly sophisticated perception tasks:

2D Object Detection: More complex image-space tasks, such as 2D object detection, add an additional layer of complexity. Now the network must make multiple predictions per image, and not only classify objects but also locate them in the image. The input x_i could still be an image, but the label y_i now includes both class information and bounding box coordinates (x, y, w, h) for a variable number of objects. The loss function becomes multi-objective

$$\mathcal{L}(\theta) = \lambda_{\text{cls}} \mathcal{L}_{\text{cls}} + \lambda_{\text{loc}} \mathcal{L}_{\text{loc}}, \tag{3.4}$$

where λ_{cls} and λ_{loc} balance classification and localization objectives.

3D Perception: Moving to 3D perception, the complexity increases dramatically. Labels now specify full 3D bounding boxes with position (X, Y, Z), dimensions (L, W, H), and either ground-aligned yaw(θ) or full orientation (ϕ, θ, ψ). The input space also expands to include multiple sensor modalities (camera, lidar, radar), each requiring careful calibration and synchronization, and some projection is often needed to map the input space to the output space (e.g. image plane to 3D). The loss function grows accordingly:

$$\mathcal{L}(\theta) = \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{loc}}\mathcal{L}_{\text{loc}} + \lambda_{\text{dim}}\mathcal{L}_{\text{dim}} + \lambda_{\text{rot}}\mathcal{L}_{\text{rot}}. \tag{3.5}$$

4D Scene Understanding: Beyond frame-level perception, capturing the temporal consistency of object identities, motion trajectories, and behavioral patterns across sequences becomes essential. This temporal aspect is intrinsically linked with spatial and semantic understanding - annotators must label not only the evolution of object states but also their interactions. For instance, annotating a lane change maneuver requires tracking vehicle position and orientation over time while simultaneously encoding the relationship to surrounding traffic participants and road infrastructure. Such holistic 4D scene understanding exponentially increases the annotation complexity, as each additional temporal frame multiplies the number of spatial and semantic relationships that must be consistently labeled.

This progression illustrates how the seemingly straightforward paradigm of supervised learning becomes increasingly challenging when applied to autonomous driving. Each step up in complexity not only increases the dimensionality of the label space but also introduces new sources of uncertainty and ambiguity that must be carefully handled in the annotation process.

3.1.1 Real-world complexities

Beyond the inherent progression of complexity described above, the practical challenges of annotation in autonomous driving create additional layers of difficulty. A single frame in a complex and crowded urban scene can take an experienced annotator more than an hour to fully annotate [15, 16], with each challenge compounding the others to create a formidable annotation task.

Spatial and Temporal Complexity: The fundamental challenge lies in the high-dimensional nature of autonomous driving data. Annotations must capture precise 3D positioning (x, y, z), orientation (yaw, pitch, roll), and temporal aspects such as motion states and trajectory patterns. This dimensionality explosion is particularly challenging in dynamic environments, where annotators must maintain consistency while tracking multiple objects. For example, annotating a busy intersection requires simultaneously tracking numerous vehicles and pedestrians, each with their own trajectories and interactions.

Sensor Interpretation and Fusion: Autonomous vehicles typically employ multiple sensor modalities, each with its own characteristics and challenges. Lidar point clouds, while providing precise depth information, are often sparse and require significant expertise to interpret correctly. Camera data offers rich visual information but suffers from perspective distortion and lighting variations. Radar data adds another layer of complexity with its unique noise patterns and reflection characteristics. Annotators must cross-reference these different modalities to ensure accurate and consistent labeling, significantly increasing the cognitive load and time required for each annotation.

Occlusion and Uncertainty Handling: Real-world driving scenarios frequently involve partial observations and occlusions. Objects may be hidden behind other vehicles, buildings, or vegetation, requiring annotators to make informed estimates about their full extent and shape. This introduces an element of uncertainty into the annotation process, as different annotators might make different judgments about the same partially observed object. The challenge becomes even more pronounced when dealing with temporal sequences, where maintaining consistent object identities across frames with varying levels of occlusion (including complete occlusion) requires careful attention and expertise.

Semantic Complexity: Beyond basic geometric properties, autonomous driving annotations must capture rich semantic information. This includes fine-grained classifications (distinguishing between various vehicle types or pedestrian poses), behavioral states (turn signals, brake lights, pedestrian gestures), and interaction patterns between different road users. The semantic complexity extends to scene-level understanding, requiring annotation of road topology, traffic rules, and possibly even social behaviors. This multi-level semantic annotation adds a significant cognitive load to the annotation process and requires annotators with deep domain knowledge.

These real-world complexities interact with and amplify the challenges discussed earlier. For instance, the need for temporal consistency in annotations affects not only the basic object tracking but also the interpretation of semantic states and behaviors over time. Similarly, sensor fusion challenges impact everything from basic object detection to complex scene understanding.

Moreover, the relationship between annotation quantity and model performance is often non-linear. Empirical studies have shown that order-of-magnitude increases in labeled data are often necessary to achieve meaningful performance gains [1, 17]. This pattern has been particularly well-documented in large language models, where exponential increases in both compute and training data are required to achieve linear improvements in performance [14]. It's reasonable to assume that autonomous driving faces similar or even steeper scaling challenges, especially when evaluating performance on rare edge cases that are critical for safety. This non-linear scaling poses a significant challenge for autonomous driving development: as we push for ever-higher levels of accuracy and reliability, the annotation requirements

grow exponentially.

The quality-quantity trade-off is also an interesting aspect of the problem. As mentioned, even experienced annotators, taking their time to make the best possible annotations, may interpret ambiguous situations differently, leading to noise in the training data. Sometimes such noise is inherent to the problem (such as tracking a fully occluded object), and sometimes due to subjective interpretations of annotations guidelines. The pool of expert annotators is also limited, creating a bottleneck in the data preparation pipeline. By relaxing the quality requirements, we could produce vastly larger datasets, but with a significantly higher level of noise. Some works [18, 19] have shown that this could be a worthwhile tradeoff. For the same amount of human effort (and thus time and money), we can maximize value by annotating a mix of high-quantity low-quality and low-quantity high-quality data. However, more studies are needed on how this approach scales to very large-scale datasets where model capacity starts to become a limiting factor. There is a risk that this low-quality data could eventually serve as a lower bound on model performance, meaning that the effort spent on low-quality data only gave short-term benefits and in the long term it was completely wasted.

This fundamental tension - between the need for extremely precise and detailed annotations for training robust autonomous systems and the practical limitations of human annotation capabilities - lies at the heart of the supervision bottleneck. In the following sections, we will explore various strategies to address this challenge, from maximizing the value of existing labels to reducing the reliance on extensive human annotation entirely.

3.2 Efficient Human Supervision

Given the challenges outlined in the previous sections, increasing human efficiency becomes critical for scalability in autonomous driving development. Significant advances have been made in maximizing the impact of human effort through intelligent tooling and semi-automated approaches. These range from active learning strategies that prioritize the most informative samples for human review, to fully automatic pipelines that require human verification only in rare cases. This section examines these approaches in detail. Additionally, we introduce our approach to Research Question 1 (RQ1a), which complements these methods by maximizing the value extracted from existing annotations.

3.2.1 Offline Auto-Labeling

One key advantage in autonomous driving development is the ability to perform offline processing. Unlike real-time inference, where computational resources and latency require-

ments impose strict constraints, offline processing allows us to leverage powerful tools and techniques to improve annotation quality and consistency. This enables the use of computationally expensive models, such as large-scale transformers [20], alternative task formulations, such as end-to-end detectors [21], and even large language models with vision capabilities, like GPT-4 [22]. Besides just being bigger (more parameters), these models may use higher quality inputs (e.g., smaller voxel sizes, full resolution images), have longer temporal context (e.g., processing 3s sequences instead of 2-3 frames), use test-time augmentation [23], chain-of-thought reasoning [24], or employ other computationally intensive techniques that would be prohibitive in real-time settings.

Another powerful way to make use of "unlimited" compute is to employ ensembles, combining predictions from multiple diverse models to achieve more robust and accurate annotations [25]. The benefits are twofold: improved accuracy through consensus, and reliable uncertainty estimates. These uncertainty estimates can be used to weigh training samples based on annotation confidence and to efficiently route challenging cases to human annotators for verification and refinement.

Perhaps the most powerful aspect of offline processing is the ability to utilize temporal context, including future information. This "benefit of hindsight" enables sophisticated annotation strategies that would be impossible in real-time settings. For instance, we can use an object's future positions to refine its past locations, reducing jitter and improving consistency. Future frames can reveal objects that were previously occluded, allowing for retroactive annotation of their presence. For example, a pedestrian temporarily hidden behind a parked vehicle can be tracked consistently through the occlusion by using their visible positions before and after. We can also leverage moments of closest approach - when objects are most clearly visible - to improve annotations in frames where these objects were distant or partially visible. This temporal context is particularly valuable for understanding agent intentions and behaviors, as the full trajectory of an interaction becomes available for analysis.

Several works have studied how to build auto-annotation pipelines for 3D object detection in autonomous driving [4, 5]. These approaches typically start off by generating frame-by-frame detections using an ensemble of the best available 3D object detectors. An object-level tracking algorithm [26] is then used to associate detections across frames. Finally, a learned refinement model takes the associated detections, including their sensor measurements (e.g. all lidar points that fall within the bounding box), and refines the annotations to produce a final set of bounding boxes over time. Similar approaches can be used for other tasks, with a prominent special case of offline mapping, which has been a huge research topic [27–29] both due to the importance of the task and the possibility of making strong assumption about the (mostly) static nature of such maps.

These offline advantages, when properly leveraged, can significantly reduce the burden on

human annotators and potentially improve annotation quality. However, these systems are not without failure modes, which often arise in the most complex or ambiguous cases. In such instances, human expertise remains essential, not only to resolve these challenges effectively but also to provide critical feedback that improves the system over time. The key to building an efficient and scalable auto-annotation pipeline lies in minimizing failure modes through robust design, quickly identifying those that occur, and enabling their resolution with minimal human effort.

3.2.2 Superior Sensors

Another strategy for efficient supervision involves leveraging high-quality, often expensive sensors during the data collection and annotation phase. This approach recognizes that while deployment vehicles must optimize for cost and practicality, development fleets can utilize more sophisticated sensor configurations to generate high-quality training data.

For instance, high-resolution lidar sensors that generate dense point clouds can provide detailed geometric information that makes object boundaries and classifications more apparent. Multi-camera setups with overlapping fields of view can help resolve ambiguities and occlusions through multi-view geometry. High-dynamic-range cameras can capture clear images in challenging lighting conditions, while precision GPS/IMU systems can provide accurate ground truth for motion and localization tasks.

This superior sensor data serves multiple purposes. First, it simplifies the annotation process by providing clearer, more detailed information to annotators. Second, it can serve as ground truth for training models to perform well with more cost-effective sensors - a form of teacher-student learning where models learn to replicate the performance of superior sensors using more practical hardware configurations. Third, the high-quality data can be used to validate and refine annotations generated by other means, such as automated labeling systems.

A very concrete example of this in autonomous driving is the use of lidar 3D object detectors as teachers for camera-only 3D object detectors [30–32]. This asymmetric setting is particularly powerful, as the lidar provides direct measurements of 3D geometry, while the camera-only detector must infer this information from 2D images, thus leading to meaningful learning even under noisy and partial labels from the teacher. Furthermore, the lidar may be too expensive for deployment on customer vehicles, but still feasible to deploy on data collection vehicles.

3.2.3 Using the Human Where It Counts

As discussed in the previous section, automated annotation systems can provide initial labels for large datasets while identifying uncertain cases that require human verification. This uncertainty-based routing of human attention naturally connects to the broader field of active learning, which studies how to select the most informative samples for annotation to maximize model performance [33]. In autonomous driving, this selection process operates at multiple scales: choosing which driving sequences to annotate, which frames within those sequences, and even which parts of individual frames require human attention [34].

At the dataset level, selection strategies typically consider multiple factors. Uncertainty-based sampling identifies scenes where the model's predictions have low confidence [35], indicating potential edge cases or difficult scenarios that require human expertise. This uncertainty can be effectively estimated using model ensembles, as discussed earlier, providing a natural bridge between automated and human-in-the-loop annotation processes. Diversity sampling ensures broad coverage across different environmental conditions, object types, and interaction patterns. Expected model change estimation [36] identifies samples likely to cause the largest updates to the model's parameters, maximizing the learning impact of each annotation.

Within individual samples (images or point clouds), interactive annotation techniques [2] can further optimize human effort. For instance, in 3D scene labeling, a model might automatically segment most of the scene but request human verification for ambiguous regions or object boundaries. These approaches often combine automatic proposals with efficient user interfaces that allow rapid verification and refinement [37], effectively leveraging human expertise while minimizing tedious manual work.

While these approaches significantly improve the efficiency of human annotation, they still, by definition, rely on a human in the loop, which will always be somewhat of a bottleneck. Furthermore, even with the approaches described here, it's not easy to accurately allocate the human effort to the most valuable samples. Sure, we can annotate the most difficult, or most uncertain cases, but it's not clear whether those are the most valuable samples for the ultimate task of safe driving.

3.2.4 Making the Most of Your Labels (Paper I)

Given the significant investment required for human annotation in autonomous driving, maximizing the utility of each label becomes crucial. A vast body of research focuses on improving model performance through better architectures, loss functions, and data augmentation strategies. While these advances are important, they primarily optimize how we use annotations in their intended way - as direct supervision signals for specific tasks. Less

attention has been paid to extracting additional value from existing annotations in novel ways.

One established approach is multi-task learning, where a single model is trained on multiple related tasks using shared representations[38]. However, the effectiveness of this approach is highly situational and depends on finding naturally complementary objectives with well-balanced loss functions [39, 40]. Instead, we propose an orthogonal approach: training networks to predict human annotations in future frames using only unlabeled past data.

This approach is conceptually similar to how language models learn by predicting the next word in a sentence. Just as predicting the next word requires understanding grammar, context, and semantic relationships, predicting future annotations requires understanding scene dynamics, object relationships, and physical constraints. Importantly, predicting future annotations offers a more stable learning objective than directly predicting future sensor data (such as raw images or point clouds). While future sensor data is extremely high-dimensional and has high inherent uncertainty, future annotations capture the essential semantic and geometric information we care about while abstracting away these irrelevant variations.

Our work on "future object detection" (Paper I) demonstrates the power of this approach. Using standard 2D bounding box annotations, we train a network to predict future bounding boxes from a short sequence of past frames. Despite the apparent simplicity of this task - predicting 2D boxes from 2D data - the network develops sophisticated capabilities:

- The model learns an implicit understanding of 3D scene geometry and object motion, despite being trained only on 2D annotations.
- In uncertain situations, the network learns to predict multiple possible future trajectories, effectively reasoning about the inherent uncertainty in dynamic scenes.
- Most surprisingly, the model exhibits emergent tracking capabilities, learning to associate objects across frames without explicit tracking supervision, as shown in fig. 3.1.

This work exemplifies how seemingly simple prediction tasks can guide models to learn rich, generalizable representations of the world. By requiring the network to reason about future states, it develops an understanding of scene dynamics, physical constraints, and object interactions. The emergence of tracking capabilities is particularly noteworthy - it suggests that temporal prediction tasks can naturally lead to the development of fundamental perception capabilities, without explicit supervision.

The success of this approach suggests a promising direction for maximizing the value of human annotations through prediction-based pre-training. Rather than just using an-

(a) input w/ attention **(b)** input w/ attention **(c)** future w/ predicted box

Figure 3.1: Emergent tracking capabilities visualized through attention maps, showing how the model learns to associate objects across frames without explicit tracking supervision. Illustration from Paper I.

notations as direct supervision signals, we can use them as targets for prediction tasks that encourage a deeper understanding of scene dynamics and temporal relationships. While our work demonstrates the potential of this approach, several important questions remain for future research. We focused primarily on qualitative analysis of emergent capabilities, leaving systematic evaluation on downstream tasks for future work. Additionally, direct comparison with sensor prediction approaches, which have shown recent progress despite their inherent challenges [41], would be valuable to explore. Future work should also investigate how this approach scales with larger datasets and more complex annotation types. Nevertheless, our results point toward more efficient ways of incorporating human supervision into autonomous driving systems, potentially offering a complementary approach to existing self-supervised techniques.

3.3 Learning Without Labels

While the strategies discussed in the previous section can significantly improve the efficiency of human supervision, they still fundamentally rely on explicit labeling. However, the persistent challenge of the annotation bottleneck in autonomous driving has motivated researchers to explore more radical approaches that can learn directly from raw sensor data with minimal or no human supervision [6, 7, 9, 10, 42]. These approaches not only offer the potential to dramatically reduce annotation costs but also to leverage the vast amounts of unlabeled data collected during vehicle operations.

In this section, we explore three key paradigms for learning without explicit labels: self-supervised learning, which creates supervision signals from the data itself; weak supervision, which leverages readily available but noisy sources of information; and language supervision, which uses natural language as a flexible form of supervision. Each approach offers unique advantages and faces distinct challenges in the context of autonomous driving.

3.3.1 Self-Supervised Learning

Self-supervised learning has emerged as a powerful paradigm for learning rich representations from unlabeled data, making it particularly relevant for autonomous driving where vast amounts of raw sensor data are collected during vehicle operations. The core idea is to create pretext tasks that the model can learn from without requiring explicit human annotations, often by exploiting the natural structure and relationships within the data itself. These pretext tasks are carefully designed to encourage the model to learn meaningful features that can transfer to downstream tasks of interest.

General Approaches in Computer Vision: Recent years have seen remarkable progress in self-supervised visual representation learning, with several key approaches demonstrating how carefully designed learning objectives can lead to powerful visual representations. Contrastive learning methods like SimCLR [6] learn by maximizing cosine similarity (sim) between differently augmented views of the same (z_i, z_j), while minimizing similarity to all other images (z_k):

$$\mathcal{L}_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} 1_{[k \neq i]} \exp(\text{sim}(z_i, z_k)/\tau)} \tag{3.6}$$

The intuition behind this approach is powerful yet simple: features that remain consistent across different views of the same image (e.g., different crops, color changes, or rotations) are likely to capture meaningful semantic information. The temperature parameter τ is essentially a hyperparameter that controls how the model treats difficult negative examples, helping it learn fine-grained distinctions between similar but different objects.

DINO (DIstillation with NO Labels) [7] uses a similar pretext task of matching features across different views of the same image, but takes a different approach. Instead of using negative examples, a teacher-student architecture is used, where the teacher f_t provides stable learning targets for the student f_s, and the loss is essentially cross-entropy, with centering (C) for the teacher:

$$P_s(x) = \text{softmax}(f_s(x)/\tau_s)) \tag{3.7}$$

$$P_t(x) = \text{softmax}((f_t(x) - C)/\tau_t)) \tag{3.8}$$

$$\mathcal{L}_{\text{DINO}} = H(P_t(x), P_s(x')) = -P_t(x) \log P_s(x') \tag{3.9}$$

The key insight of DINO, and similar methods [42], is that by using a momentum-updated teacher network (essentially an ensemble of past student networks), the model can learn

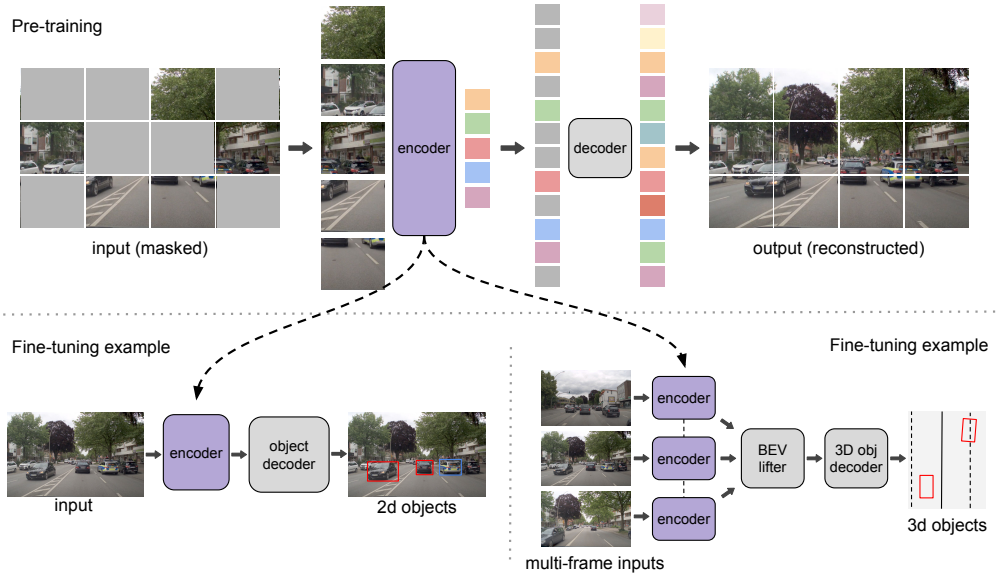


Figure 3.2: Overview of how self-supervised methods can be used in practice. The self-supervised pre-training (top) is exemplified using the Masked AutoEncoder [8] due to its simplicity. The input is split into patches and tokenized, and most of these tokens are dropped. The encoder (what we want to use later) extracts meaningful features from the remaining patches, and a decoder attempts to reconstruct the whole image. During task-specific fine-tuning (bottom) the original decoder is discarded and replaced with a more suitable task-decoder. For 2D object detection (bottom left) this might be a DETR decoder [21], but the pre-trained image encoder can also be part of more complex AD-specific architectures (bottom right) which transform image features into a bird's eye view (BEV) for more accurate 3D processing [43].

consistent and meaningful representations without negative examples and without collapsing to trivial solutions. This approach has proven particularly effective at learning representations that capture object-centric features and semantic relationships.

MAE (Masked AutoEncoder) [8] demonstrates that even simpler reconstruction objectives can be highly effective when combined with appropriate augmentations and architectural choices, see fig. 3.2. By masking a large portion of the input (typically 75%) and reconstructing it (using simple mean squared error), MAE forces the model to develop a deep understanding of image structure and content. The high masking ratio is crucial - it prevents the model from taking shortcuts and forces it to learn semantic and structural patterns rather than low-level pixel relationships.

Building upon these foundational approaches, DINOv2 [9] has recently emerged as a powerful synthesis of various self-supervised techniques. By combining distillation, teacher-student, image level objectives (like DINO), patch level objectives (like MAE), many other carefully tuned design choices, and training on a very large curated dataset[1], DINOv2

[1]The authors show that DINOv2 outperfoms prior self-supervised methods when trained on public data as well, but the magic sauce contained in the scale and curation of the private training set is not be understimated.

achieves state-of-the-art performance across a wide spectrum of vision tasks. Its success demonstrates how the complementary strengths of different self-supervised approaches can be unified into a single framework, producing rich visual representations that rival or even exceed those learned through traditional supervised training.

The remarkable performance of these methods, particularly DINOv2, suggests exciting possibilities for autonomous driving. While most self-supervised approaches have focused on 2D image understanding (besides language of course), their principles could be extended to 3D perception. Just as these methods learn powerful visual features from 2D image structure and relationships, similar approaches could potentially learn rich 3D representations from lidar point clouds and multi-view videos. Alternatively, there are promising methods that learn to lift the knowledge of pre-trained 2D models into the 3D domain [44].

Tailored Approaches for Autonomous Driving: The autonomous driving domain offers unique opportunities for self-supervised learning by leveraging the rich geometric and temporal structure inherent in driving data. For instance, Monodepth2 [45] demonstrates how geometric consistency between multiple views can enable depth estimation without depth labels:

$$\mathcal{L} = \mathcal{L}_{photo} + \lambda \mathcal{L}_{smooth} \tag{3.10}$$

This approach exploits the fundamental relationship between camera motion and scene structure, effectively using the camera's movement as a form of supervision. Given a sequence of images I_t, the photometric loss $\mathcal{L}_{photo}$ measures how well the predicted depth can explain adjacent frames through reprojection:

$$\mathcal{L}_{photo} = \min_{t' \in \{t-1, t+1\}} \rho(I_t, I_{t' \to t}) \tag{3.11}$$

where $I_{t' \to t}$ is frame t' warped to frame t using the predicted depth (d_t) and known camera motion, and ρ combines SSIM and L1 distance. The minimum over adjacent frames helps handle occlusions. The smoothness loss $\mathcal{L}_{smooth}$ encourages locally smooth depth predictions while allowing discontinuities at image edges:

$$\mathcal{L}_{smooth} = |\partial_x d_t| e^{-|\partial_x I_t|} + |\partial_y d_t| e^{-|\partial_y I_t|} \tag{3.12}$$

The intuition is powerful: if we correctly predict the depth of a scene, we should be able to use that depth to explain how the scene appears from slightly different viewpoints. The photometric loss captures this by essentially asking "if I move the camera slightly, does my depth prediction explain what I see?", while the smoothness loss incorporates the prior

knowledge that depth typically varies smoothly except at object boundaries (where the pixels typically also change rapidly, causing a large $\partial_x I_t$ term).

Recent works Agro *et al.* [13] and Khurana *et al.* [46] extend these geometric principles to learning a more complete 4D scene understanding. They show how self-supervised optimization of continuous occupancy fields can capture rich spatio-temporal scene representations without explicit supervision. Both methods use future lidar point clouds to learn a 4D occupancy field, with the key insight that directly predicting future point clouds is suboptimal. Instead, they predict dense future occupancy and supervise it using the future point cloud through differentiable rendering or clever sampling. At a high level this follows a similar principle to our work in Paper 1, where we avoid predicting raw sensor signals in favor of more structured representations (occupancy in their case, object boxes in ours) that abstract away irrelevant variations in sensor space while preserving the essential information that facilitates learning a robust world representation.

Beyond these AD-specific approaches, general self-supervised methods have also been successfully adapted to the autonomous driving domain. For instance, the MAE approach described earlier has been modified to handle lidar data by reconstructing masked voxelized point clouds [47]. This adaptation exemplifies how advances in general self-supervised learning can be effectively transferred to autonomous driving, providing powerful pre-training objectives for 3D perception tasks.

The Role of Self-Supervised Learning: While self-supervised learning has shown remarkable success in learning rich representations of the world, there's an important distinction between implicit and explicit semantics. Self-supervised models can learn deep semantic understanding of scenes - recognizing objects, understanding spatial relationships, and capturing temporal dynamics. However, this understanding remains implicit in the model's representations, not explicitly aligned with human-defined categories and concepts. For instance, a model might learn to distinguish between different types of vehicles based on their shape and motion patterns, but without supervised fine-tuning, it cannot explicitly label them as "cars," "trucks," or "buses."

This makes self-supervised learning an ideal pre-training strategy rather than a complete replacement for human supervision. Models initialized with self-supervised weights demonstrate significantly improved performance, faster convergence, and better generalization when fine-tuned on limited labeled data[6, 8, 48]. The self-supervised pre-training provides a strong foundation of implicit understanding, while the supervised fine-tuning aligns this understanding with human-defined concepts, effectively addressing the annotation bottleneck by minimizing the required amount of human annotation.

3.3.2 Weak Supervision

While self-supervised learning creates supervision signals from the data itself, weak supervision leverages readily available but potentially noisy or imprecise sources of information. In autonomous driving, the domain naturally provides several such supervision sources. Driver interventions - moments when a human driver takes control from an autonomous system - can serve as weak labels for potentially challenging or unsafe situations. Vehicle telemetry data, including brake pressure, steering angle, and acceleration patterns, provides implicit supervision about appropriate driving behavior in different contexts. High-definition maps offer structured information about road layout and navigation constraints, while established traffic rules encode basic behavioral expectations.

The power of weak supervision lies in its scalability and natural alignment with the task at hand. Unlike human annotations, which require explicit effort to create, these signals are automatically generated during normal vehicle operations. For instance, fleet vehicles continuously generate telemetry data that captures how human drivers respond to various traffic situations. This data, while noisy and not explicitly annotated, contains valuable information about appropriate driving behavior.

Comments on End-to-End Driving: Perhaps the most ambitious application of weak supervision in autonomous driving is the recent push toward end-to-end architectures that learn to directly predict driving actions from sensor inputs. These approaches typically use the vehicle's future trajectory as a form of supervision, essentially learning to imitate human driving behavior. By bypassing intermediate perception and prediction tasks, end-to-end systems potentially offer a more direct path to autonomous driving while avoiding many of the annotation challenges discussed earlier.

The appeal of end-to-end approaches is clear: they eliminate the need for expensive intermediate annotations and potentially learn more optimal representations for the ultimate task of driving. However, their success heavily depends on the quality and quantity of available driving data. While basic driving behavior can be learned from demonstrations, handling rare and safety-critical scenarios remains a significant challenge.

This suggests that while end-to-end architectures may indeed represent the future of autonomous driving, their training likely needs to be supplemented with the various supervision strategies discussed in this chapter. Self-supervised pretraining can help learn robust representations from unlabeled data, explicit supervision can guide the learning of critical semantic concepts, and various forms of weak supervision can provide additional learning signals. The key challenge lies in effectively combining these different forms of supervision to create systems that are both scalable and reliable.

3.3.3 Language Supervision (Paper II)

Language supervision has emerged as a powerful paradigm for learning visual representations, particularly following the success of CLIP (Contrastive Language-Image Pre-training) [10].[2] These models learn by aligning visual and textual representations in a shared embedding space, leveraging the natural correspondence between images and their descriptions. The key insight is that language can serve as a flexible, scalable form of supervision that captures rich semantic information without requiring explicit task-specific labels.

Vision-Language Models: CLIP demonstrates how contrastive learning between image and text embeddings can create powerful visual representations. The objective is similar to that of SimCLR (discussed earlier), but with two terms to handle the two modalities symmetrically:

$$\mathcal{L}_{\text{CLIP}} = -\log \frac{\exp(sim(i_i, t_i)/\tau)}{\sum_j \exp(sim(i_i, t_j)/\tau)} - \log \frac{\exp(sim(i_i, t_i)/\tau)}{\sum_j \exp(sim(i_j, t_i)/\tau)}, \tag{3.13}$$

where i and t are embeddings of an image and its corresponding text description, instead of two augmentations of the same image. By training on hundreds of millions of image-text pairs from the web, CLIP learns to align visual and linguistic concepts in a way that generalizes remarkably well to new tasks and domains.

Adaptation to Autonomous Driving: In Paper II, we present LidarCLIP, which extends vision-language learning to 3D point cloud data. While autonomous driving lacks the vast image-text datasets available for general computer vision, we show how to effectively transfer knowledge from pretrained vision-language models to the lidar domain. This enables zero-shot transfer of semantic understanding to 3D perception tasks, reducing the need for explicit 3D annotations. See fig. 3.3 for an overview of our method.

The power of language supervision in autonomous driving extends beyond basic perception tasks. Language provides a natural interface for specifying complex behavioral objectives, describing unusual scenarios, and communicating safety constraints. For instance, a model could learn to recognize "a pedestrian about to cross the street" or "a truck blocking multiple lanes" from textual descriptions, without requiring explicit annotations of these scenarios.

The retrieval capabilities enabled by such language-vision models are particularly valuable for autonomous driving development. In cloud-based development, these models enable efficient mining of specific scenarios from vast amounts of unlabeled data, such as finding instances of "vehicles merging in heavy rain" or "pedestrians crossing between parked

[2] In most cases language supervision is simply a particular form of weak supervision, and some formulations could maybe even be considered self-supervised, but it is noteworthy enough to merit a separate section.

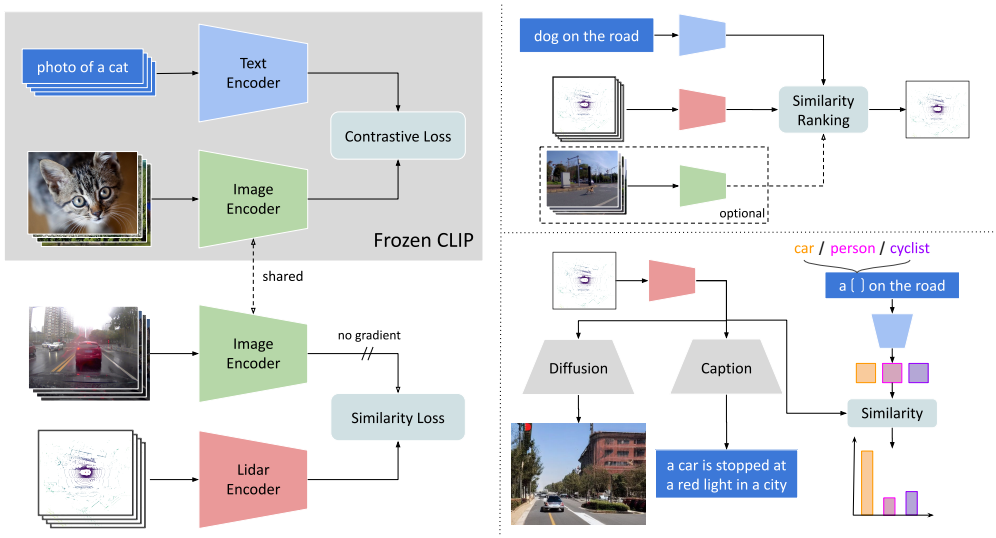


Figure 3.3: Schematic of LidarCLIP. We assume access to aligned texted an image encoders (CLIP), pre-trained on massive online datasets. We then train a lidar encoder to match the embeddings of the frozen image encoder on a large automotive dataset. By proxy this also aligns the lidar embeddings with the text embeddings, since all map into the same space. This enables a range of applications: primarily retrieval (top right), but also the more exotic lidar-to-image generation, point cloud captioning, and zero-shot object classification (bottom right). Illustration from Paper II.

cars." Even in deployed vehicles, this retrieval capability could help identify and log relevant edge cases for further analysis. Beyond single-modality retrieval, our approach enables cross-modal queries between images and point clouds, leveraging the shared embedding space. This is particularly powerful when combining embeddings from multiple modalities of the same scene, as each modality captures complementary aspects of the environment. For instance, lidar provides precise geometric information while images capture rich visual details, allowing for more nuanced and accurate retrieval queries.

As demonstrated in fig. 3.4, the multi-modal nature of our approach enables sophisticated retrieval strategies that would be impossible with single-modality models. By directing different aspects of a query to different modalities - for example, using image embeddings to identify poor visibility conditions while using lidar embeddings to verify the presence of specific objects - we can precisely identify challenging scenarios that combine multiple factors.

Our approach to aligning lidar to a pre-existing text-image space has some issues however. For one, the lidar embedding is trained to match an image embedding of the same scene, which is not necessarily a reasonable objective since the sensors can pick up different aspects of the scene. For example, there is a big difference between a round prohibitory sign and round mandatory sign, which will have a big effect on the image embedding but is impossible to distinguish using lidar alone. Another issue is the global alignment of the a

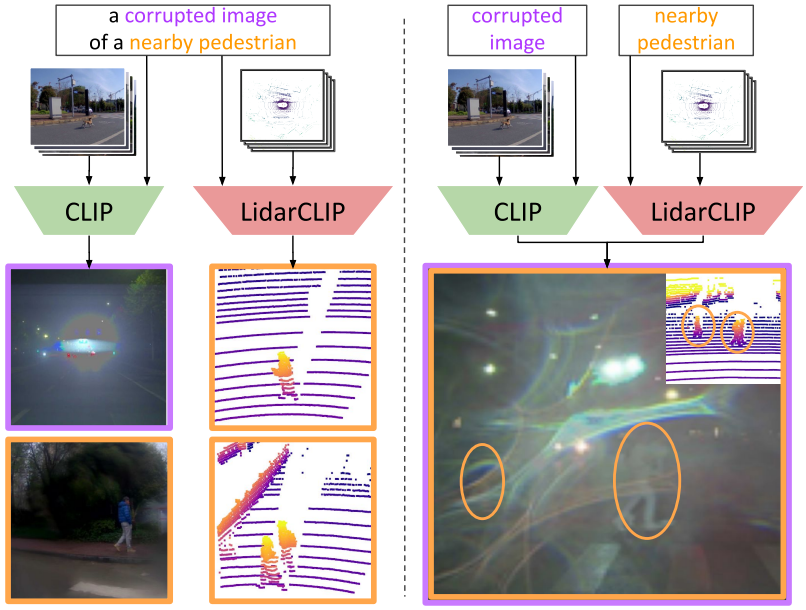


Figure 3.4: Example of multi-modal retrieval using LidarCLIP. In the typical case (left) we query both modalities with the full query, but this fails as shown in the retrieved examples. However, by using different queries for each modality (right), we can use image embeddings to find corrupted images, and lidar embeddings to find nearby pedestrians, and when the retrieval scores are matched we can find extremely valuable edge-cases as the one shown here. Illustration from Paper II.

single embedding of the entire scene. This is understandable when aligning text and image, since it is not clear which parts correspond to each other, but for image and lidar we have accurate calibration and could potentially learn a local feature alignment instead, which is interesting an interesting direction that has been explored in some subsequent works [49, 50].

3.4 Putting it All Together

The various approaches discussed in this chapter each address different aspects of the supervision bottleneck, but their true potential emerges when combined into an integrated pipeline. The recent success of the Segment Anything Model (SAM) [3] demonstrates how such integration can create powerful, scalable systems through self-supervised pretraining, targeted fine-tuning with high-quality annotations, iterative refinement through pseudo-labeling, resulting in a system with strong zero-shot capabilities in novel scenarios.

Drawing from these insights, we can envision a comprehensive pipeline for autonomous driving development, also visualized in fig. 3.5:

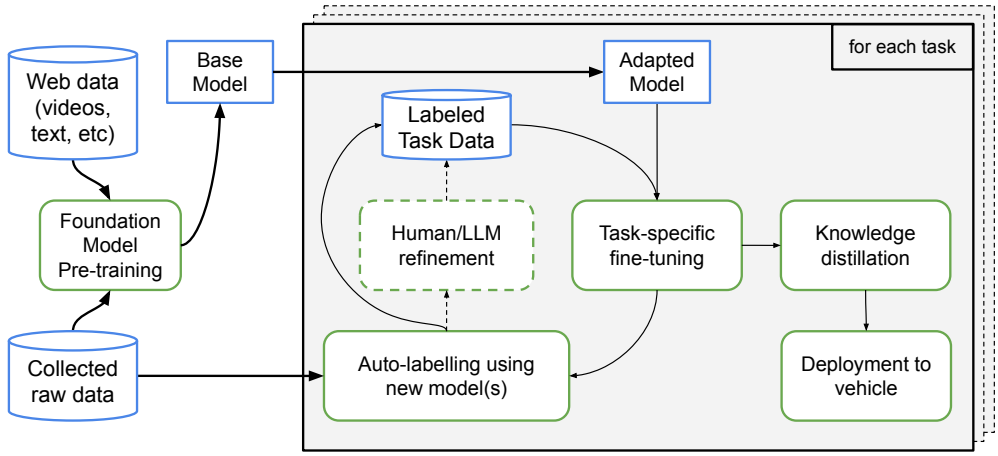


Figure 3.5: A high-level visualization of how the different components discussed in this chapter come together. First, powerful foundation model is pre-trained on all available data. Then a task-specific fine-tuning loop is begun with recursive self-improvement. Finally, once satisfactory performance is achieved, the large task-specific model is distilled into an efficient model that can be deployed to the target vehicle.

Foundation: The pipeline begins with training a large foundation model that combines multiple complementary forms of scalable supervision discussed above. By integrating self-supervised learning, weak supervision signals, and language supervision, we create a model with a rich understanding of the driving domain across multiple dimensions. The resulting foundation model develops strong temporal and geometric understanding of driving scenes through self-supervision, grounds this understanding in human concepts through language supervision, and learns practical driving behaviors through weak supervision signals. This multi-faceted approach creates a robust foundation that captures both the physics and semantics of autonomous driving.

Task Adaptation: The foundation model can then be adapted to specific autonomous driving tasks through various approaches. One path involves fine-tuning on carefully curated human annotations, benefiting from superior sensors and offline processing capabilities. Alternatively, the model could be trained end-to-end using trajectory data and behavioral cloning, potentially supplemented with auxiliary tasks that leverage the pretrained representations. The choice depends on the specific requirements and constraints of the target application.

Scaled Auto-Labeling: An ensemble of such adapted (or fine-tuned) models can be deployed to generate high-quality annotations for a much larger dataset. Different models in the ensemble might use different sensor combinations or training approaches, providing diverse perspectives on each scene. Areas where the ensemble disagrees highlight potentially challenging cases that might benefit from human verification. This verification process, guided by active learning principles, can be scaled up or down based on safety require-

ments and resource constraints.

Iterative Refinement: The larger auto-labeled dataset can be used to train new adapted models, creating a flywheel effect where each iteration improves both the quality of the dataset and the models' performance.

Knowledge Distillation: Finally, the knowledge from the adapted models is distilled into a compact, efficient model suitable for deployment in vehicles, preserving the rich understanding while meeting real-time computational constraints.

Importantly, this pipeline remains flexible - the role of human supervision can be adjusted based on empirical performance and safety requirements, potentially decreasing over time as the self-supervised and language-guided components become more sophisticated. Furthermore, the task adaptation stage is highly flexible as well. We have primarily discussed this in the context of traditional perception tasks and direct trajectory prediction. However, one of the tasks could just as well be a pure language commentary, explaining the driving decisions, as has become popular recently [51, 52].

In conclusion, we have outlined an approach for addressing the supervision bottleneck in autonomous driving. While the individual components of this pipeline may appear straightforward, their successful integration requires careful attention to implementation details. In our field a well-executed simple approach often outperforms the more sophisticated alternatives. By thoughtfully combining multiple supervision strategies and maintaining flexibility in their relative roles, we can create scalable development pipelines that adapt to advancing capabilities in self-supervised and language-guided learning. This approach paves the way for more efficient and robust autonomous driving systems, while remaining open to evolving best practices in model development and validation.

Chapter 4

The Data Bottleneck

No matter how clever we are, at the end of the day the quality of our self-driving systems will be bounded by the quality and quantity of the data that has been used for training and evaluation. This has been demonstrated over and over in the field of deep learning – data is king. From the initial breakthrough of deep networks for image classification, enabled by ImageNet [23], to the recent advances in generative text systems such as GPT 4 [22], access to large amounts of high-quality data has been the common denominator (given an architecture and loss function that can scale enough to make use of that data).

The development of the family of LLama models is particularly noteworthy of study, as the entire training pipeline is described in great detail in the white papers [53–55]. Particularly, one can note that the architecture and training objectives remain almost identical between LLama 2 and Llama 3. Instead, the massive increase in capabilities is attributed to longer training, and especially much higher quality and quantity of data, including a healthy amount of synthetic data. This pattern – where data quality and quantity drive progress more than architectural innovations – has profound implications for autonomous driving development.

In this chapter, we examine how these principles apply to autonomous driving through three complementary perspectives. First, we analyze what makes a great autonomous driving dataset and discuss our contribution in the form of the Zenseact Open Dataset (Paper III). Next, we explore how neural rendering techniques can enable scalable generation of synthetic data through learned reconstruction and modification of real scenarios, and present our method that is tailored to handle the additional complexities of autonomous driving scenarios (Paper IV). Finally, we demonstrate how neural rendering can address the challenge of safety validation by creating a closed-loop simulator with our neural renderer at its core (Paper V). This enables reconfiguring ordinary driving logs into safety-critical scen-

arios that would typically require dedicated test tracks, allowing evaluation of autonomous driving systems on challenging cases that cannot be safely tested in the real world.

Through these three perspectives, we address both the challenge of ensuring comprehensive data coverage (RQ2a) and the need for effective safety validation of the full autonomous driving system (RQ2b).

4.1 Real-World Data Collection

The development of robust autonomous driving (AD) systems requires vast amounts of diverse, high-quality data that captures the full complexity of real-world driving. This data must span different geographical locations, weather conditions, traffic patterns, and cultural contexts. Moreover, it must include both common scenarios that form the backbone of everyday driving and rare events that are critical for safety validation. In this section, we first examine the unique characteristics and challenges of AD data collection, particularly the trade-offs between traditional data collection campaigns and fleet-based approaches. We then analyze existing open datasets and introduce the Zenseact Open Dataset (Paper III), which addresses critical gaps in geographical and environmental diversity.

4.1.1 Characteristics and Challenges of AD Data Collection

Autonomous driving datasets differ fundamentally from traditional computer vision datasets. While most computer vision tasks focus on a central object, AD scenes are complex and multi-faceted, with multiple objects of interest coexisting at varying scales. Moreover, while many driving scenarios may appear similar at first glance, tiny differences can be extremely important - a slight change in a pedestrian's body language might indicate their intent to cross the street, or a partially obscured road sign might contain crucial information.

The multi-modal nature of AD data further sets them apart. They often incorporate multiple sensor types, including different cameras, lidars, radars, and other sensors (ultrasonics, infrared cameras, event cameras, etc). This multi-sensor approach provides a richer representation of the environment but also introduces challenges in sensor fusion and synchronization. Moreover, AD datasets typically include multiple annotation types, such as 2D and 3D bounding boxes, segmentation masks, and object trajectories, adding additional complexity.

But most importantly, AD data inherently reflects the messiness and ambiguity of real-world scenarios, see fig. 4.1. Sensor quality can vary, and calibration issues are not uncommon. Annotation ambiguities arise from real-world complexities such as reflections,

Figure 4.1: Examples of the kind of diverse samples we would like to see in our AD dataset. Varying weather and light conditions from snowy to night to sunny day. Unusual objects, such as the children in snow-sleds and bikes on the back of a car. And somewhat atypical driving environments like tunnels, snow covered roads, and cobblestone roads. All images are taken from ZOD.

billboards, and partially occluded objects, as discussed in section 3.1.1. Environmental factors—including weather conditions (e.g., heavy rain), time of day (e.g., low sun shining directly into the camera), and seasonal changes (e.g., snow on the road obstructing lane markers)—add further variability. These factors collectively contribute to making AD datasets particularly challenging and distinct from more controlled computer vision datasets. Importantly, cleaning up these imperfections would be counterproductive, as these difficulties will inevitably be encountered during real-world vehicle operations.

Perhaps the final nail in the proverbial coffin is the need to capture the ”long tail” of driving scenarios. While most driving time consists of relatively straightforward situations, the rare

and unusual cases are the most valuable for both training and safety evaluation. Beyond the aforementioned conditions (e.g., difficult weather), this long tail consists of rare objects (e.g., furniture fallen from vehicles), unexpected road actors (e.g., kangaroos crossing the road), unusual road configurations (e.g., diagonal pedestrian crossings), and atypical behaviors and interactions (e.g., drunk driving). This latter is further complicated by strong behavioral differences across regions and cultures, which manifest in diverse pedestrian behaviors, traffic patterns, and even vehicle types (e.g. tuk-tuks and Jeepneys in Southeast Asia, pick-up trucks in the US, and EPA-tractors in Sweden that look identical to other cars but can only drive 30 km/h) [56].

Traditionally, autonomous driving data is collected using specially equipped vehicles with high-end sensors and precise calibration. While this approach ensures high-quality data, it has inherent limitations. These vehicles typically operate in predefined areas and conditions, leading to highly correlated data that may not capture the full diversity of real-world driving. Moreover, the cost of operating such vehicles means that the total driving time and geographical coverage are limited.

A more scalable approach is to collect data from customer vehicle fleets during regular operation. This approach naturally captures a much wider range of scenarios and conditions, as vehicles are genuinely being used in diverse environments and situations. Fleet vehicles are more likely to encounter rare but important events simply due to the massive amount of collective driving time. However, this approach presents its own challenges. Continuous data collection from thousands or millions of vehicles is infeasible due to bandwidth and storage limitations[57]. For context, a typical autonomous vehicle generates several terabytes of raw sensor data per hour of operation [58]. Even with aggressive compression and filtering, streaming this volume of data from millions of vehicles would require unprecedented infrastructure. Instead, intelligent data selection strategies are needed, such as those briefly discussed in section 3.2.3, to identify and preserve the most valuable data.

Furthermore, fleet vehicles are constrained to production-grade sensors, which limits the application of supervision techniques discussed in section 3.2.2. While some approaches remain viable, such as using production lidar to supervise camera perception, the inability to mount superior sensors on production vehicles precludes many advanced ground truth generation methods. The fleet approach also impacts trajectory data quality in complex ways. When collecting from human-driven vehicles, the trajectories reflect authentic driving behaviors and interactions, providing valuable data from the true driving distribution. However, these trajectories cannot be assumed to represent optimal or expert driving. The situation becomes particularly nuanced when collecting data from self-driving fleets, such as the current situation with Waymo's large fleet of robotaxis in San Francisco. Here, the human drivers have adjusted their driving behaviour as they learn the strengths and weaknesses of the self-driving system, thus potentially significantly skewing the collected data distribution away from "normal driving". To exaggerate the point, if the robotaxi's exhibit

slighltly over-aggressive behaviour in for example cut-ins, human drivers will learn that they need keep some additional margin to allow for this behaviour. This effect can amplify over time through a feedback loop: as humans adapt their behavior, the autonomous system learns from this adapted behavior, potentially leading to further behavioral changes in human drivers. This gradual distribution shift poses a significant challenge when deploying to new regions where drivers have not yet learned to accommodate autonomous vehicles - the learned behavior may be extremely unsafe when interacting with the original, unadapted distribution of human driving.

Even with fleet data collection, certain scenarios may remain difficult to capture. When collecting from human-driven fleets, market penetration limits geographical coverage - we cannot collect data from regions with insufficient market penetration (due to regulatory restrictions, local purchasing power, import tariffs, market preferences, etc.). When collecting from self-driving fleets, the operational design domain (ODD) of the autonomous system [59] naturally restricts data collection to conditions where the system is already operating reasonably well, leading to a catch-22 situation. While ideally the fleet can collect data from the boundary of its capabilities, in practice we may need to complement fleet collection with a small amount of targeted data gathering. This can be done using custom vehicles, equipped with additional sensors and driven by expert drivers in specific conditions or regions that are underrepresented in the fleet data - similar to mostly unsupervised learning, with a sprinkle of human supervision.

4.1.2 Open Datasets for Autonomous Driving (Paper III)

Several notable AD datasets have emerged over the years, each with its own strengths and limitations. The KITTI dataset [60], one of the earliest and most influential, provided carefully calibrated multi-sensor data and established many standard benchmarks still used today. nuScenes [61] significantly expanded upon this foundation, introducing a focus on temporal sequences and 360-degree perception through carefully synchronized multi-sensor data collection. The Waymo Open Dataset [62] and Argoverse (especially 2) [63, 64] further refined this paradigm, providing higher quality annotations and more diverse driving scenarios across multiple cities and weather conditions. While these datasets make reasonable design choices in focusing on temporal sequences and dense annotations in specific locations, this inherently leads to strong correlations between samples and limited data diversity.

These datasets, while crucial for academic research, represent only a tiny fraction of the data used in industrial autonomous driving development. Companies like Tesla and Waymo collect orders of magnitude more data through their vehicle fleets [57], with individual companies reporting datasets of millions of sequences compared to the few thousand available in open datasets. This massive disparity in data volume raises concerns about academic

research potentially overfitting to these relatively small datasets, particularly as companies demonstrate capabilities that seem completely unattainable on the scale of public datasets, creating a widening gap between academic and industrial development that risks slowing down overall progress in the field.

We designed the Zenseact Open Dataset (ZOD) to complement existing autonomous driving datasets, with a particular emphasis on geographical and environmental diversity. While existing datasets excel in certain aspects, such as dense temporal annotations or surround-view sensing, ZOD focuses on capturing the broad spectrum of real-world driving conditions across Europe. Although not collected from a true customer fleet, ZOD approximates many benefits of fleet-based collection through its extensive temporal and geographical coverage. By deliberately selecting samples that are well-distributed in both time and space, ZOD achieves greater scenario diversity than traditional datasets collected through short-term campaigns in limited areas.

ZOD is structured in three complementary tiers to support different research needs. At its core are 100,000 individual frames, carefully selected to maximize diversity across geographical locations, weather conditions, and driving scenarios, as demonstrated in fig. 4.1. This allows researchers to build robust single-frame perception models that can serve as foundation blocks for more complex temporal tasks. The diversity between frames is particularly valuable for self-supervised learning approaches, where having uncorrelated samples is often more beneficial than having many frames from the same sequence [6, 42]. For tasks requiring temporal context, ZOD provides approximately 1,500 sequences of similar length to other AD datasets (typically 15-20 seconds). Finally, for long-term prediction or planning tasks, ZOD includes about 30 extended driving sequences lasting several minutes each, enabling research into tasks requiring extended temporal context and evaluation of mapping or localization drift over longer distances.

The dataset features highly detailed and long-range annotations with 3D annotations extending up to 250 meters, while 2D annotations continue as long as objects remain visible, sometimes beyond 500 meters depending on conditions. This is complemented by multi-task annotations including linked 2D-3D objects, instance-level lane markings, and drivable road segmentation, see fig. 4.2. However, it's worth noting that ZOD currently lacks temporal annotations (such as object trajectories), which is an area where other datasets may be more suitable.

In terms of sensors, ZOD provides high-quality data from a focused sensor suite. The primary sensor is a front-facing 8MP camera with a 120-degree field of view, complemented by three lidar sensors including a top-of-the-line VLS-128, and recently released radar point clouds. While this configuration lacks the 360-degree camera coverage found in some other datasets, it emphasizes exceptional range and data quality in the driving direction.

Through this combination of geographical diversity, multi-tier structure, and detailed an-

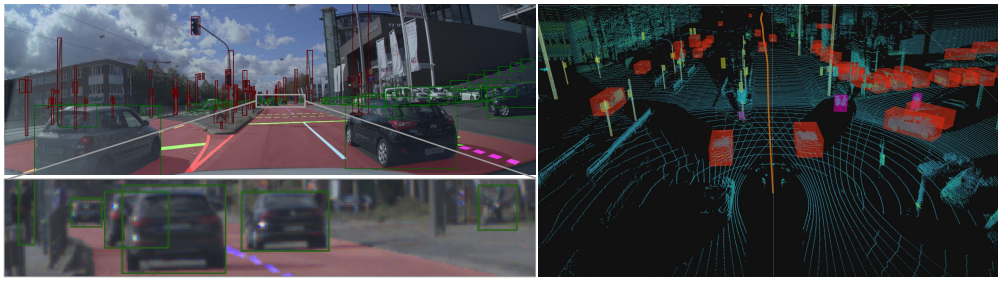


Figure 4.2: Examples of ZOD's detailed annotations, showing linked object detection at extended ranges and multi-task annotations including lanes and road segmentation. Illustration from Paper III.

notations, ZOD complements existing datasets by addressing specific gaps in the autonomous driving research ecosystem. While the dataset has begun to see adoption in the research community, we believe its full potential remains untapped. ZOD's unique strengths in environmental diversity and long-range perception could be better leveraged through dedicated challenges and benchmarks that specifically evaluate robustness across different environments and accuracy at extended ranges. As the autonomous driving field continues to mature, we expect the value of geographically and environmentally diverse data to become increasingly apparent, making ZOD's complementary nature to existing datasets even more significant.

4.2 Creating Digital Twins of Real Data

No matter how comprehensive our dataset collection efforts, we face a fundamental limitation: real-world data represents static recordings of past events, akin to a movie that plays out the same way every time. However, autonomous driving systems are inherently interactive - their actions influence how scenarios unfold. A slight change in the behavior of a self-driving car could have cascading effects that completely transform the outcome of a given situation, reminiscent of the butterfly effect where minor perturbations lead to dramatically different results over time.

Traditional data augmentation techniques have long been employed to artificially expand dataset diversity. Simple image-space transformations like color jittering, brightness adjustment, and geometric warping provide basic robustness to visual variations [65]. More sophisticated approaches leverage 3D scene understanding for structural modifications: depth-aware view synthesis enables realistic perspective changes [66], while instance-level copy-paste methods can insert new objects with proper occlusion handling [67]. Recent works have pushed the boundaries further with physics-based augmentations that simulate different weather conditions [68] and lighting scenarios [69]. However, these methods remain

fundamentally constrained by the originally recorded scenario and cannot reconfigure it to introduce novel interactions (e.g. changing the trajectory of a neighboring vehicle so that it performs a cut in scenario - necessitating a response from the ego vehicle).

Game engines and purpose-built simulators have emerged as a complementary approach, offering complete control over scenario generation. Early works demonstrated the potential of repurposing commercial games for autonomous driving research, with Grand Theft Auto V being particularly influential due to its realistic urban environments [70]. This inspired the development of dedicated simulators like CARLA [71], which provides precise control over weather, lighting, and actor behavior while maintaining physical accuracy. AirSim [72] extended similar capabilities to aerial vehicles, while NVIDIA's Drive Sim [73] showcases the potential of real-time ray tracing for sensor simulation. However, these approaches face a fundamental trade-off between visual fidelity and scenario complexity - achieving photorealistic rendering with accurate physics becomes computationally prohibitive at scale, especially when simulating multiple sensor modalities.

Recent advances in generative AI, particularly video diffusion models, have demonstrated remarkable capabilities in creating synthetic driving scenarios[74, 75]. These models can generate highly photorealistic sequences with unprecedented creativity - from mundane variations like weather changes to fantastical scenarios like Santa's sleigh landing in downtown traffic during a snowstorm. However, their limitations are severe: generated content often violates basic physics and geometry, lacks precise control over scene dynamics, and struggles with multi-sensor consistency. Moreover, the generation process is computationally expensive, making real-time interactive simulation impractical.

What we truly need is a method to convert our static driving logs into dynamic, interactive 3D environments that maintain physical realism while enabling scenario modification. Such "digital twins" would bridge the gap between real and synthetic data, combining the authenticity of real-world recordings with the flexibility of simulation. This capability would enable unprecedented experimentation, from exploring "what-if" scenarios by modifying weather conditions or object behaviors, to generating rare but critical cases that are difficult or dangerous to capture in real-world testing.

The most immediate application of such digital twins is closed-loop resimulation. While software development practices commonly employ nightly regression tests to catch unintended consequences of code changes, proper testing of AD systems requires more than just running new models on pre-recorded data. Open-loop evaluations fail to account for the interplay between a system's actions and its subsequent perceptions - for instance, whether accurate tracking is maintained during an emergency maneuver. Digital twins enable true closed-loop testing where the autonomous system's decisions actively influence how the scenario unfolds.

Recently emerging methods in neural scene reconstruction, such as Neural Radiance Fields

[76] and 3D Gaussian Splatting [77], offer a promising direction. These techniques learn to optimize a 3D scene representation directly from sensor data, in essence creating a learned game engine environment that maintains physical consistency with the real world and enables rendering of novel views. In the following sections, we will introduce the fundamentals of these methods and explore the unique challenges of adapting them to autonomous driving scenarios, where dynamic objects, multiple sensor modalities, and latency requirements create additional complexity beyond traditional static scene reconstruction.

4.2.1 Primer on Neural Rendering

Neural rendering is a rapidly evolving field that combines traditional computer graphics techniques with deep learning. While the term can be used somewhat ambiguously, here we refer to generating (rendering) sensor data from a learned (neural) 3D/4D scene representation.

The core idea is straightforward: given enough views (e.g. images) capturing an environment from different perspectives, along with a neural rendering process to generate new views from an underlying 3D representation, it is possible to learn that representation through pure optimization – in some sense brute-forcing the problem by minimizing the difference between rendered and observed views. While the success of this optimization approach is not guaranteed, the underlying principle is sound: only a representation that accurately captures the true scene geometry and appearance could successfully explain all observed views [1]. Imagine a chair, which has been photographed from all directions – the learned representation must include exactly 4 legs, as any additional (or missing) leg would be visible, and cause reconstruction errors, in some of the views.

There are of course some intricacies to this: we need our rendering process to be differentiable, so that we can backpropagate gradients to the scene representation, and physically plausible (for example respecting occlusions). Furthermore, we need a well-structured representation, with learnable parameters, that can effectively approximate the true environment and that exhibits reasonable convergence from the initial scene parameters. This combination of differentiable rendering and learnable parameters gives rise to the term *neural rendering*.

At the forefront of this field are Neural Radiance Fields (NeRFs) and, more recently, 3D Gaussian Splatting.

[1] There are some degenerate solutions, such as infinitesimal planes in front of each training image that essentially just display the image in question. However, in practice the learned representations have limited resolution and such near-camera artifacts will cause errors in other views that make them a sub-optimal solution – making it easier to learn the correct representation than to "cheat" in this way.

Neural Radiance Fields (NeRFs)

NeRFs [76] represent a scene as a continuous 5D function that maps a 3D location (x, y, z) and viewing direction (θ, ϕ) to a color (r, g, b) and volume density σ. This function is approximated by a neural network, originally a simple Multi-Layer Perceptron (MLP).

The rendering process in NeRFs is based on classical volume rendering. For each pixel in the desired output image, we cast a ray through the scene and sample points along this ray. The color $\boldsymbol{C}(\boldsymbol{r})$ of a ray $\boldsymbol{r}$ is computed using the volume rendering equation

$$\boldsymbol{C}(\boldsymbol{r}) = \int_{t_n}^{t_f} T(t)\sigma(\boldsymbol{r}(t))\boldsymbol{c}(\boldsymbol{r}(t), \boldsymbol{d})dt, \tag{4.1}$$

where t_n and t_f are the near and far bounds of the ray, $\boldsymbol{r}(\boldsymbol{t}) = \boldsymbol{o} + t\boldsymbol{d}$ is the ray with origin $\boldsymbol{o}$ and direction $\boldsymbol{d}$, $\boldsymbol{c}(\boldsymbol{r}(t), \boldsymbol{d})$ is the color at position $\boldsymbol{r}(t)$ in direction $\boldsymbol{d}$, $\sigma(\boldsymbol{r}(t))$ is the volume density at $\boldsymbol{r}(t)$, and $T(t)$ is the accumulated transmittance

$$T(t) = \exp\left(-\int_{t_n}^{t} \sigma(\boldsymbol{r}(s))ds\right). \tag{4.2}$$

In practice, this integral is approximated using numerical quadrature

$$\hat{\boldsymbol{C}}(\boldsymbol{r}) = \sum_{i=1}^{N} T_i(1 - \exp(-\sigma_i\delta_i))\boldsymbol{c}_i, \tag{4.3}$$

$$T_i = \exp\left(\sum_{j=1}^{i-1} \sigma_j\delta_j\right) \tag{4.4}$$

where δ_i is the distance between adjacent samples.

While the original NeRF achieved impressive results, it suffered from slow rendering times due to the need to evaluate the relatively large MLP for many points along each ray. Another way to think about it is that each sample, independently of location, had to execute the full MLP, which is a compressed version of the full scene, i.e. there is no locality. Subsequent work has focused on accelerating NeRFs, with Instant Neural Graphics Primitives (INGP)[78] being a notable advancement. INGP significantly accelerates NeRF rendering through several key innovations, including multi-resolution hash encoding, fully-fused MLP kernels, and an explicit occupancy grid. The hash-encoding stores learned embeddings in a local data structure, which means that the MLP has a much easier job and can be

much smaller, since it only needs to interpret a set of local embeddings, rather than memorizing the full scene. These improvements allow INGP to train and render high-quality novel views much faster than original NeRFs, leading to widespread adoption. Of course, there are many other ways to accelerate NeRFs, see e.g. [79–82].

Beyond NeRFs

While NeRFs took the field of neural rendering by storm in the early 2020s, the space of neural scene representations has rapidly diversified. Researchers have explored many variations of neural fields, encoding different types of information from geometric distances to semantic features. Some have moved away from fields entirely, exploring primitive-based representations that offer different trade-offs. These methods can be broadly categorized by both their underlying representation (fields vs. primitives) and rendering approach (ray tracing vs. rasterization)[2]. Interestingly, the evolution of neural rendering mirrors in many ways the development of computer game graphics - though in reverse, as neural rendering started with ray tracing of fields, whereas computer games started with rasterized triangles. And now both fields are converging to hybrid rasterization and ray tracing of geometric primitives.

Signed Distance Fields (SDFs) represent an early alternative field representation, with a long history in computer graphics, encoding scenes as continuous functions that measure the distance to the nearest surface at any point in space [83]. Their key advantages include a strong geometric prior that encourages sharp, well-defined surfaces, efficient ray marching through empty space, and natural handling of inside/outside queries which enables operations like boolean composition of shapes. However, they struggle with semi-transparent effects common in real-world scenes such as fog, motion blur, or partially reflective surfaces, as the binary surface representation cannot easily model varying densities.

Language-Enhanced Neural Fields (LERF) [84] represent another direction in field-based representations, replacing traditional RGB color with learned language embeddings at each point in space. This enables semantic queries and editing of the scene while maintaining the benefits of volumetric representation. Similar approaches have encoded other types of information into neural fields, from semantic segmentation [85] to self-supervised DINO features [11, 86], demonstrating the flexibility of the field-based paradigm.

3D Gaussian Splatting [77] marked a significant departure from field-based approaches by representing scenes as collections of oriented 3D Gaussian primitives. Each Gaussian is defined by its position, covariance matrix (determining its shape and orientation), and ap-

[2]Interestingly, while fields can be ray traced and primitives can be either ray traced or rasterized, rasterizing a continuous field is impossible by definition. Finding efficient approximations to this remains an intriguing research frontier.

pearance properties (color and opacity). The rendering process is highly efficient: each 3D Gaussian is projected to create a 2D Gaussian splat in screen space, which can be rasterized using modern graphics hardware. The method achieves remarkable speed by leveraging the rasterization pipeline and adapting the Gaussian density to scene complexity - areas with fine detail are represented by many small Gaussians, while smooth regions use fewer, larger primitives. This adaptive density, combined with the inherent soft nature of Gaussian primitives, results in efficient optimization and high-quality, high-speed rendering (100s-1000s of frames per second).

Recent work has sought to combine the benefits of primitive-based representations with more flexible rendering approaches. Several works [87, 88] adapt the Gaussian representation to a ray tracing framework, enabling accurate handling of complex camera models, shadows, reflections, and global illumination. Similarly, EVER (Exact Volumetric Ellipsoid Rendering) [89] provides analytical solutions for rendering ellipsoidal primitives, offering both mathematical elegance and practical efficiency. These hybrid approaches maintain many of the computational benefits of primitive-based representations while supporting the more accurate physics simulation needed for sensor modeling, and represent a promising direction for autonomous driving applications where both accuracy and speed are crucial.

Ideally one would have a unified representation that can be rendered using both rasterization and ray tracing, potentially rasterizing the first pass and then ray tracing any secondary effects, like reflections or shadows [88]. Hybrid approaches that combine the strengths of both paradigms represent a promising direction for future research, particularly in the context of autonomous driving where both accuracy and speed are crucial.

4.2.2 Neural Rendering for Autonomous Driving (Paper IV)

Adapting neural rendering techniques to autonomous driving scenarios presents several fundamental challenges that push these methods far beyond their original design parameters. While methods like NeRF and 3D Gaussian Splatting have shown impressive results on controlled datasets, autonomous driving data differs in almost every aspect:

First, driving scenes are inherently dynamic, with multiple objects moving at high speeds and potentially complex interactions between them. This violates the fundamental assumption of multi-view consistency that underlies most neural rendering approaches - a single 3D representation cannot explain all observations since they are inherently 4D (3D + time). The straightforward solution of using a 4D field [90] relies on having access to many different simultaneous views of the scene, which is of course not available in our setting with a single driving log, leaving the reconstruction extremely underconstrained.

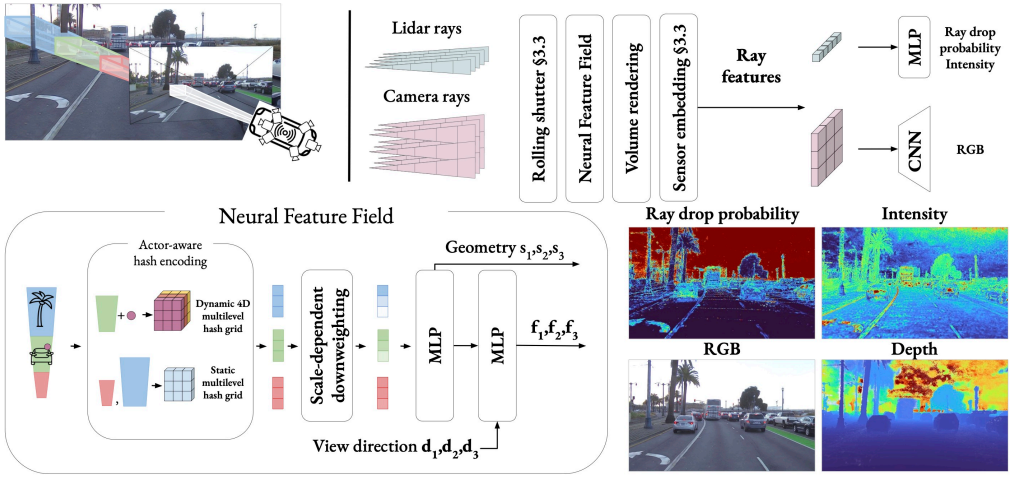


Figure 4.3: Method overview of NeuRAD. At the core lies the neural feature field which learns a joint representation of the entire scene, including dynamic objects, sky, and distant objects. This field allows us to render lidar and camera rays into abstract features, which are decoded to sensor space using modality-specific decoders. Notably, NeuRAD models important sensor characteristics like rolling shutter, lidar ray drop, and appearance variation across cameras. Illustration from Paper IV.

Second, the data collection geometry in driving scenarios is highly constrained. Unlike typical neural rendering datasets where cameras orbit around a central subject, vehicle-mounted cameras move roughly in a straight line, creating a highly anisotropic viewing distribution. This limited angular coverage makes it difficult to resolve depth ambiguities and leads to significant uncertainty in regions observed from only a single angle. The inside-out nature of the camera setup - where cameras point outward from roughly the same center point - further complicates matters compared to the outside-in setup common in object-centric neural rendering.

Third, the scale of driving scenes far exceeds typical neural rendering applications. While most NeRF scenes span a few meters, driving scenarios must handle hundreds of meters in every direction. This not only increases computational requirements but also creates challenges in balancing detail across different scales - nearby objects require high-resolution representation while distant objects may only occupy a few pixels yet remain semantically important.

Finally, autonomous driving data introduces several practical challenges. Vehicle sensor suites typically include multiple cameras with different intrinsics, fields of view, and mounting positions, all of which must be handled consistently. Additional sensor modalities like lidar and radar provide valuable 3D information but require careful sensor fusion. Fast-moving sensors introduce artifacts like rolling shutter and motion blur that must be explicitly modeled. The combination of these factors makes it impossible to directly apply existing neural rendering methods to autonomous driving data.

Naturally, there have been many attempts to address these challenges in the literature. Neural Scene Graphs [91] use available 3D annotations to create a scene graph, which treats moving objects as separate static NeRFs which can be rigidly inserted into the static NeRF using the object pose for a given timestamp. An additional bonus of this approach is that it allows for easy editing of the scene, by simply moving the object NeRFs around. UniSim [92] further builds on this idea, improving the neural field with ideas from INGP [78], adding a CNN upsampler to increase rendering speed, and adding a GAN to improve the quality of novel views - overall achieving very strong rendering results.

EmerNeRF [11] addresses the dynamic nature of driving scenes by having three fields, a 3D radiance field modelling the static environment, a 4D radiance field modelling the dynamic environment, and a separate 4D flow field. The flow forces the 4D field to be consistent under the flow transformation and both fields are regularized: acceleration for the flow to encourage smooth motions, and density for the dynamic field since most of the world should be static. Notably, all these fields can be optimized jointly using the typical rendering loss without any extra supervision (such as 2D optical flow). Besides the clear benefit of not needing annotations to reconstruct dynamic scenes, this approach is able to model non-rigid and non-annotated motions, such as pedestrians and flying debris. Furthermore, the learned flow field can be a useful supervision for various downstream tasks.

These methods, while promising, each come with distinct issues and limitations that make them unsuitable for simulating the complete autonomous driving sensor suite found in widely-used datasets. While they demonstrate impressive results on front-view scenarios, or in a camera-only setting, they struggle to handle the full complexity of multi-modal surround-view autonomous driving data. In Paper IV, we address these challenges through NeuRAD, a robust neural rendering framework designed to work with any autonomous driving dataset out of the box. Through extensive experimentation, we found that proper modeling of seemingly minor sensor effects proves crucial for achieving high-quality reconstruction. For instance, we explicitly model the rolling shutter effect, which becomes particularly pronounced during high-speed driving, especially in side cameras and the relatively slow-spinning surround lidars. Similarly, we account for lidar ray drop phenomena, which occur not only at long distances but also when rays glance off reflective surfaces [93]. Lastly, multi-camera setups can experience significant color inconsistencies between cameras due to variations in hardware specifications and image processing pipelines. These inconsistencies are further complicated by dynamic exposure adjustments during driving, such as when transitioning from dark tunnels into bright daylight.

In developing NeuRAD, we strived for architectural simplicity, with the idea that it would lead to better generalization across datasets and environments. While our high-level architecture draws inspiration from UniSim, particularly in using a feature field with an upsampling CNN decoder to improve rendering quality and speed, we developed a more streamlined yet expressive underlying field representation. This approach, incorporating

Figure 4.4: Qualitative examples showcasing the capabilities of NeuRAD. At the core is the ability to accurately re-render images, depth maps, and point clouds. By manipulating pose of the view that is rendered one can simulate a novel driving trajectory (shown on ZOD) and/or a different mounting position of the sensor (shown on PandaSet). Furthermore, since NeuRAD uses an explicit decomposition of static scene and dynamic actors, the actors can be removed or modified (shown on KITTI, NuScenes). All capabilities can of course be combined, resulting in realistic surround-consistent novel scenario simulation (shown on PandaSet). Illustration from Paper IV.

key insights from [94], enables efficient handling of dynamic elements and avoids explicit boundaries between near-and-far regions. The improved field representation also handles the extreme scale variations inherent in driving scenes, where the same object might be observed both at close range and hundreds of meters away.

The effectiveness of these design choices is demonstrated through comprehensive evaluation across five popular autonomous driving datasets, where NeuRAD achieves state-of-the-art performance in both quantitative metrics and qualitative assessment. To facilitate further research and development, we have released NeuRAD as an open-source framework, building upon the widely-used nerfstudio [95]. Encouragingly, the framework has seen some adoption within the community, with multiple groups and individuals building upon our work for various autonomous driving applications.

There are many potential improvements to NeuRAD around more efficient representations, deformable objects, faster rendering, secondary rays, etc. One particularly interesting avenue of research is solving the data collection geometry problem, where the collected views of the scene are very limited, using sensor-space generative models (e.g. video diffusion models). However, there is also an elephant in the room – we are missing an entire sensor modality. While radar data is frequently not provided in open AD datasets, it remains a very important sensor for commercial vehicles. And with the ascent of 4D radars [96], which provide much higher resolution than conventional radars, the importance of radar is not likely to decrease any time soon. While there has been some promising work on neural

rendering for radar data [97, 98], developing a unified representation that can handle the fundamentally different physical principles and noise characteristics of all sensor modalities remains an open challenge.

4.3 Simulating What Cannot Be Collected (Paper v)

The neural rendering techniques discussed in Section 4.2 provide a powerful foundation for converting static driving logs into dynamic, interactive environments. However, this capability raises an important question: how can we leverage these reconstructed environments to advance autonomous driving research and development? One of the most pressing challenges in autonomous driving is the difficulty of testing how driving algorithms perform in realistic, safety-critical scenarios. This challenge has become particularly acute with the rise of end-to-end driving systems, where we can no longer cleanly separate perception and planning components for independent testing in abstract, object-level simulators.

In industry, this challenge is typically addressed through extensive testing on dedicated tracks, where vehicles undergo standardized safety evaluations such as Euro NCAP tests [99]. These tests include scenarios like emergency braking for suddenly appearing pedestrians, responses to cut-in vehicles, and various other safety-critical situations. While effective, this approach requires significant infrastructure investment, is extremely time- and resource-intensive, and suffers from limited scenario diversity compared to real-world conditions. These limitations make track testing infeasible for most academic research and early-stage development, and highly impractical even for the big players in the industry.

Our work in Paper v demonstrates how neural rendering can bridge this gap. We introduce NeuroNCAP, a closed-loop simulation framework built around our neural renderer that enables NCAP-style safety testing using only existing driving logs. The framework operates by reconfiguring recorded scenarios to match standardized safety test configurations while maintaining the realism and complexity of the original scene. At its core, NeuroNCAP implements a simulation loop that:

0. Initializes the scenario by reconfiguring NeuRAD according to the scenario description.
1. Generates sensor observations given the current vehicle state.
2. Requests a trajectory prediction from the tested driving model (using the current sensor observations).
3. Computes acceleration and steering using a simple linear–quadratic regulator (LQR) controller, following [100].

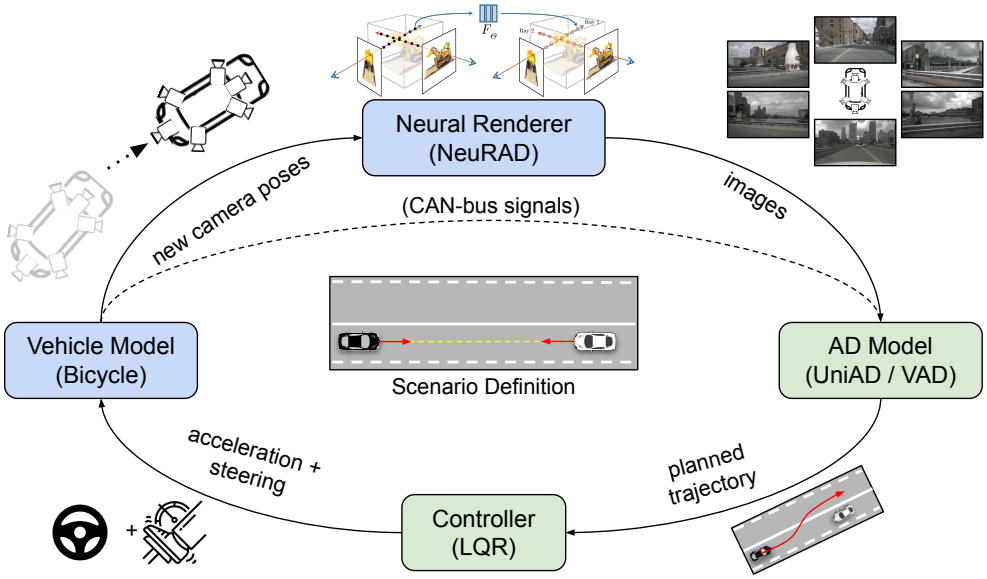


Figure 4.5: Closed-loop simulation diagram of NeuroNCAP. A scenario definition reconfigures a real-world driving log to safety-critical collision scenario. The neural renderer then generates realistic sensor data, which the AD model consumes to plan a driving trajectory. A simple controller and vehicle model propagate the ego pose forward in time, and this cycle repeats until the scenario is passed or a collision occurs. Illustration from Paper v.

4. Propagates the vehicle state using a bicycle dynamics model with realistic constraints.
5. Checks for collisions, if no collision go to step 1, otherwise abort the simulation.

Using the NuScenes dataset[61], which has become the de facto standard for evaluating end-to-end driving approaches, we create a suite of 20 test scenarios, with an allowed range of randomization per scenario. These scenarios correspond to three different collision risks: stationary obstacles, frontal collisions, and side impacts. For each scenario, we compute a very simple safety score (inspired by the 5-star NCAP scoring system) that only considers collision severity:

$$\text{NeuroNCAP Score} = \begin{cases} 5.0 & \text{if no collision} \\ 4.0 \cdot \max(0, 1 - v_i/v_r) & \text{otherwise} \end{cases}, \tag{4.5}$$

where v_i is the impact speed, and v_r is the reference impact speed that would occur if no action is performed.

Our evaluation reveals several important insights about current end-to-end driving approaches[101, 102]. Models that perform well on standard metrics often struggle in safety-critical scenarios, particularly when their actions can influence the scene evolution. This

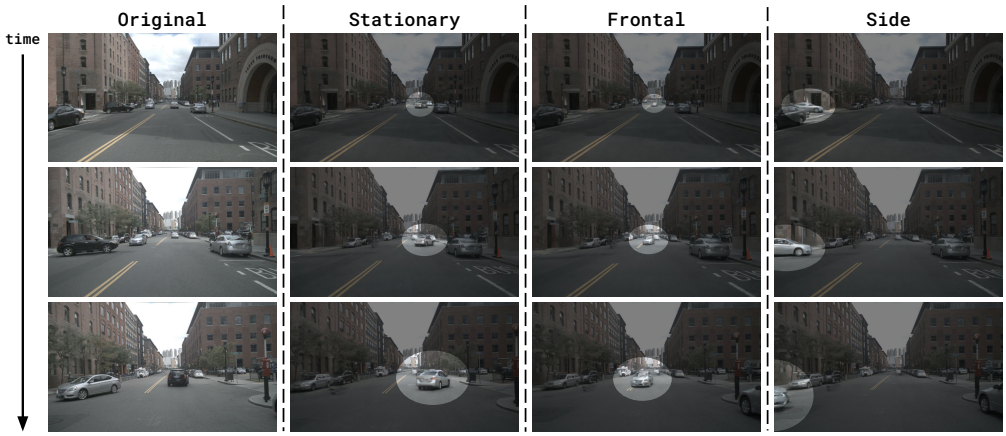


Figure 4.6: Example of the same original NuScenes scene (left) reconfigured and rendered for each type of NeuroNCAP collision scenario: stationary, frontal, and side. The shaded regions highlight the newly inserted object, and time flows from top to bottom. Illustration from Paper II.

suggests that current evaluation protocols, which typically focus on open-loop trajectory prediction, may not adequately capture the challenges of real-world autonomous driving. Furthermore, the ability to systematically vary the scenario parameters reveals edge cases and failure modes that would be difficult to identify through traditional testing methods.

To accelerate progress in this critical area of AD development, we publicly release our simulator and a suite of safety-critical scenarios as an easy-to-run evaluation package. This resource invites the research community to explore the behavior of their AD models in controlled yet highly configurable and challenging environments.

While our current scenario suite demonstrates the potential of this approach, it represents only a first step. The manual effort required to create these scenarios limits both their number and complexity. Future work should focus on automated scenario reconfiguration - developing algorithms that can automatically transform existing driving logs into diverse NCAP-like scenarios. Such automation would enable the creation of more sophisticated test cases across a much larger number of base scenes, providing broader coverage of potential edge cases and environmental conditions. Furthermore, just like in real NCAP testing, our actors are entirely on rails, meaning that they do not interact with each other or the tested vehicle. While reasonable since we want a controlled collision trajectory, more complex multi-agent scenarios would probably require some behaviour modelling (while still keeping at least one of the behaviours as dangerous and collision-causing).

The importance of simulation-based safety validation is increasingly recognized by regulatory bodies. Recent Euro NCAP reports explicitly acknowledge that physical test track evaluations cannot scale to cover the vast number of scenarios required for validating fully autonomous systems [103]. As the complexity of autonomous driving systems continues to

grow, we believe that neural rendering-based simulation will become the de facto approach for comprehensive safety validation, complementing limited physical testing with extensive virtual scenario exploration.

Chapter 5

Concluding Remarks

Throughout this work, we have tackled two major scaling bottlenecks in autonomous driving development, focusing on efficient supervision strategies, comprehensive data collection, and neural simulation techniques for safety validation. However, while we consider high-quality data and supervision to be the main scaling challenges, there are other aspects as well. We need both efficient algorithms whose capabilities continue to scale with the data and supervision, and we need enough compute to train these algorithms for long enough. Recent advances in large language models (LLMs) and diffusion models have shown incredibly promising scaling behavior[55, 104, 105], with models continuing to improve as we throw more compute and data at them, enabled by truly monstrous compute clusters. While this is promising, there may be some differences that prohibit autonomous driving development from directly following this recipe.

One such challenge lies in the fundamental nature of our data – while language models work with relatively compact text data (a few kilobytes per example), autonomous driving requires processing and storing massive amounts of synchronized multimodal data: multiple high-resolution video streams, dense point cloud sequences, radar data, and high-definition maps, easily reaching hundreds of megabytes per training example. All of these modalities must scale together, making the storage and processing requirements orders of magnitude more demanding.

Another potential scaling roadblock is the on-board compute. We have seen in general that model size, compute, and data need to be scaled together [106]. While techniques like distillation can greatly help compress the final model size [9], it remains a significant challenge to compress these increasingly powerful models to a point where they can be deployed on hardware that can run in the vehicle (as evidenced by Tesla's recent struggles with deploying their latest software version 13.x on older Hardware 3 compute platforms

[107]). Fortunately, time is on our side here, with continuous advances in both model efficiency techniques and on-board compute capabilities.

New scaling approaches are also emerging in the field. Recent work on language models has shown remarkable success with scaling test-time compute and sophisticated reasoning techniques [108–111], raising interesting possibilities for autonomous driving. As modern autonomous driving models become increasingly LLM-like in their architecture and capabilities, it is possible that we can adapt these same reasoning techniques. Importantly, this reasoning need not happen in explicit language - recent works have demonstrated that reasoning can occur effectively in latent space [112]. This approach seems more suitable for autonomous driving applications where the role of explicit natural language in generating core driving logic remains questionable[1]. A significant benefit of test-time compute scaling is that it naturally adapts to scenario complexity - allowing the vehicle to slow down and dedicate more processing time to particularly challenging situations can dramatically improve overall system capabilities. Naturally, some reasonable boundaries must be set, as we cannot spend minutes deliberating during time-critical scenarios like potential collisions. While this direction shows promise for advancing autonomous driving capabilities, it places even stronger demands on the on-board compute.

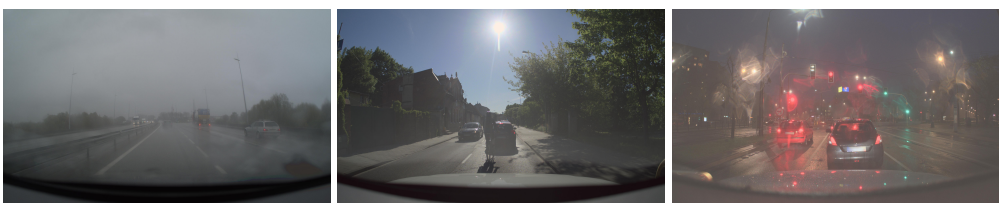
Our work on LidarCLIP points to an important consideration in this potential LLM-centric future for autonomous driving: these models must be able to reason effectively across all relevant modalities, not just visual data. However, it is still far from clear that they will be a core component of the self-driving stack - they may instead be used as tools for human interaction and explainability, or not have a place in the vehicle at all. Nevertheless, LLMs are still likely to play a crucial role offline, particularly in auto-annotation pipelines, where we previously highlighted the importance of human arbiters for extremely difficult or novel scenarios. It is entirely plausible that this role could be fulfilled by a powerful language model, acting as a proxy for human judgment and further increasing the scalability of our annotation processes.

All this raises some interesting thoughts on the role of humans in the future of autonomous driving development. We will likely continue to rely heavily on implicit human supervision, whether through fleet collection from human drivers, mining YouTube data (which humans have selected to upload), or using LLMs trained on vast amounts of human-generated text. However, the role of explicit human supervision will continue to diminish, especially in the training phase, as we increasingly rely on automated tools to reach the necessary volumes of data. For testing, even with our advances in neural simulation, it is hard to imagine that we will not want to perform some final verification with expert humans behind the wheel.

[1]While natural language is very suitable as a human-machine interface and can aids in system interpretability, the utilization of chain-of-thought reasoning through explicit language appears suboptimal for the real-time, continuous nature of the driving task.

Beyond fleet learning, web-scale video data presents another compelling opportunity for autonomous driving development. For example, a plethora of (in)famous Russian dashcam videos provide an incredible source of edge cases and unusual scenarios. This data could be valuable in multiple ways: either as a source of pre-training data to improve model robustness, or, connecting to our work on synthetic data, these videos could serve as an invaluable source of edge case scenarios. We could either generate a synthetic environment (like a NeRF) from the video where we can simulate the actual production sensors, or convert the videos to high-level scenarios which can then be used to transform existing driving logs into the desired scenarios (similar to what we do in NeuroNCAP).

Neural rendering has demonstrated significant promise as a simulation technology for autonomous driving and embodied robotics in general. However, certain phenomena remain challenging to simulate with this approach due to the simplified rendering process. This particularly applies to lens effects, such as water droplets on camera sensors during heavy rain, and flares from headlights at night. When such effects are combined, for example with oncoming headlights in heavy rain (see fig. 5.1), the limitations of current neural rendering processes become particularly apparent.



(a) water on windshield **(b)** sun glare **(c)** rainy and many lights

Figure 5.1: Heavy rain and strong sun glare, and the worst case when both are combined (headlights in heavy rain). These effects pose major difficulties for neural reconstruction methods like NeRFs and 3DGS.

Pure generative models, in contrast, do not exhibit particular difficulty with these effects compared to their handling of other phenomena. This raises the question: is neural rendering a dead end when it comes to these truly worst-case settings? One promising direction forward is a hybrid approach, combining neural rendering's precision and consistency with generative models functioning as a specialized effects layer, similar to post-processing in traditional computer graphics. This approach faces two primary challenges: maintaining geometric consistency while applying generative effects, and achieving the real-time performance necessary for practical applications. However, these challenges appear tractable given the strong prior information provided by the underlying neural rendering process.

Another promising direction is to move away from image-space generative models toward 3D-aware generative models whose outputs can be directly integrated with neural rendering. While this approach doesn't immediately address the challenges of heavy rain and lens effects, it enables us to leverage the full capabilities of generative models for creating

complex scenarios while maintaining geometric consistency. Recent work [113] has shown particular promise in generating diverse yet physically plausible driving scenarios, though significant challenges remain in achieving the necessary level of realism and fidelity.

A significant challenge we have only briefly touched upon is determining what scenarios to actually simulate. Even with a hypothetically perfect neural renderer or generative model, the question remains of how to use it effectively. The current iteration of NeuroNCAP, despite using only relatively simple scenarios, required significant manual effort to create. This process could feasibly be automated using trajectory-level diffusion models [114], LLM-based scenario generators [115], or reinforcement learning [116]. Once we have a way to generate plausible scenarios, we need to steer them towards cases that are relevant for the current model in an adversarial manner [116–118]. The challenge here lies in generating scenarios that are both challenging for the current model and relevant for real-world driving. Humans are particularly good at allowing certain theoretical collision possibilities because we understand their extremely low probability (e.g., oncoming traffic on a narrow road could suddenly swerve and cause a fatal accident). The goal is not to develop driving behavior that eliminates all crash possibilities, as such behavior would be impractically conservative in many real-world scenarios. Rather, the goal is to develop behavior that never causes an accident, while of course attempting to avoid potential accidents caused by other drivers whenever possible.

Finally, as I hinted earlier, the goal with our work on neural simulation is not just to enable efficient testing, but also to enable full closed-loop training. Recent advances in neural rendering, particularly the development of 3D Gaussian Splatting for autonomous driving [119], offer promising directions for improving the computational efficiency of these evaluations. Together with advances in feed-forward generalizable reconstruction methods [120, 121], which promise to generate neural simulation environments in mere seconds, it truly feels like all puzzle pieces are coming together and we are getting close to the end-game.

Bibliography

1. Sun, C., Shrivastava, A., Singh, S. & Gupta, A. K. *Revisiting Unreasonable Effectiveness of Data in Deep Learning Era* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2017), 843–852.
2. Acuna, D., Ling, H., Kar, A. & Fidler, S. *Efficient Interactive Annotation of Segmentation Datasets with Polygon-RNN++* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), 859–868.
3. Kirillov, A. *et al. Segment Anything* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023), 4015–4026.
4. Qi, C. R. *et al. Offboard 3d object detection from point cloud sequences* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), 6134–6144.
5. Ma, T. *et al. Detzero: Rethinking offboard 3d object detection with long-term sequential point clouds* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023), 6736–6747.
6. Chen, T., Kornblith, S., Norouzi, M. & Hinton, G. *A simple framework for contrastive learning of visual representations* in *Proceedings of the International Conference on Machine Learning (ICML)* (2020).
7. Caron, M. *et al. Emerging Properties in Self-Supervised Vision Transformers* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2021).
8. He, K. *et al. Masked Autoencoders Are Scalable Vision Learners* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 15979–15988.
9. Oquab, M. *et al.* DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research* (2024).
10. Radford, A. *et al. Learning Transferable Visual Models From Natural Language Supervision* in *Proceedings of the International Conference on Machine Learning (ICML)* **139** (2021), 8748–8763.

11. Yang, J. *et al. EmerNeRF: Emergent Spatial-Temporal Scene Decomposition via Self-Supervision* in *Proceedings of the International Conference on Learning Representations (ICLR)* (2024).

12. Rhinehart, N., Kitani, K. M. & Vernaza, P. *R2P2: A ReparameteRized Pushforward Policy for Diverse, Precise Generative Path Forecasting* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2018).

13. Agro, B., Sykora, Q., Casas, S., Gilles, T. & Urtasun, R. *UnO: Unsupervised Occupancy Fields for Perception and Forecasting* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024).

14. Kaplan, J. *et al.* Scaling Laws for Neural Language Models. *arXiv preprint arXiv:2001.08361* (2020).

15. Su, H., Deng, J. & Fei-Fei, L. *Crowdsourcing annotations for visual object detection* in *Proceedings of the Workshops at the 26th AAAI Conference on Artificial Intelligence* (2012).

16. Lee, J., Walsh, S., Harakeh, A. & Waslander, S. L. *Leveraging pre-trained 3d object detection models for fast ground truth generation* in *Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC)* (2018), 2504–2510.

17. Zhai, X., Kolesnikov, A., Houlsby, N. & Beyer, L. *Scaling vision transformers* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 12104–12113.

18. Das, A., Xian, Y., He, Y., Akata, Z. & Schiele, B. *Urban scene semantic segmentation with low-cost coarse annotation* in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (2023), 5978–5987.

19. Veit, A. *et al. Learning from noisy large-scale datasets with minimal supervision* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), 839–847.

20. Dosovitskiy, A. *et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale* in *Proceedings of the International Conference on Learning Representations (ICLR)* (2021).

21. Carion, N. *et al. End-to-end object detection with transformers* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2020), 213–229.

22. Achiam, J. *et al.* GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774* (2023).

23. Krizhevsky, A., Sutskever, I. & Hinton, G. E. *Imagenet classification with deep convolutional neural networks* in *Advances in Neural Information Processing Systems (NeurIPS)* **25** (2012).

24. Wei, J. *et al.* Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems (NeurIPS)* **35**, 24824–24837 (2022).

25. Lakshminarayanan, B., Pritzel, A. & Blundell, C. *Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles* in *Advances in Neural Information Processing Systems (NeurIPS)* **30** (2017).

26. Wang, Q., Chen, Y., Pang, Z., Wang, N. & Zhang, Z. Immortal tracker: Tracklet never dies. *arXiv preprint arXiv:2111.13672* (2021).

27. Wong, K., Gu, Y. & Kamijo, S. Mapping for autonomous driving: Opportunities and challenges. *IEEE Intelligent Transportation Systems Magazine* **13,** 91–106 (2020).

28. Jo, K. & Sunwoo, M. Generation of a precise roadway map for autonomous cars. *IEEE Transactions on intelligent transportation systems* **15,** 925–937 (2013).

29. Bao, Z., Hossain, S., Lang, H. & Lin, X. A review of high-definition map creation methods for autonomous driving. *Engineering Applications of Artificial Intelligence* **122,** 106125. ISSN: 0952-1976 (2023).

30. Peng, L. *et al. Lidar point cloud guided monocular 3d object detection* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2022), 123–139.

31. Chong, Z. *et al. MonoDistill: Learning Spatial Features for Monocular 3D Object Detection* in *Proceedings of the International Conference on Learning Representations (ICLR)* (2022).

32. Hekimoglu, A., Schmidt, M. & Marcos-Ramiro, A. *Monocular 3d object detection with lidar guided semi supervised active learning* in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (2024), 2346–2355.

33. Settles, B. *Active learning literature survey* tech. rep. (University of Wisconsin-Madison Department of Computer Sciences, 2009).

34. Feng, D., Wei, X., Rosenbaum, L., Maki, A. & Dietmayer, K. *Deep active learning for efficient training of a lidar 3d object detector* in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)* (2019), 667–674.

35. Kendall, A. & Gal, Y. *What uncertainties do we need in bayesian deep learning for computer vision?* in *Advances in Neural Information Processing Systems (NeurIPS)* **30** (2017).

36. Freytag, A., Rodner, E. & Denzler, J. *Selecting influential examples: Active learning with expected model output changes* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2014), 562–577.

37. Andriluka, M., Uijlings, J. R. & Ferrari, V. *Fluid annotation: a human-machine collaboration interface for full image annotation* in *Proceedings of the ACM International Conference on Multimedia (ICM)* (2018), 1957–1966.

38. Caruana, R. Multitask learning. *Machine Learning* **28,** 41–75 (1997).

39. Zhang, Y. & Yang, Q. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering* **34,** 5586–5609 (2021).

40. Ruder, S. An Overview of Multi-Task Learning in Deep Neural Networks. *arXiv preprint arXiv:1706.05098* (2017).

41. Wu, H. *et al.* Unleashing large-scale video generative pre-training for visual robot manipulation. *arXiv preprint arXiv:2312.13139* (2023).

42. Grill, J.-B. *et al. Bootstrap your own latent-a new approach to self-supervised learning* in *Advances in Neural Information Processing Systems (NeurIPS)* **33** (2020), 21271–21284.

43. Li, Z. *et al. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2022), 1–18.

44. Varma, M. *et al. Lift3D: Zero-Shot Lifting of Any 2D Vision Model to 3D* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 21367–21377.

45. Godard, C., Mac Aodha, O., Firman, M. & Brostow, G. J. *Digging into Self-Supervised Monocular Depth Prediction* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2019).

46. Khurana, T., Hu, P., Held, D. & Ramanan, D. Point Cloud Forecasting as a Proxy for 4D Occupancy Forecasting. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (2023).

47. Hess, G. *et al. Masked autoencoder for self-supervised pre-training on lidar point clouds* in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (2023), 350–359.

48. Chen, T., Kornblith, S., Swersky, K., Norouzi, M. & Hinton, G. E. Big self-supervised models are strong semi-supervised learners. *Advances in Neural Information Processing Systems (NeurIPS)* **33,** 22243–22255 (2020).

49. Najibi, M. *et al. Unsupervised 3d perception with 2d vision-language distillation for autonomous driving* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023), 8602–8612.

50. Chen, R. *et al. Clip2scene: Towards label-efficient 3d scene understanding by clip* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), 7020–7030.

51. Renz, K. *et al.* CarLLaVA: Vision language models for camera-only closed-loop driving. *arXiv preprint arXiv:2406.10165* (2024).

52. Cui, Y. *et al.* Drivellm: Charting the path toward full autonomous driving with large language models. *IEEE Transactions on Intelligent Vehicles* (2023).

53. Touvron, H. *et al.* Llama: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971* (2023).

54. Touvron, H. *et al.* Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023).

55. Dubey, A. *et al.* The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).

56. Hell, L., Sprenger, J., Klusch, M., Kobayashi, Y. & Müller, C. *Pedestrian behavior in japan and germany: A review* in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)* (2021), 1529–1536.

57. Tesla, Inc. *Tesla AI Day 2022* Video presentation. Available at `https://www.youtube.com/watch?v=ODSJsviD_SU&t=4800s` [Accessed: 2025-12-05]. 2022.

58. Bloomberg News. *Carmakers Look to Satellites for Future of Self-Driving Vehicles* `https://www.bloomberg.com/news/articles/2021-09-17/carmakers-look-to-satellites-for-future-of-self-driving-vehicles`. Published on September 17, 2021. Accessed: 13 January 2025.

59. International, S. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles. *SAE Standard J3016* (2021).

60. Geiger, A., Lenz, P., Stiller, C. & Urtasun, R. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research* **32,** 1231–1237 (2013).

61. Caesar, H. *et al. nuscenes: A multimodal dataset for autonomous driving* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), 11621–11631.

62. Sun, P. *et al. Scalability in Perception for Autonomous Driving: Waymo Open Dataset* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), 2443–2451.

63. Chang, M.-F. *et al. Argoverse: 3d tracking and forecasting with rich maps* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), 8748–8757.

64. Wilson, B. *et al.* Argoverse 2: Next generation datasets for self-driving perception and forecasting. *arXiv preprint arXiv:2301.00493* (2023).

65. Shorten, C. & Khoshgoftaar, T. M. A survey on image data augmentation for deep learning. *Journal of Big Data* **6,** 1–48 (2019).

66. Park, E., Yang, J., Yumer, E., Ceylan, D. & Berg, A. C. *Transformation-grounded image generation network for novel 3d view synthesis* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), 3500–3509.

67. Dwibedi, D., Misra, I. & Hebert, M. *Cut, paste and learn: Surprisingly easy synthesis for instance detection* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2017), 1301–1310.

68. Tremblay, M., Halder, S. S., De Charette, R. & Lalonde, J.-F. Rain rendering for evaluating and improving robustness to bad weather. *International Journal of Computer Vision* **129,** 341–360 (2021).

69. Dai, D. & Van Gool, L. *Dark model adaptation: Semantic image segmentation from daytime to nighttime* in *Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC)* (2018), 3819–3824.

70. Richter, S. R., Vineet, V., Roth, S. & Koltun, V. *Playing for data: Ground truth from computer games* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2016), 102–118.

71. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A. & Koltun, V. *CARLA: An open urban driving simulator* in *Proceedings of the Conference on Robot Learning (CORL)* (2017), 1–16.

72. Shah, S., Dey, D., Lovett, C. & Kapoor, A. *Airsim: High-fidelity visual and physical simulation for autonomous vehicles* in *Field and Service Robotics: Results of the 11th International Conference* (2018), 621–635.

73. NVIDIA. *NVIDIA DRIVE Sim* Accessed: 2024-12-31. 2021.

74. Hu, A. *et al.* GAIA-1: A Generative World Model for Autonomous Driving. *arXiv preprint arXiv:2309.17080* (2023).

75. Xing, Z. *et al.* A survey on video diffusion models. *ACM Computing Surveys* **57,** 1–42 (2024).

76. Mildenhall, B. *et al. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2020).

77. Kerbl, B., Kopanas, G., Leimkühler, T. & Drettakis, G. 3d gaussian splatting for real-time radiance field rendering. *Proceedings of the ACM SIGGRAPH Conference* (2023).

78. Müller, T., Evans, A., Schied, C. & Keller, A. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics* **41,** 1–15 (2022).

79. Fridovich-Keil, S. *et al. Plenoxels: Radiance fields without neural networks* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 5501–5510.

80. Fridovich-Keil, S., Meanti, G., Warburg, F. R., Recht, B. & Kanazawa, A. *K-planes: Explicit radiance fields in space, time, and appearance* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), 12479–12488.

81. Chen, A., Xu, Z., Geiger, A., Yu, J. & Su, H. *Tensorf: Tensorial radiance fields* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2022), 333–350.

82. Yariv, L. *et al. Bakedsdf: Meshing neural sdfs for real-time view synthesis* in *Proceedings of the ACM SIGGRAPH Conference* (2023), 1–9.

83. Park, J. J., Florence, P., Straub, J., Newcombe, R. & Lovegrove, S. DeepSDF: Learning continuous signed distance functions for shape representation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* 165–174 (2019).

84. Kerr, J., Kim, C. M., Goldberg, K., Kanazawa, A. & Tancik, M. *Lerf: Language embedded radiance fields* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023), 19729–19739.

85. Chou, Z.-T., Huang, S.-Y., Liu, I., Wang, Y.-C. F. *et al. GSNeRF: Generalizable Semantic Neural Radiance Fields with Enhanced 3D Scene Understanding* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 20806–20815.

86. Kobayashi, S., Matsumoto, E. & Sitzmann, V. Decomposing nerf for editing via feature field distillation. **35,** 23311–23330 (2022).

87. Moenne-Loccoz, N. *et al.* 3D Gaussian Ray Tracing: Fast Tracing of Particle Scenes. *ACM Transactions on Graphics* (2024).

88. Wu, Q., Esturo, J. M., Mirzaei, A., Moenne-Loccoz, N. & Gojcic, Z. 3DGUT: Enabling Distorted Cameras and Secondary Rays in Gaussian Splatting. *arXiv preprint arXiv:2412.12507* (2024).

89. Mai, A. *et al.* Ever: Exact volumetric ellipsoid rendering for real-time view synthesis. *arXiv preprint arXiv:2410.01804* (2024).

90. Pumarola, A., Corona, E., Pons-Moll, G. & Moreno-Noguer, F. *D-nerf: Neural radiance fields for dynamic scenes* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), 10318–10327.

91. Ost, J., Mannan, F., Thuerey, N., Knodt, J. & Heide, F. *Neural scene graphs for dynamic scenes* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), 2856–2865.

92. Yang, Z. *et al. Unisim: A neural closed-loop sensor simulator* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), 1389–1399.

93. Huang, S. *et al. Neural lidar fields for novel view synthesis* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023), 18236–18246.

94. Barron, J. T., Mildenhall, B., Verbin, D., Srinivasan, P. P. & Hedman, P. *Zip-nerf: Anti-aliased grid-based neural radiance fields* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023), 19697–19705.

95. Tancik, M. *et al. Nerfstudio: A modular framework for neural radiance field development* in *Proceedings of the ACM SIGGRAPH Conference* (2023), 1–12.

96. Fan, L. *et al.* 4D mmWave radar for autonomous driving perception: a comprehensive survey. *IEEE Transactions on Intelligent Vehicles* (2024).

97. Borts, D. *et al. Radar Fields: Frequency-Space Neural Scene Representations for FMCW Radar* in *Proceedings of the ACM SIGGRAPH Conference* (2024).

98. Huang, T. *et al. DART: Implicit Doppler Tomography for Radar Novel View Synthesis* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), 24118–24129.

99. Euro NCAP. *Euro NCAP* 2025. `https://www.euroncap.com`.

100. Caesar, H. *et al. NuPlan: A closed-loop ML-based planning benchmark for autonomous vehicles* in *CVPR ADP3 workshop* (2021).

101. Hu, Y. *et al. Planning-oriented autonomous driving* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), 17853–17862.

102. Jiang, B. *et al. Vad: Vectorized scene representation for efficient autonomous driving* in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023), 8340–8350.

103. Euro NCAP. *Euro NCAP 2023 Protocol* 2023. `https://www.euroncap.com/en/for-engineers/protocols/`.

104. Labs, B. F. *Announcing Black Forest Labs* Accessed: 2025-01-03. 2024.

105. Esser, P. *et al. Scaling rectified flow transformers for high-resolution image synthesis* in *Proceedings of the International Conference on Machine Learning (ICML)* (2024).

106. Hoffmann, J. *et al.* Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556* (2022).

107. Tesla, I. *Tesla Q3 2024 Earnings Call Transcript* Earnings call audio/transcript. 2024. `https://ir.tesla.com/` (2025).

108. Snell, C., Lee, J., Xu, K. & Kumar, A. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314* (2024).

109. Gulcehre, C. *et al.* Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998* (2023).

110. Wang, X. *et al.* Self-consistency improves chain of thought reasoning in language models. *Proceedings of the International Conference on Learning Representations (ICLR)* (2023).

111. Kumar, A. *et al.* Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917* (2024).

112. Hao, S. *et al.* Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769* (2024).

113. Ren, X. *et al. SCube: Instant Large-Scale Scene Reconstruction using VoxSplats* in *Advances in Neural Information Processing Systems (NeurIPS)* (2024).

114. Jiang, C. M. *et al.* SceneDiffuser: Efficient and Controllable Driving Simulation Initialization and Rollout. *arXiv preprint arXiv:2412.12129* (2024).

115. Tan, S., Ivanovic, B., Weng, X., Pavone, M. & Kraehenbuehl, P. *Language Conditioned Traffic Generation* in *Proceedings of the Conference on Robot Learning (CORL)* (2023), 2714–2752.

116. Rowe, L. *et al. CtRL-Sim: Reactive and Controllable Driving Agents with Offline Reinforcement Learning* in *Proceedings of the Conference on Robot Learning (CORL)* (2024).

117. Hanselmann, N., Renz, K., Chitta, K., Bhattacharyya, A. & Geiger, A. *King: Generating safety-critical driving scenarios for robust imitation via kinematics gradients* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2022), 335–352.

118. Zhang, C. *et al. Learning to Drive via Asymmetric Self-Play* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2024).

119. Hess, G., Lindström, C., Fatemi, M., Petersson, C. & Svensson, L. SplatAD: Real-Time Lidar and Camera Rendering with 3D Gaussian Splatting for Autonomous Driving. *arXiv preprint arXiv:2411.16816* (2024).

120. Yang, J. *et al.* STORM: Spatio-Temporal Reconstruction Model for Large-Scale Outdoor Scenes. *arXiv preprint arXiv:2501.00602* (2024).

121. Chen, Y., Wang, J., Yang, Z., Manivasagam, S. & Urtasun, R. *G3R: Gradient Guided Generalizable Reconstruction* in *Proceedings of the European Conference on Computer Vision (ECCV)* (2024).

Scientific publications

Printed by Media-Tryck, Lund 2025 NORDIC SWAN ECOLABEL 3041 0903

Doctoral Theses in Mathematical Sciences 2025:1
ISBN 978-91-8104-298-6
ISSN 1404-0034