A Layered Foundation for Reliable Trajectory Forecasting: Data, Evaluation, and Methods

Erica Weng

CMU-RI-TR-26-10

February 28, 2026

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee:
Kris Kitani, *Co-chair, Carnegie Mellon University*
Deva Ramanan, *Co-chair, Carnegie Mellon University*
Aaron Steinfeld, *Carnegie Mellon University*
Hamid Rezatofighi, *Monash University*

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Robotics.

Abstract

Reliable trajectory forecasting is a foundational requirement for autonomous robotic systems operating in environments with humans, where reliability means producing predictions that are collision-free, socially consistent, and robust across both routine and safety-critical scenarios. Despite substantial progress in modeling techniques, existing forecasting systems often fail under distribution shift, exhibit socially implausible behaviors, or report misleading performance. The field has largely treated these as modeling problems and thus has invested heavily in ever more expressive architectures while under-investing in the infrastructure that models depend on. This thesis takes a different position: that reliable trajectory forecasting requires treating data curation, evaluation design, and modeling as co-equal engineering challenges, organized as a layered stack where each layer depends on the soundness of those below it. Good methods are only as useful as the benchmarks that evaluate them, and good benchmarks are only as meaningful as the data that underlies them.

Forecasting systems are only as reliable as the data they learn from, yet current datasets systematically under-represent the rare, safety-critical tail behaviors that matter most for deployment. We present JaywalkerVR, a Virtual Reality human-in-the-loop system, and the CARLA-VR dataset of safety-critical pedestrian-vehicle interactions collected with it. We show that this incomplete coverage significantly impairs forecasting reliability, and that augmenting training data with VR-collected interactions yields 10.7% lower displacement error and 4.9% lower collision rate on interactive scenarios, establishing the base layer upon which meaningful evaluation and modeling must rest.

Even with better data, progress is illusory if we measure it poorly. Widely used forecasting metrics obscure critical failure modes such as collisions and socially implausible interactions, giving a false sense of readiness for deployment. Building on the data foundation, we introduce joint evaluation metrics (JADE, JFDE) and collision rate, revealing a $2\times$ gap between marginal and joint performance. Optimizing for joint metrics with no architectural changes yields a 16% collision rate reduction, confirming that evaluation design directly shapes the models the community builds. Without these metrics, improvements in model design cannot be trusted to reflect genuine progress.

Only once data and evaluation are sound does it become productive to ask how we can improve these models. Building on these foundations, we present PECT (Pose and Environment-Contextualized Transformer), a three-stream architecture that incorporates human body pose and dense Bird's Eye View environmental semantics alongside trajectory history. We introduce the environment collision rate (ECR) metric and a gated curriculum fusion strategy that aligns trajectory, pose, and dense environment features so that the additional modalities improve collision avoidance rather than introducing noise. PECT improves agent-agent collision rate by 6–12% and environment collision rate by 8–10%, without sacrificing displacement accuracy. The value of these richer inputs is only legible because the underlying data coverage and evaluation criteria are equipped to surface the differences that matter.

Taken together, this thesis argues that the trajectory forecasting community should approach deployment readiness not as a modeling problem but as a systems problem. Data, evaluation, and methods are deeply interdependent — neglecting any one undermines the others. By addressing all three as a unified stack, this work contributes a framework, concrete tools, and a philosophy for building forecasting systems genuinely aligned with the demands of real-world autonomous decision-making.

Contents

When this dissertation is viewed as a PDF, the page header is a link to this Table of Contents.

List of Figures

List of Tables

Chapter 1

Introduction

Autonomous robotic systems operating in environments with humans such as self-driving vehicles, delivery robots, and assistive mobile platforms, must anticipate where the people around them will be in the near future. This capability, trajectory forecasting, is a foundational requirement for safe autonomous decision-making: a self-driving car must predict whether a pedestrian will step into the road, and a campus robot must anticipate whether a group of students will stay together or disperse. Getting these predictions right enables smooth, safe, and socially acceptable robot behavior; getting them wrong can lead to dangerous collisions, unnecessary emergency stops, or erratic motion that erodes public trust in autonomous systems.

The trajectory forecasting community has made substantial progress over the past decade. Deep generative models [67, 176, 242], graph neural networks and transformers [80, 177, 245], and hierarchical architectures [134, 228, 252] have driven steady improvements on standard benchmarks such as ETH/UCY [110, 154], Stanford Drone Dataset [171], nuScenes [18], and the Waymo Open Motion Dataset [46]. Yet despite this progress, existing forecasting systems can remain unreliable in the situations where reliability matters most. Models trained on current datasets can fail catastrophically on rare but safety-critical events, such as a child darting into the road or a wheelchair user navigating a construction zone. Standard evaluation metrics can report excellent performance while failing to detect physically implausible predictions, such as trajectories where pedestrians walk through one another or pass through walls. And most state-of-the-art methods rely on trajectory history and vector map information alone, ignoring rich signals such as body pose and environmental affordances, which humans routinely use to anticipate one another's behavior.

Data gaps, evaluation blind spots, and underutilized sensor information: these three shortcomings are not independent problems but interconnected facets of a deeper challenge. A model trained on biased data cannot be expected to generalize; a metric that conceals failure modes cannot guide improvement; and a method that ignores available signals cannot fully exploit the data it does have. Meaningful progress requires addressing all three.

This thesis argues that reliable trajectory forecasting does not depend on model architecture alone, and that addressing any one of these shortcomings in isolation is insufficient. Instead, it requires a *layered foundation*: good data to train on, good benchmarking to measure progress faithfully, and good methods that leverage the full richness of available sensor information—each layer building on the ones below it.

These three layers form a pyramid, illustrated in Figure 1.1. *Data* is the foundation: the distribution of training examples determines what behaviors a model can learn, and gaps in data coverage translate directly into gaps in prediction capability. Data coverage also determines what can be *evaluated*: a benchmark that lacks edge-case and long-tail scenarios cannot measure a model's performance on precisely the situations where failures carry the greatest real-world consequences—a single collision is far more costly than a thousand slightly suboptimal but safe predictions. *Evaluation* is the middle layer: the metrics and benchmarking protocols we use determine which improvements are visible and which are hidden, shaping the optimization objectives that guide model development. *Methods* sit at the top: forecasting architectures that incorporate multiple sensor modalities can produce more accurate and socially plausible predictions, but only when trained on comprehensive data and assessed on benchmarks that reward the right properties. The dependence is primarily upward: methods rest on evaluation, which rests on data; though feedback flows downward as well, with method failures revealing data gaps and method capabilities motivating new evaluation criteria. For example, a method that incorporates body pose may expose that existing datasets lack sufficient pose annotations, while a collision-aware method may motivate new metrics that existing benchmarks do not report.

This thesis addresses all three layers in turn, demonstrating that each contributes meaningfully and that their combination yields improvements that no single layer can achieve alone. The following sections preview the key challenges and contributions at each layer, motivating the specific problems addressed in Chapters 2–4.

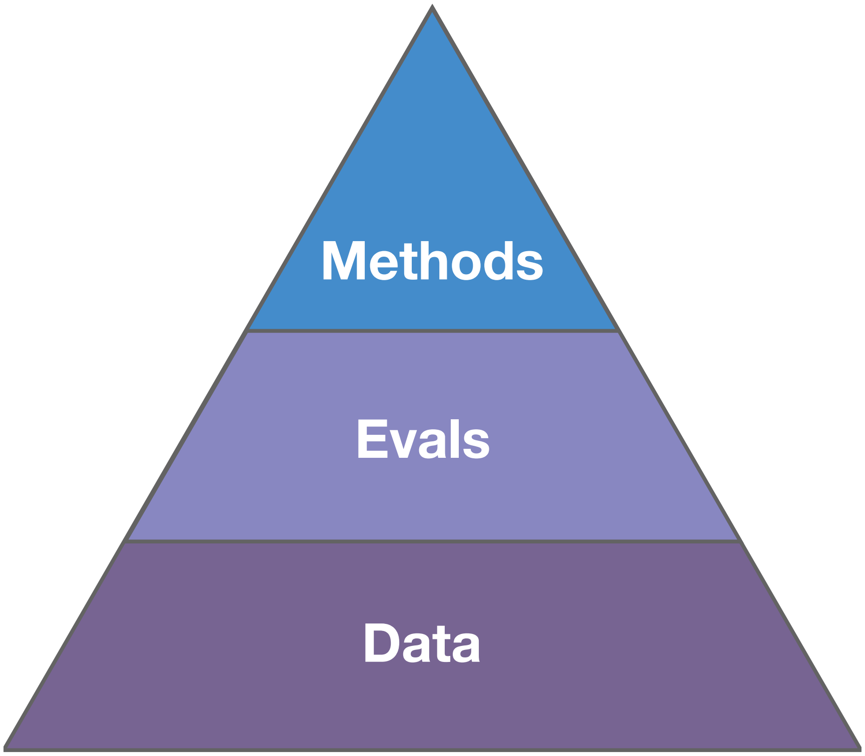


Figure 1.1: The layered foundation for reliable trajectory forecasting. Data forms the base: comprehensive coverage of rare, safety-critical scenarios is a prerequisite for both learning and evaluating robust predictions. Evaluation occupies the middle layer: evaluation benchmarks depend on good underlying data coverage to measure progress comprehensively, and metrics must faithfully measure the properties required for downstream deployment. Methods sit at the top: forecasting architectures can only succeed when supported by the data and evaluation layers beneath them.

1.1 Data: Comprehensive Coverage

The first layer concerns what models learn from. Modern trajectory forecasting models are data-hungry deep learning systems, and their effectiveness is ultimately constrained by the training data available to them. Existing autonomous driving datasets, including nuScenes [18], Waymo Open Motion Dataset [46], and Argoverse [26], contain vast amounts of data collected from real-world driving, but they are dominated by routine scenarios: vehicles following lanes, pedestrians crossing at marked crosswalks, orderly traffic flow. Safety-critical situations involving vulnerable road users, such as close vehicle-pedestrian encounters, and interactions involving children or elderly individuals are inherently rare in naturalistic data collection. Moreover, deliberately creating such scenarios for data collection purposes raises serious ethical and safety concerns: one cannot ethically ask a child to jaywalk across a busy road to collect training data.

This long-tail data problem has direct consequences: a model trained predominantly on routine interactions may extrapolate lane-following behavior with high accuracy while failing entirely on a jaywalker emerging between parked cars, making it essential to develop methods for collecting diverse, safety-critical interaction data.

Simulated environments offer one potential avenue for generating rare scenarios at scale. Simulators like CARLA [44] enable the generation of arbitrary scenarios, including dangerous situations that would be impossible to capture in real-world data collection, and recent advances in neural rendering [136, 205, 234, 238, 256] have largely closed the *sensor* realism gap, producing visually and geometrically convincing synthetic environments. However, these approaches treat pedestrians purely as visual assets, not as behaving entities. Synthetic data still suffers from a fundamental *behavioral* sim-to-real gap: simulated pedestrians exhibit mechanical, rule-based behaviors that fail to capture the nuances of real human decision-making— the hesitation before crossing, the head turn to check for traffic, the subtle body language that precedes a change in direction, waving to signal intent, or body language and gesturing that indicate human interaction and social group dynamics.

Chapter 2 addresses this gap by presenting JaywalkerVR, a Virtual Reality human-in-the-loop system for collecting safety-critical pedestrian-vehicle interaction data. The key insight is that while the environment is simulated, the human behavior within it is real: participants wearing VR headsets genuinely respond to virtual traffic as they would to real traffic, producing naturalistic trajectories that capture the complexity of human decision-making in dangerous scenarios. Using this system, we collect the CARLA-VR dataset and demonstrate that augmenting model training data with VR-collected interactions leads to significant improvements in forecasting performance on safety-critical scenarios. We validate the realism of the collected data through both first-person user studies and third-person evaluations, and confirm through expert interviews that the long-tail data problem is a recognized challenge in the autonomous vehicle industry.

1.2 Evaluation: Measuring What Matters

Better data is necessary but not sufficient for progress—we also need evaluation protocols that faithfully measure model performance. Without good evaluation, even well-trained models may appear to succeed while harboring critical failure modes that remain undetected until deployment. The second layer of reliable forecasting concerns how we benchmark models, and this thesis demonstrates that standard practices in trajectory forecasting evaluation have significant limitations that can mislead progress assessment. Because evaluation metrics serve as both optimization targets and comparison tools, flawed metrics do not merely give an incomplete picture of progress; they actively misdirect the field by rewarding the wrong properties and hiding critical failure modes.

The dominant evaluation metrics in pedestrian trajectory forecasting, *ADE* and *FDE*, evaluate each agent's predicted trajectory independently, selecting the best prediction sample per agent. This *marginal* evaluation allows "mix-and-match" across prediction samples, crediting models for coherent joint predictions they never actually made. The consequences are concrete: a model can achieve excellent marginal scores while consistently producing physically implausible predictions, because these failure modes are invisible to metrics that evaluate each agent in isolation.

The core insight is that trajectory forecasting is inherently a *joint* prediction problem: the future trajectories of interacting agents are constrained by social and physical dynamics. A model that produces excellent individual predictions but never assembles them into a coherent joint future is fundamentally limited for downstream planning, where an autonomous system must reason about what *will* happen, not an artificial combination of what *might* happen to different agents independently.

Chapter 3 addresses this limitation by advocating for *joint metrics* in trajectory forecasting evaluation. Joint metrics require that all agents' predictions come from the same sample, ensuring that high scores require coherent multi-agent futures rather than artificially assembled combinations. We present, to our knowledge, the first comprehensive evaluation of state-of-the-art pedestrian forecasting methods under joint metrics, revealing a striking gap: joint metric performance is typically twice as poor as marginal performance, indicating that current methods are far less capable of modeling multi-agent interactions than standard benchmarks suggest. Furthermore, we demonstrate that optimizing for joint metrics by modifying loss functions to include joint terms, with no architectural changes, yields substantial improvements in joint prediction quality and collision avoidance, confirming that the evaluation-optimization coupling runs deep: the metrics we use shape the models we build.

1.3 Methods: Leveraging Rich Sensor Information

With good data to train on and good evaluation to measure progress, the third layer turns to the forecasting methods themselves. The central question is: what information sources should a trajectory prediction model use, and how should it integrate them?

For vehicle trajectory prediction, the answer has largely converged: trajectory history plus HD map information suffices for most scenarios, since lane boundaries, traffic rules, and right-of-way conventions constrain motion to well-defined corridors. Nearly all top-performing methods on the Waymo Open Motion Dataset leaderboard [186, 187] use only vectorized agent history and HD map polylines as input, foregoing camera and LiDAR entirely. Pedestrian prediction faces a fundamentally different challenge. Pedestrians can move in any direction, are not bound to marked paths, and make decisions based on intent, social context, and environmental affordances that trajectory history alone cannot capture. While a pedestrian *typically* walks on sidewalks and crosses at marked crosswalks, they *can* walk on grass, cut through parking lots, sit at outdoor tables, or weave between parked cars. The semantics of pedestrian environments are less structured than those of roads: environmental features encode soft preferences rather than hard constraints. A patch of grass is traversable but unpreferred; a café table is an obstacle to some pedestrians and a destination to others; a doorway entrance is a barrier from one perspective and a goal from another. This semantic ambiguity is largely absent in the vehicle domain, where the meaning of road elements is unambiguous and legally codified. The contrast highlights an asymmetry: while vehicle prediction benefits from strong structural priors embedded in maps, pedestrian prediction must infer these priors from richer sensory inputs.

This ambiguity makes purely trajectory-based prediction insufficient. Historical position and velocity alone cannot distinguish whether a pedestrian approaching a grassy median will walk around it or cut straight through, or whether someone angling toward a café is passing by or stopping. Two information sources can resolve these ambiguities. *Body pose and orientation* reveal intent before it manifests in observable motion, like a pedestrian turning their head both ways to check for vehicles before crossing. These behavioral cues are available seconds before the corresponding trajectory change, providing an early warning signal that trajectory-only methods miss entirely. *Environmental context*, meaning the semantic understanding of traversable surfaces, obstacles, and spatial affordances, provides scene-specific priors that constrain feasible motion in ways that sparse map polylines cannot capture. A Bird's Eye View (BEV) representation encoding sidewalks, grass, building facades, and street furniture captures rich contextual information about where pedestrians can and cannot go, and where they are likely or unlikely to go, enabling predictions that respect the physical layout of the scene.

Chapter 4 presents PECT (Pose and Environment-Contextualized Transformer), a multimodal trajectory prediction framework that explicitly incorporates human body pose signals and dense environmental semantics alongside trajectory history. A key technical challenge is integrating three fundamentally different modalities without the noisier or domain-shifted signals corrupting the well-calibrated trajectory representations; we address this through a gated curriculum fusion strategy. PECT demonstrates improvements in both agent-agent and environment collision rates without sacrificing displacement accuracy, and we introduce the environment collision rate (ECR) metric to capture a failure mode—predicted trajectories passing through walls and static structures—that standard metrics are blind to.

1.4 Contributions and Outline

The remainder of this thesis is organized around the three layers of the pyramid. Each chapter addresses one layer and makes the following contributions:

Chapter 2: Data. We present JaywalkerVR, a VR-based human-in-the-loop system for collecting realistic pedestrian-vehicle interaction data in safety-critical scenarios, and the CARLA-VR dataset collected with it. We validate the realism of VR-collected data through user studies and demonstrate that augmenting training data with VR-collected interactions meaningfully improves forecasting on interactive scenarios.

Chapter 3: Evaluation. We present, to our knowledge, the first comprehensive evaluation of state-of-the-art pedestrian forecasting methods under joint metrics, which require coherent multi-agent predictions rather than allowing independent per-agent sample selection. We reveal a substantial gap between marginal and joint performance and show that optimizing for joint metrics, with no architectural changes, yields significant improvements in both joint prediction quality and collision avoidance.

Chapter 4: Methods. We present PECT (Pose and Environment-Contextualized Transformer), a three-stream architecture that jointly integrates body pose and dense environmental semantics with trajectory history. We introduce the environment collision rate (ECR) metric and a gated curriculum fusion strategy for three-modality integration. PECT improves collision metrics without degrading displacement accuracy.

Chapter 5: Discussion and Conclusions. We synthesize findings across all three layers, discuss cross-cutting themes including the relationship between multi-modal features and the long-tail data problem, and identify open problems including the role of foundation models in the future of trajectory forecasting.

Collectively, these contributions provide a unified framework for understanding and improving trajectory forecasting reliability, offering actionable insights and tools toward building forecasting systems better aligned with the requirements for real-world autonomous decision-making.

Chapter 2

Data: Improving Data Quality and Coverage

This chapter addresses the foundational layer of the thesis pyramid: data quality and coverage. As discussed in the introduction, existing trajectory datasets are dominated by routine scenarios and lack sufficient representation of rare but safety-critical events. Models trained predominantly on common interactions—pedestrians crossing at marked crosswalks, vehicles following lanes—can fail to generalize when confronted with the long-tail distribution of dangerous situations that matter most for safe deployment. Simulated environments can generate such scenarios at scale, but suffer from a fundamental behavioral sim-to-real gap: simulated pedestrians exhibit mechanical, rule-based behaviors rather than genuine human decision-making.

This chapter presents a novel approach that bridges this gap: a Virtual Reality (VR) human-in-the-loop system called JaywalkerVR. By immersing human participants in virtual traffic scenarios through VR headsets, we can safely collect trajectory and body pose data in safety-critical situations that would be dangerous or unethical to recreate in the real world. The key insight is that while the environment is simulated, the human behavior within it is real—participants genuinely respond to virtual traffic as they would to real traffic, producing naturalistic trajectories that capture the complexity of human decision-making. Using this system, we collect the CARLA-VR dataset and demonstrate that augmenting model training with VR-collected interactions improves displacement error by 10.7% and collision rate by 4.9% on interactive scenarios.

The chapter proceeds as follows. Section 2.1 reviews related work on trajectory datasets, synthetic data generation, and VR-based data collection. Section 2.2 presents findings from semi-structured interviews with autonomous vehicle professionals, confirming that the limitations we identify in existing datasets are recognized challenges in the field. Section 2.3 describes the JaywalkerVR system architecture and design. Section 2.4 presents the CARLA-VR dataset collected using this system. Section 2.5 validates the realism of VR-collected data through first-person user studies and third-person evaluations. Section 2.6 demonstrates the impact of this data on trajectory forecasting performance. Finally, Section 2.7 discusses limitations and future directions, and Section 2.8 summarizes the chapter's contributions to the thesis.

2.1 Background and Related Work

This section reviews prior work relevant to data collection for trajectory forecasting, organized into three areas: the landscape of existing trajectory datasets and their limitations, approaches to synthetic data generation, and the use of virtual reality for human behavior studies.

2.1.1 Trajectory Forecasting and Its Data Requirements

Modern trajectory forecasting models are deep, data-driven systems that predict futures for multiple interacting vehicles and pedestrians [75, 76]. Popular architectures from recent years include methods built on deep generative architectures [60, 64, 67, 92, 96, 97, 168, 169, 170, 176, 254], conditional variational autoencoders (CVAEs) [80, 108, 201, 225, 242], hierarchical architectures [25, 36, 125, 127, 128, 134, 228, 232, 239, 246, 252, 253], and transformers [5, 6, 10, 23, 27, 77, 80, 93, 117, 135, 142, 177, 193, 240, 243, 245]. More recently, the field has seen a shift toward autoregressive and language-modeling paradigms: MotionLM [183] casts multi-agent forecasting as next-token prediction over discrete motion vocabularies, and SMART [230] extends this paradigm to scalable real-time generation, ranking first on the Waymo Open Motion Dataset Sim Agents leaderboard. Query-centric architectures such as QCNet [259] and game-theoretic approaches like GameFormer [78] have further advanced interaction-aware prediction, while concurrent self-supervised pretraining strategies such as Forecast-MAE [35] and SmartPretrain [237] have demonstrated that general motion representations can transfer across architectures and datasets. Subsequent to our work, the field has continued to evolve: Bahari et al. [11] propose the first certified trajectory prediction framework (CVPR 2025), adapting randomized smoothing to provide guaranteed robustness against adversarial and noisy inputs, TrajICL [53] introduces in-context learning for pedestrian trajectory prediction without fine-tuning (NeurIPS 2025), and RealTraj [52] combines self-supervised pretraining on synthetic data with weakly-supervised fine-tuning to minimize real-world data collection costs. Despite the variety among architectures, one commonality they all share is reliance on training with ample amounts of good quality data to produce accurate prediction results.

A major challenge for such learning-based prediction systems is the lack of data in complex and dangerous scenes, especially as data-hungry models like Transformers [210] have become the standard. Collecting data from such scenes is challenging from public roads, and public datasets in particular lack such scenarios. Structured data collection, in which human participants carry out long-tail behaviors, can be dangerous—for example, asking children to jaywalk across a busy road is both unsafe and unethical.

2.1.2 Existing Trajectory Datasets and Their Limitations

Public datasets such as nuScenes [19], the Waymo Open Motion Dataset [46], Argoverse [26], and KITTI [55] are commonly used for training and evaluating trajectory prediction models. These datasets are collected in the real world by vehicles driving in public traffic environments. However, they are dominated by commonly-occurring environments and scenes. There is little variety in available scenarios, and there is a particular lack of uncommon environments such as narrow roads or alleyways, and uncommon scenarios such as pedestrian jaywalking, pedestrians walking alongside vehicles on the road, or dangerous or close contacts between pedestrians and vehicles.

Pedestrian-specific datasets such as ETH [154], UCY [110], and Stanford Drone Dataset [171] provide bird's-eye view trajectories but lack vehicle-pedestrian interactions. PedX [91] and PIE [164] contain pedestrian trajectories collected at intersections, which is closer to our use case. However, they too are limited in scope: PedX contains full body pose data but only in limited environments such as crosswalks, and PIE contains only 2D bounding box annotations. The recently released datasets from major autonomous vehicle companies [18, 26, 226] also contain pedestrians, but as we verify through expert interviews in Section 2.2, these datasets lack diverse pedestrian behavior and diverse vehicle-pedestrian interactions. Most vehicle-pedestrian interactions in these datasets occur at intersection crosswalks. Since our work, new efforts have begun to address some of these gaps: Uhlemann et al. [207] introduce a dedicated pedestrian benchmark derived from Argoverse 2 that targets pedestrians in urban traffic environments (WACV 2025), and the Waymo Open Dataset for End-to-End Driving [221] curates 4,021 segments specifically for challenging long-tail scenarios with occurrence frequencies below 0.03%. These efforts focus on curating subsets of existing naturalistic driving data; our contribution is complementary, providing a controlled VR-based collection methodology that elicits realistic human behavior in safety-critical scenarios that cannot be ethically or safely staged on public roads.

2.1.3 Synthetic Data and the Sim-to-Real Gap

One method used to supplement real datasets with more data from uncommon scenes is generating synthetic data using traffic and pedestrian simulators [16, 99, 115, 116]. With simulators, it is possible to generate data in many scenarios at low cost. Controllable simulators can generate large quantities of synthetic data for various scenarios of choice [32, 44, 238], including scenarios that are uncommon in real-life but critical to pedestrian safety.

However, in terms of collecting pedestrian behavior data, most synthetic dataset generation methods use rudimentary autonomous policies [31, 65, 209] to generate pedestrian agent behavior. While some simulators only produce simplistic point trajectory behavior [16, 100], others are extensively customizable and controllable worlds that can model complex sensor data [32, 44, 238]. Although simulators simplify the process of collecting large-scale data in diverse environments, there is a domain gap between synthetic and real data that is often difficult to measure, and many simulators do not reproduce human behavior well. Because AV models ultimately must work for real pedestrians and vehicles, synthetic data falls short in this regard. Since our work, a paradigm shift toward generative world models has begun to narrow the sim-to-real gap: GAIA-2 [172] uses latent diffusion to synthesize controllable, multi-camera driving videos conditioned on ego dynamics, agent configurations, and road semantics, while SimScale [189] demonstrates that co-training on real and neurally-rendered simulated data yields up to 6.8 EPDMS improvement on challenging planning benchmarks. Federated digital twin frameworks such as SV-FDT [233] leverage multi-source surveillance video to build pedestrian-vehicle interaction models that bridge physical and virtual traffic environments. These approaches generate photorealistic sensor data rather than simplified trajectories, but they still rely on learned agent policies rather than real human behavior, which remains the core limitation our VR-based approach addresses.

Some methods solicit input from real pedestrians via data-collection participants using mouse clicks or keyboard controls to control a pedestrian avatar in a virtual environment shown on a display screen [115]. The Garden of Forking Paths [115] collects annotations of trajectory continuations from human annotators to create a multi-future pedestrian trajectory dataset in simulated environments reconstructed from real-world scenes. However, these methods have limitations, as clicks and keyboard controls fall short of the full degree of control pedestrians have over their movements and trajectories during navigation in real urban experiences.

2.1.4 Virtual Reality for Human Behavior Studies

Many works have studied human behavior within virtual environments. Some works study the efficacy and quality of the VR experience via objective measures such as stride length and gait [81] as well as subjective measures such as participants' ratings on presence questionnaires [12, 45, 190, 192]. Some study specific behaviors of interest like normal walking [145], evacuating a building [7], collision-avoidance [13], proxemics and group behavior [146, 158]. Many methods evaluate the efficacy of VR systems by comparing human behavior in VR vs. real environments [141]. Other works evaluate how different conditions of the VR system's features affect user sense of presence and behavior, such as level of photorealism [262], locomotion methods [144, 149, 200], and avatar appearance [148, 179].

Extensive study has been done to characterize pedestrian behavior in traffic environments and responses to vehicle behaviors [29, 37, 49, 82, 109, 222]. Some works study AV behavior or evaluate improvements to the AV-pedestrian communication interface such as intent signaling methods [1, 37, 41, 42, 49, 51, 82, 109, 109]. Many of these works use immersive VR environments to simulate scenarios of interest and study pedestrians' responses [15, 15, 48, 133, 147, 147, 182, 206].

Some works have proposed using scenario simulators with VR headsets to collect pedestrian behavior data more accurate than that found in autonomous simulators, or to study pedestrian responses to vehicle motion. For example, Dosovitskiy et al. [44] and Schmitt et al. [181] created VR simulators where pedestrians are asked to click a button when they decide to cross the street in VR, and Silvera et al. [188] create VR driving simulators to record driver trajectory information. However, these works focus only on verifying pedestrian behavior rather than recording pedestrian trajectory and body pose data. To our knowledge, no prior work had collected data from human trajectories in uncommon but safety-critical scenarios, such as jaywalking, vulnerable road users, or pedestrian behavior in infrequently-seen urban environments. Concurrent and subsequent VR-based datasets have begun to address scale and diversity in other settings: LocoVR [199] (ICLR 2025) provides over 7,000 two-person trajectories captured in VR across 130 indoor home environments with rich social navigation dynamics, and DiVR [198] proposes a cross-modal transformer that integrates static and dynamic VR scene context for trajectory prediction. The PMR dataset [214] (ICLR 2025) uses a mixed reality platform to capture 12,138 sequences of pedestrian interactions across 12 urban settings with multi-view and multi-modal annotations, demonstrating that mixed reality can naturally elicit pedestrian intent including extreme cases. PedGen [261] (ICLR 2025) takes a complementary approach, learning to generate diverse context-aware pedestrian movements from large-scale web videos, and Zheng et al. [255] propose an immersive digital twin framework that uses mixed-reality interfaces to study human-autonomy coexistence in urban transportation. These works validate the broader premise that VR and mixed reality can elicit realistic human trajectories at scale, but focus on indoor navigation or intent classification; our work is distinguished by its use of immersive VR with full-body locomotion to capture naturalistic pedestrian decision-making in outdoor vehicle-pedestrian interaction scenarios that are too dangerous to stage in the real world.

2.2 Understanding Dataset Limitations: Expert Perspectives

Before presenting our VR-based data collection approach, we sought to confirm that the limitations we identified in existing vehicle-pedestrian interaction datasets are recognized challenges in the field. We conducted semi-structured interviews with professionals who had experience working with such datasets. Our goal was to answer the research question: *What* *are the limitations of existing vehicle-pedestrian datasets with respect to the availability and quality of pedestrian data?*

2.2.1 Interview Methodology

We recruited 3 academic researchers and 1 industry practitioner through direct contacts. Two academic participants have performed research in the AV perception, prediction, and planning stack (referred to as I1 and I2). The other academic participant performs research in social navigation for mobile robots that interact closely with humans (I3). The industry participant is a systems test engineer at a large AV company (I4). All participants self-reported that they had experience working with vehicle-pedestrian interaction data.

In our interviews, we first asked participants to describe their area of expertise. Then, we asked them to describe their understanding of the current state of vehicle and pedestrian trajectory datasets: *Which trajectory datasets have you worked with? What do you feel are the current limitations of these datasets?* We probed deeper with questions such as *What kinds of scenarios are lacking in real datasets?* and *What existing methods are there for improving performance in uncommon or out-of-distribution scenarios?* Finally, we asked directly for their opinion about the potential limitations and benefits of data collected in a virtual environment via a VR simulator.

2.2.2 Key Themes from Interviews

Using inductive thematic analysis [17], we grouped the themes that arose during the interviews into four distinct categories.

Lack of Uncommon but Important Scenarios. Vehicle-pedestrian datasets lack scenarios that are important to AV safety but rare in-the-wild or difficult to collect data for. I2 commented that vehicle-pedestrian datasets contain plenty of data of pedestrians crossing roads at intersections, but little data of pedestrians walking on the sidewalk alongside the road. These scenarios are just as important as pedestrians crossing at crosswalks, because pedestrians walking alongside the road could become jaywalkers in the near future.

Lack of Interesting Vehicle-Pedestrian Interactions. Both I1 and I2 pointed out that the most popular public vehicle-pedestrian datasets lack “interesting interactions.” One of the most popular datasets [26, 226] contains a subset compiled specifically for the "interestingness" of its scenarios based on internally-defined heuristics such as number of other agents present, speed changes, and lane changes. However, I1 remarked that these scenarios were still mostly uninteresting, claiming that it is difficult to come up with a straightforward heuristic to separate "interesting" scenes from "boring" scenes. I2 stated that around 75-80% of the data they worked with is "uneventful," and that there was insufficient diversity in scenes. I3 claimed that the pedestrian-only datasets they worked with also lacked variety in environment layouts and did not contain many pedestrian interactions.

Lack of Fine-grained Trajectory Features. Vehicle-pedestrian datasets lack richness in representation. I1 commented that existing trajectory datasets are heavily preprocessed from raw lidar data, losing much of the richness of the raw representation. Popular trajectory datasets such as Argoverse [26, 226] and nuScenes [18] represent pedestrian trajectories as 2D points rather than full 3D human bodies. I2 noted that while pedestrian body orientation is also recorded, it still falls far from the detail available in full body pose. I3 explained that pedestrian bounding boxes often overlap with one another because humans occupy their bounding boxes only sparsely. Overlapping bounding boxes appear to be colliding, which may lead to model training issues such as failure to understand collision-avoidance.

Pedestrian body skeleton pose and head direction can inform understanding of pedestrian intent and high-level motion. For example, pedestrians often look both ways before they cross the street, and a pedestrian looking in the opposite direction of an oncoming car is less likely to stop for that car. As I4 pointed out, their company's AV system has limitations in predicting the behavior of traffic officers, often mis-predicting them as pedestrians crossing the road rather than static pedestrians who will not intercept the vehicle's path.

Summary. The shortcomings of current datasets point to the need for additional data solutions to supplement existing datasets. The VR pedestrian simulator we present in this chapter can be manipulated to recreate diverse and uncommonly seen traffic scenarios, addressing the first two themes. It can also record body pose information, addressing the third theme. Some concerns were raised that data collected from a VR system may have a domain gap with real-world data, just as other simulators do. In Section 2.5, we provide evidence that despite this domain gap, the VR environment still elicits genuine and realistic pedestrian responses.

2.3 JaywalkerVR: A VR System for Safety-Critical Data Collection

Having established the need for better data collection methods, we now present JaywalkerVR, a human-in-the-loop VR pedestrian simulator for autonomous driving that can replicate real pedestrian behaviors and interactions. The system enables efficient, affordable, and safe collection of long-tail pedestrian-vehicle interaction data. This system and the resulting dataset were originally presented in [143].

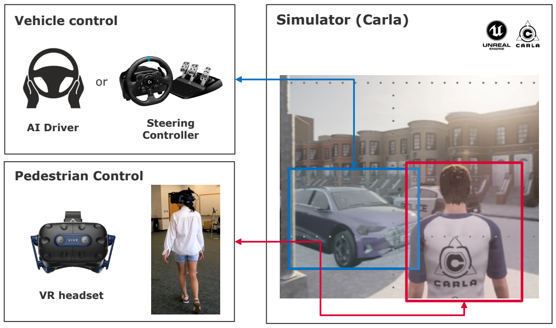


Figure 2.1: JaywalkerVR System Overview. People wearing VR headsets can experience 360-degree immersive simulator images and interact with vehicles and pedestrians in the same way they would in the real world. Simultaneously, the pedestrian avatar moves according to their movement in real world. Vehicles are controlled by CARLA AI agent (automatic control function) or manually using steering controller.

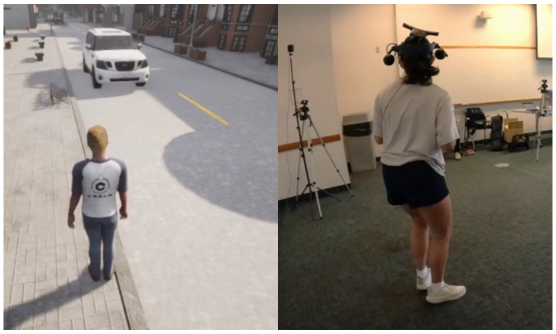


Figure 2.2: Example of JaywalkerVR simulation. Left: Subject's avatar in the JaywalkerVR from a third-person perspective. Right: Captured image of a subject wearing the VR headset in the data collection environment in the real world.

2.3.1 System Architecture

We developed our VR human-in-the-loop pedestrian simulator based on CARLA [21], a popular open-source driving simulator for autonomous driving based on Unreal Engine 4 that includes convenient map and agent assets for defining scenarios.

An overview of our simulator is shown in Figure 2.1. We use a VR headset so that human participants interact with agents as realistically as possible compared to prior work. Since we need annotated interaction data between vehicles and pedestrians, especially pedestrian trajectory and head rotation data, our system simulates the walker avatar's motion according to actual human motion. To synchronize the motion between real human and pedestrian avatars in the simulation world, we use the tracking information from the headset such as 3D location and rotation angle.

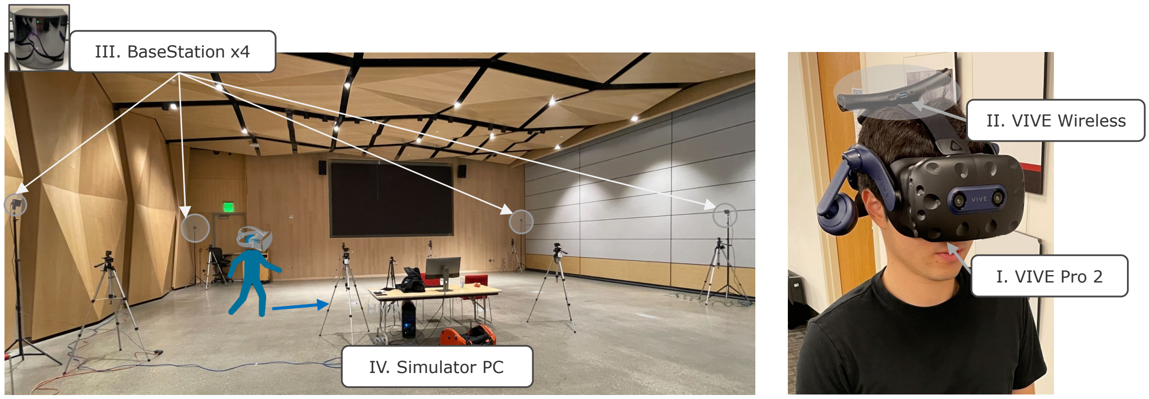


Figure 2.3: Room and equipment setting. (I) VIVE Pro 2: used for 1. visualizing simulator images and 2. tracking human position/rotation in the room (II) VIVE Wireless Adapter: used for allowing VIVE Pro 2 completely wireless (III) BaseStation 2.0: Track VR headset's position/rotation based on lighthouse tracking algorithm (IV) Simulator PC: Execute CARLA based VR simulator.

2.3.2 Walker Control and Tracking

We use the tracking function of the VR headset to control the pedestrian avatar. This function relies on the HTC BaseStation 2.0, an "Outside-In" tracking system which employs a lighthouse tracking method to accurately determine the position of the headset within the tracking range. The official tracking range extends up to approximately 10 meters in both dimensions, as shown in Figure 2.3.

We synchronize the real-world sensor values of the headset and apply them to the entire skeleton mesh to obtain pedestrian positions, then calibrate the headset using the room size and position. Using this information, we use the SteamVR plugin in Unreal Engine to obtain the 3D position $[x, y, z]$ of the VR headset and use it to control the position of the pedestrian skeletal mesh in CARLA. In each scenario, we synchronize the pre-defined start position of the pedestrian avatar with the standing position of the human subject, and control the skeletal mesh model to follow the real human's movement. We use the headset's yaw angle to adjust the yaw angle of the whole skeleton mesh. Finally, we update the pedestrian's movement animation to match their actual walking speed, enabling a person wearing a VR headset to control and move the avatar freely within the experimental setup.

2.3.3 Pedestrian Model

The walker skeleton model is provided in CARLA by default, and the movement of this skeleton model can be controlled by keyboard or joystick input devices. However, there are no native functions that control the skeleton model according to the movement of a VR headset. We modified the walker blueprint to control the skeleton model by synchronizing it with the motion of the VR headset. The virtual camera module is attached to the walker's head, and the walker's blueprint is modified to provide a first-person feel. The camera module acts as the avatar's virtual eyes, and the skeletal mesh defines the walker's appearance.

In addition, we developed an inverse kinematics (IK) setup for the representation of walking animation. The skeleton model is designed to make walking motions in response to the movement speed of the VR headset.

2.3.4 Scenario Generation and Data Recording

To define arbitrary scenarios for data collection, we implemented a scenario generation function using the CARLA Python API, in particular, the TrafficManager components. First, the CARLA AI Agent, which is the driving policy for autopilot implemented in the CARLA standard, was used to control the vehicle agent of CARLA, and traffic flow was generated after the Autopilot function was enabled in each spawned vehicle. In terms of route planning, desired routes automatically run according to the route plan determined by the AI Agent by creating a route plan in which vehicle spawn points are arranged. The behavior of the AI Agent uses the default setting and stops when a pedestrian is detected. Each agent's data, such as position and size, are collected at 20 Hz.

2.3.5 Hardware Configuration

We used the HTC Vive Pro 2 VR headset which has SteamVR support. We used four HTC BaseStation 2.0 units for tracking the headset. We also installed the VIVE Wireless adapter, allowing the headset to be used completely wirelessly. We used a desktop PC containing a PCI express slot to install the image emitter module of the VIVE Wireless adapter for the simulator, with an Intel core i9-12900KF CPU, NVIDIA GeForce RTX 3080 GPU, and 64GB RAM.

Since VIVE Wireless is only supported by Windows 10 or 11, we set up our CARLA-based VR pedestrian simulator on a Windows 11 desktop PC. We used Unreal Engine UE 4.26.2 and CARLA 0.9.13.

For the validation studies described in Section 2.5, the data collection environment required a rectangular collection space free of obstacles measuring approximately 40' × 20' in dimensions. Four HTC VIVE base stations were set up in the four corners of this space. Eight GoPro Hero 10 cameras were set up along the two long edges of the space to record the subject from different angles so that their skeleton pose could be extracted. A schematic of the VR system components is shown in Figure 2.4, and the actual data collection space with system setup is shown in Figure 2.5.

2.4 The CARLA-VR Dataset

Using the JaywalkerVR system, we collected a high-quality vehicle-pedestrian interaction dataset called CARLA-VR. This dataset addresses the lack of long-tail data samples in commonly used real-world autonomous driving datasets.

2.4.1 Safety-Critical Scenarios

We defined four pedestrian-vehicle interaction scenarios in CARLA for collecting VR human data, as shown in Figure 2.6.

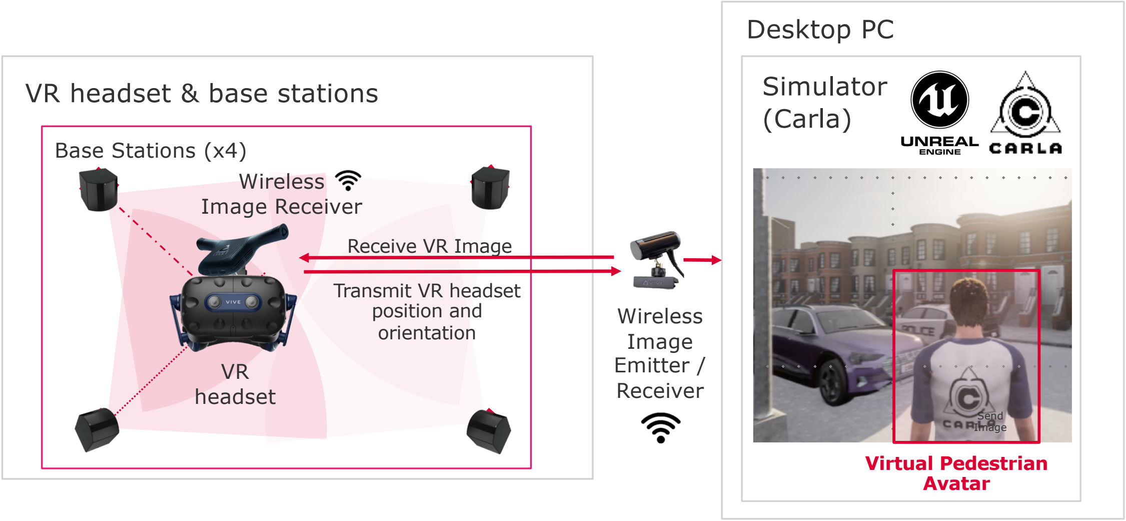

VR headset & base stations
Base Stations (x4)
Wireless
Image Receiver
Receive VR Image
Transmit VR headset position and orientation
VR headset
Wireless Image Emitter / Receiver
Desktop PC
Simulator (Carla)
UNREAL ENGINE
CARLA
Send Image
Virtual Pedestrian Avatar

Figure 2.4: Schematic of the VR data collection system.

Jaywalk. Pedestrians jaywalk across a road while yielding to vehicles coming from both directions on a two-lane road. In this scenario, we expect participants to try to interact with oncoming vehicles, such as yielding to vehicles, and to cross the street on their own timing and with their own decision-making. For example, some participants behave aggressively, but others behave nervously and miss the opportunity to walk. We obtain a variety of behaviors from each subject, such as different speeds of walking and different timings of crossing.

Parked Cars. Pedestrians walk along the edge of the road, avoiding parked vehicles and moving to a position one car ahead while paying attention to vehicles approaching from behind. In this scenario, we also expect participants to start walking on their own timing.

4-Way Stop. Pedestrians cross the crosswalk while paying attention to cars coming from four directions at a four-way stop. In this scenario, we expect participants to cross the crosswalk at various times as decided by each of them, responding to vehicles coming from different directions.

Parking Lot Entrance. Pedestrians walk through the entrance to a parking lot while paying attention to and avoiding any entering and exiting vehicles. In this scenario, we

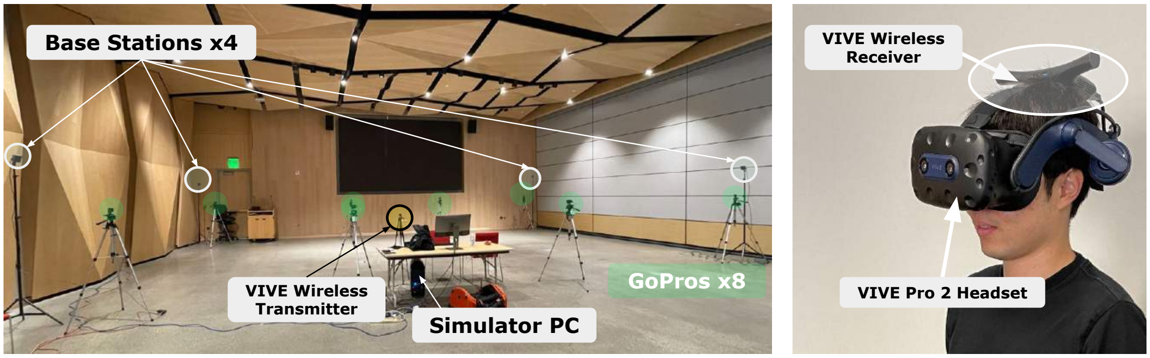


Figure 2.5: Left: The VR system set up in the classroom data collection space. Right: A pedestrian subject wearing the VR headset.

expect participants to behave by yielding or not yielding to vehicles at various decision points.

2.4.2 Dataset Statistics

We collected data from 80 participants in each of the four scenarios. In the Jaywalk, Parked Cars, and 4-Way Stop scenarios, the surrounding vehicles are controlled by a CARLA AI agent in completely autonomous driving mode. In the Parking Lot Entrance scenario, the vehicles are controlled by a human driver using a steering controller, as CARLA did not support implementing a route plan for the vehicle to enter and exit the parking lot.

We collected a total of 572 scenes comprising 12,702 frames. The data contains position $[x, y, z]$ (m), rotation $[\theta, \phi, \psi]$ (deg), velocity $[v_x, v_y, v_z]$ (m/s), acceleration $[a_x, a_y, a_z]$ (m/s^2) in global coordinates in CARLA's map, object type (car, pedestrian) and object shape information (length, width, height). Each scene data is between 10 and 30 seconds long and was recorded at 20Hz. Note that the "body pose" captured by CARLA-VR refers to full 6-DoF rigid body pose (3D position and 3D rotation) derived from VR headset tracking, plus the derived velocity and acceleration quantities listed above. This is distinct from the full-body skeletal keypoint annotations used in Chapter 4, which are obtained from the JRDB dataset's multi-sensor pose estimation pipeline. However, because our data collection environment includes 8 surround GoPro cameras, full-body skeletal keypoints could in principle be extracted from the CARLA-VR recordings using multi-view pose estimation methods such as Harmony4D [89]; we discuss how this could strengthen the realism validation in Section 2.7.

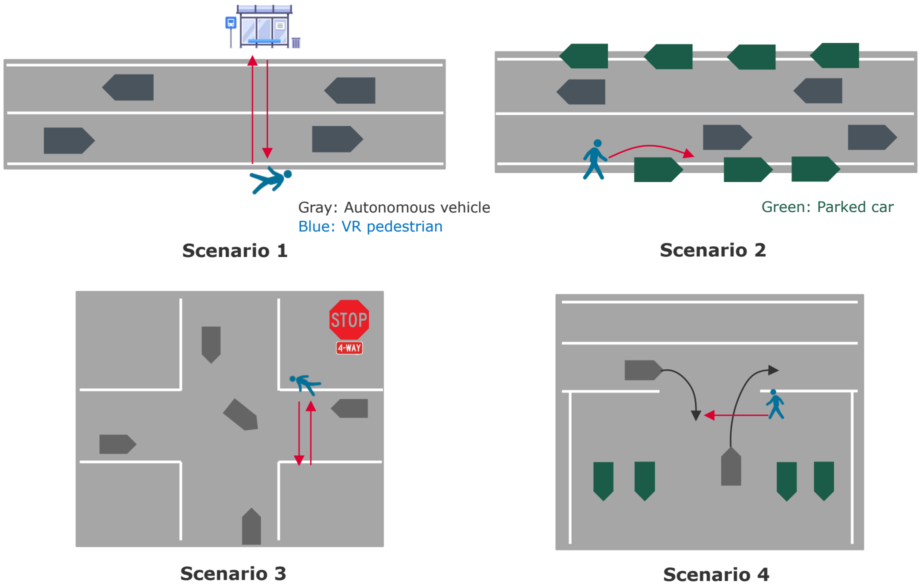


Figure 2.6: Experimental scenarios. (1) **Jaywalk:** Pedestrians jaywalk toward a bus stop while avoiding vehicles. (2) **Parked Cars:** Pedestrians avoid parked vehicles and move along the road. (3) **4-Way Stop:** Pedestrians cross the crosswalk, avoiding cars at a four-way stop. (4) **Parking Lot Entrance:** Pedestrians walk through the entrance of a parking lot paying attention to cars.

2.5 Validating VR Data Realism

A critical question for VR-based data collection is whether the collected data genuinely reflects real human behavior, or whether the virtual environment introduces artifacts that limit its utility. We address this question through a comprehensive validation study with two components: a first-person user study evaluating sense of presence and behavioral similarity, and a third-person evaluation comparing VR trajectories to real-world and synthetic trajectories. This validation study was originally presented in [224].

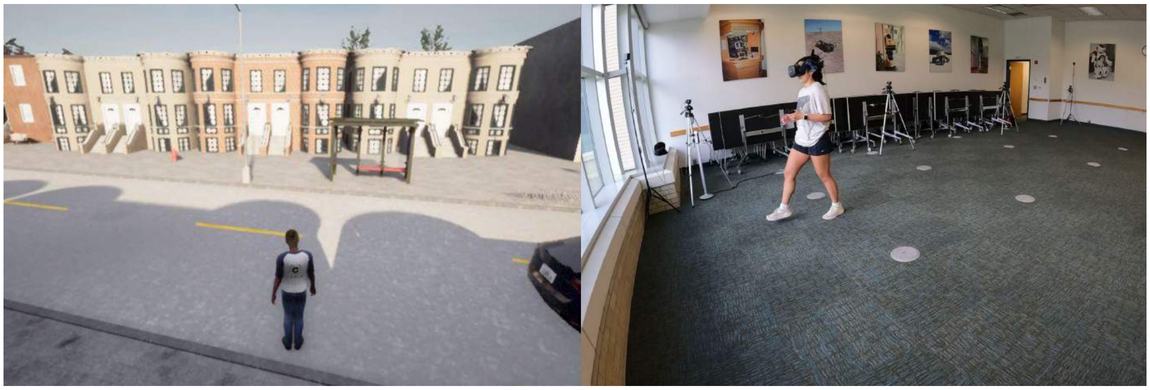


Figure 2.7: Left: Participant's avatar within the virtual environment from a third-person perspective. Right: participant wearing the VR headset in the data collection environment.

2.5.1 First-Person User Study: Sense of Presence and Behavioral Similarity

Study Design

We recruited participants by word-of-mouth, university email lists, and physical flyers posted around the university campus. Participants with physical disabilities, vulnerable individuals, and minors were excluded from participation. Participants came to the university campus to participate in the study, where we set up the data collection environment in a large, empty classroom.

After signing an informed consent form, participants were asked to put on the VR headset and familiarize themselves with the VR headset and virtual environment by walking around in a version of the simulation without moving vehicles. Then, we asked participants to complete 3 tasks, each featuring a different traffic environment in which they had to walk to reach a goal destination while in close interaction with moving vehicles: 1) jaywalking across a two-way street, 2) walking alongside moving vehicles on the road, and 3) crossing a crosswalk at a 4-way intersection with stop signs. The 3 tasks are depicted in Figure 2.8. To inform participants of their goal destinations, we placed colored square markers on the ground within the virtual environment, and used commands such as "Do you see the colored square on the ground on the other side of the road? Please walk to it" to direct the user. After completing all 3 tasks, participants were asked to complete a presence questionnaire to evaluate their experience.

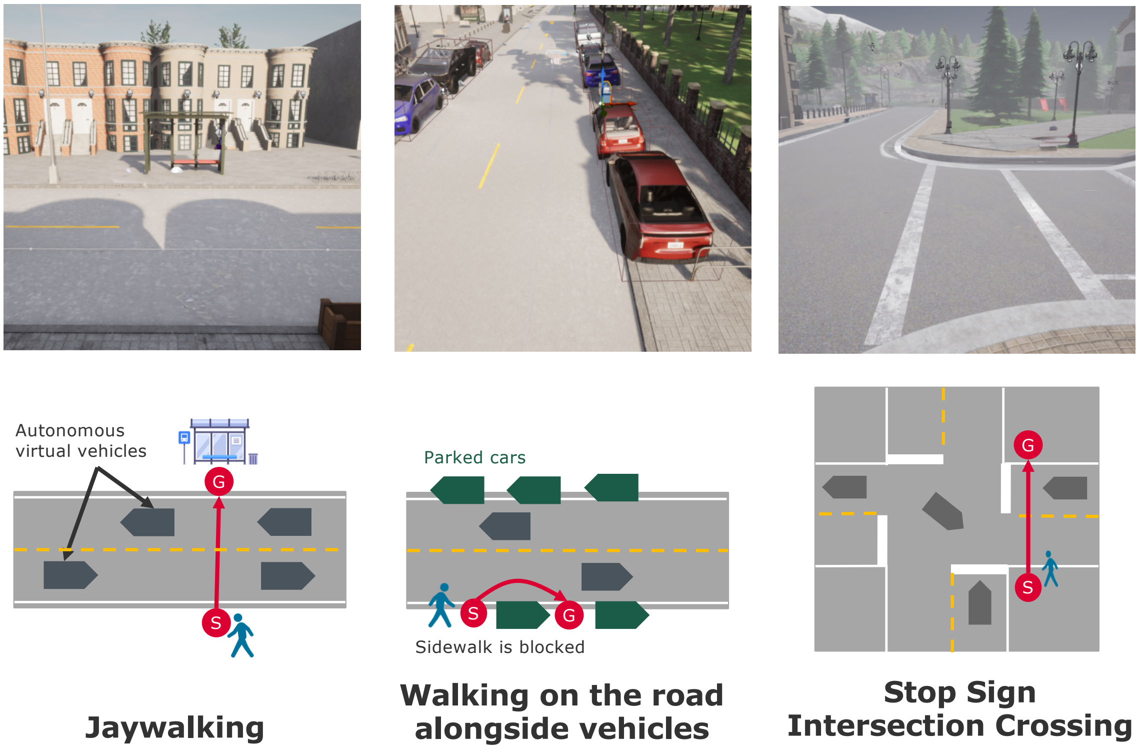


Figure 2.8: The three tasks that we ask our study participants to complete. Top: snapshot of each task as seen in the CARLA virtual environment. Bottom: schematic of each task; “S” denotes participant start location and “G” denotes goal.

Self-Reported Measures

To design the questions asked in the post-experience questionnaire, we used a combination of questions derived from previous works, questions modified from previous works, and custom questions designed specifically to evoke defining attributes of the VR user experience. We designed 12 quantitative questions using a semantic differential scale [151] from 1 to 7, as well as 4 free-response questions with free text entry. We grouped the questions into 3 categories reflecting different aspects of trustworthiness of the data collection system: sense of presence experienced within the virtual environment [12, 227] (coded by the letter P), sense of agency (A), and behavioral and experiential similarity to real-life (B). In total, the post-condition questionnaire was composed of 16 questions aimed to cover a range of subjective ratings while keeping the time for participants to complete the questionnaire

ID	Source	Category	Question	Semantic Differential Anchors (1→ 7)
P1	SUS [208]	presence	Please rate your sense of "being there" in the environment.	did not have a sense of "being there" → normal experience of "being there"
P2	SUS [208]	presence	When you think back on your experience, do you think of the environment more as images that you saw, or more as somewhere that you visited?	Images that I saw → somewhere that I visited
P3	WS [227]	presence	How real did the objects in the environment seem?	very fake, clearly images → very real, like I could touch them
P4	*custom*	presence	During the VR experience, were you more concerned with the real world (this classroom) or the virtual world?	real world → virtual world
P5	SUS [208]	presence	When you think back on your experience, do you think of the vehicles more as images that you saw, or as things you interacted with?	Images that I saw → things that I interacted with
P6	*custom*	presence	During the experience, did you often think to yourself that the vehicles were physical objects that could have actually hit you and caused you injury?	very much so → not at all
A1	*custom*	agency	How comfortable did you feel moving around in the environment?	very uncomfortable → very comfortable
A2	AE [59]	agency	How much did you feel like you could control the virtual body?	did not feel much agency → could control it like own body
A3	SPES [69]	agency	How freely did you feel you could move in the environment?	movements were restricted → movements were free
B3	NASA-TLX [68]	behavior	How mentally demanding were the tasks compared to doing them in the real world (e.g. on a real street)?	less demanding → more demanding
B1	*custom*	behavior	Did you feel your head, arm, and body movements were the same as they would have been in the real world?	completely different → exactly identical
BF1	*custom*	behavior	What parts of your movements were different than how they would have been in the real world?	*free response*
B2	*custom*	behavior	Did you feel your decisions about when to act were the same as they would have been in the real world?	completely different → exactly identical
BF2	*custom*	behavior	What parts of your decisions were different than how they would have been in the real world?	*free response*
PF1	*custom*	presence	What aspects of the systems or environment were realistic?	*free response*
PF2	*custom*	presence	What aspects of the system or environment were unrealistic?	*free response*

Table 2.1: Post-condition questionnaire used in the study.

within 10 minutes. Questions, sources, categorizations, and semantic differential anchors are recorded in Table 2.1.

Participants also reported various demographics: age, gender, VR experience (5-point Likert-type scale), video game frequency (5-point Likert-type scale), frequency of jaywalking behavior (5-point Likert-type scale), and level of alertness (Stanford Sleepiness Scale [185]). Summary of participant demographics is reported in Table 2.2.

Quantitative Results

A total of N=63 participants participated in our study. Aggregating all responses across all questions in a category, users rated their sense of presence (P) 5.6, sense of agency (A) 5.5, and similarity of their behavior and experience to that of the real world (B) 5.3. These

Variable	N=64
Age {18-37} (years)	24.71 (4.17)
% Male	63% (n=39)
% Female	37% (n=23)
Video gaming frequency {1-7}	3 {1-5}
VR experience {1-7}	2 {1-4}
Jaywalking frequency {1-7}	3 {1-5}
Level of alertness {0-7} [185]	6 {3-7}

Table 2.2: Summary of participant demographics. Continuous variables are summarized as *mean* (*standard deviation*) and ordinal variables are summarized as *median* {*range*}.

Question Category	**Presence**						**Agency**			**Behav. Sim.**		
ID (as in Table 2.1)	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**	**A1**	**A2**	**A3**	**B1**	**B2**	**B3**
Rating	5.9	5.3	5.0	6.2	5.9	5.3	5.1	5.6	5.9	5.0	5.3	4.8
Average	**5.6**						**5.5**			**5.3**		

Table 2.3: VR Presence Questionnaire Evaluation Results.

numbers suggest that users experience a relatively moderate-high sense of presence, agency, and behavior similarity to the real world.

The users' aggregate response of 5.3 in the behavioral similarity category is slightly lower than that in the sense of presence and sense of agency categories, possibly suggesting that high sense of presence is not sufficient to guarantee that a pedestrian behaves exactly the same in VR as they do in the real world. One possible explanation is that, for the high proportion of users who rarely used VR, the novelty factor of the VR environment causes them to behave differently, even though they are experiencing a high degree of presence and immersion.

We report the average user ratings for each question in Table 2.3.

To assess the internal consistency of each subscale, we computed Cronbach's α across the N=63 participants with complete responses. Note that B3 (mental demand) is reverse-coded relative to B1 and B2, since higher scores on B3 indicate *greater* demand and

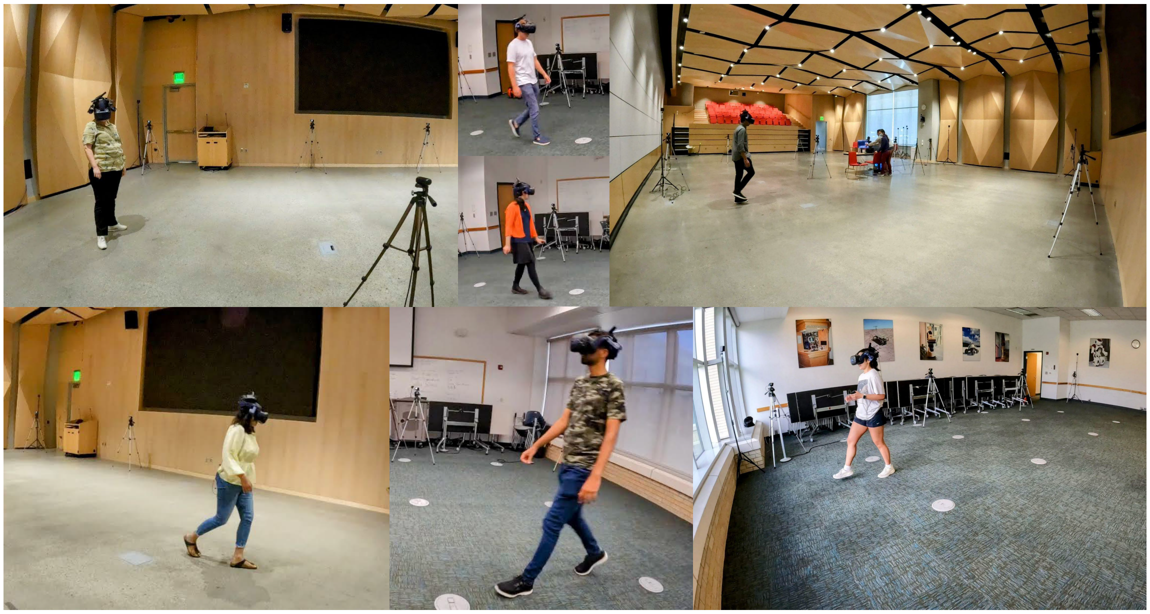


Figure 2.9: Various study participants wearing the VR headset and walking around in two different classroom-setting data collection environments.

thus *less* similarity to real-world behavior. The Presence subscale ($\alpha = 0.65$, 6 items) falls in the "questionable" range per standard interpretation thresholds [56], which is not uncommon for short subscales that combine items from multiple validated instruments (SUS [208], WS [227]) with custom questions. The Agency subscale ($\alpha = 0.41$, 3 items) and Behavioral Similarity subscale ($\alpha = 0.56$, 3 items with B3 reverse-coded) exhibit lower internal consistency, reflecting both the small number of items per subscale and the fact that each item targets a somewhat distinct facet of the construct (e.g., comfort vs. control vs. freedom of movement for Agency). These moderate-to-low α values suggest that the subscale scores should be interpreted with caution: while the individual item ratings provide useful signal about specific aspects of the VR experience, the category-level averages aggregate items that are not strongly interchangeable measures of a single latent construct. This is a known limitation of short, ad-hoc questionnaires assembled from heterogeneous sources [180], and future work could improve reliability by expanding each subscale with additional items drawn from established instruments.

Qualitative Results

The qualitative responses evoke greater detail from the participants' experiences. In response to the free-response question, *What aspects of the systems or environment were realistic?*, participant responses fell into the following major categories:

1. Traffic patterns and flow were very authentic.
2. The visual rendering contributed to a realistic experience: the relative sizes and dimensions of cars, buildings, roads, sidewalks, trees and other aspects of the environment felt accurate, appropriately sized, and well-rendered. The environment felt realistically designed. Some users commented that the curb appeared so realistic that it caused them to stumble when they actually tried to step onto it, as no physical curb existed in real-life.
3. Movement of images along the line of vision was smooth when the user moved their head or body.
4. Cars moved and behaved realistically with respect to speed, positioning, and timing.
5. The environment elicited emotions and caution similar to real-world experiences, such as feeling threatened by cars and being conscious of making mistakes, like crossing the road at the wrong time.

In response to the question, *What parts of the system or environment were unrealistic?*, the main concerns brought up by participants include:

1. Dizziness and nausea while using the system
2. System stability issues: lag, flicker, and instances in which the virtual space would re-calibrate and the participant would be transported within the virtual environment to a different location despite not having moved
3. Awareness of the real world and obstacles in the real world, which took away from sense of presence
4. Lack of peripheral vision due to the construction of the VR headset goggles, which resulted in participants turning more to the left and right to check for vehicles than they do in real life
5. Lack of environmental sound

6. Imperfect visual information: lack of stop signs and walk signals at intersections, game-asset quality of buildings and surroundings, not all body parts visible in the simulation
7. Imperfect tactile information: sidewalk curb observed in virtual environment, but real-world environment lacked a tactile height change
8. Mechanical and unnatural movement of vehicles, which did not deviate from fixed paths nor yielded to pedestrians
9. Lack of drivers in vehicles, which rendered users unable to make eye contact with drivers to determine when to cross
10. Lack of other pedestrians in the environment
11. Being told where and when to start and stop, unlike in real life where this is self-determined

In response to the question, *What parts of your movements were different than how they would have been in the real world?*, participant responses group into the following major concerns:

1. Did not turn head as much due to weight of the VR headset and limited peripheral vision
2. Step movements differed due to difference between visual perception and tactile perception (such as the curb)
3. Walking style: keeping hands in front of body as a defensive body posture to avoid bumping into walls in the real world, or to fend off aggressive vehicles in the virtual world
4. Navigation around static objects in the virtual world such as parked cars differed due to the unexpected dynamics of those objects, such as moving aside when users collided with them

In response to the question, *What parts of your decisions were different than how they would have been in the real world?*, participants reported the following major themes:

1. Some participants were more hesitant to move in the virtual environment than in real life, due to the realism limitations discussed above.

2. Other participants had an added sense of urgency to act, because of unexpected or erratic vehicle behavior, or because lack of peripheral vision limited the participants' awareness of cars
3. Some participants exhibited more risky behavior due to less fear about being hit by a virtual car.

The qualitative responses affirm that, while the system does have areas for improvement, there are noteworthy strengths. Participants largely found the visual aspects and their own emotional responses to be authentic to their real-world experience. Some users did report aspects of the virtual environment that seemed unrealistic; yet sense of presence, agency, and behavioral similarity to real-life still seems high (Table 2.3). This suggests that complete realism of the virtual environment may not be necessary to achieve high sense of presence.

Furthermore, for users that found their behavior or movements differing from that in real life, the different behaviors are not necessarily less genuine than those of real life. Uncommon, edge-case scenarios sometimes do not feel real because they are unexpected, but they are no less important for ensuring the safety of AVs. The VR system is exactly designed to evoke pedestrian response in those edge-case scenarios, thus improving the collection of data in those scenarios.

2.5.2 Third-Person Evaluation: Comparing VR, Real-Life, and Synthetic Trajectories

To further substantiate the comparability of real-life and VR data, we performed an additional third-person evaluation study. In this study, we asked external evaluators to try to distinguish between trajectory data collected in real-life vs. VR. This study evokes a different facet of trajectory realism, one that comes not from first-hand experience, but from third-person evaluation.

Study Design

To collect real-life jaywalking trajectories, we visited a two-way single-lane street near the university that is well-trafficked by jaywalkers as well as vehicles. We chose this setting for its potential to provide real-life trajectories in a similar geographic layout to that we used in the VR jaywalking scenario. In contrast to the VR scenario, there were fewer vehicles as well as more irregular gaps between vehicles, making it easier for jaywalkers to find a gap in which to cross. A DJI Mavic Mini drone was used for data collection, flown at a height of approximately 25 meters. In a 1-hour time interval, we secured footage for 7 jaywalkers.

The trajectories of these jaywalkers were manually annotated via a trajectory annotation tool and interpolated to 20 frames per second to match the frame rate of the trajectory data collected from the VR system. The resulting trajectories were smoothed via Gaussian smoothing with a standard deviation of 0.35 seconds, as the VR system also performs some smoothing to eliminate noise from the collected trajectories.

The trajectories were rendered as animated gif images, in which the pedestrian was depicted as a small red circle, and vehicles as blue rectangles. A random rotation was added to each image to eliminate visual bias due to the layout of the scene. The spatial limits of the animation were set such that users had an ample field of view to see oncoming vehicles from both sides (which is important for judging pedestrian behavior). All animations were set to have a similar field of view. For the real-life trajectories, only the single jaywalking pedestrian was annotated. Scenes were chosen such that the pedestrians kept an ample distance from other pedestrians such that they did not visibly interact or influence each other's motion; *i.e.*, the vehicle-pedestrian interaction would be the sole interaction evaluated. An example schematic in the style of the animations is shown in Figure 2.10.

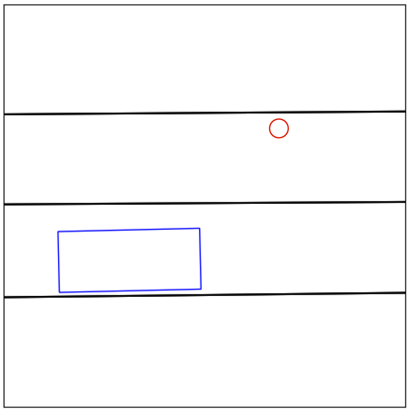


Figure 2.10: A frame from an example schematic shown to survey respondents.

The survey consisted of 3 forced-choice questions featuring a comparison between either a real-life trajectory and a VR-collected trajectory, a real-life trajectory and a synthetic piecewise-linear trajectory, or a VR-collected trajectory and a synthetic piecewise-linear trajectory. Next to each pair, we prompted respondents with the question, *Which one looks more real?* The synthetic piecewise-linear trajectory was included as a control; the expectation is that users would more easily perceive that trajectory as "fake." Pairings were selected at random, choosing one animation from 7 real-life trajectories, one from

8 VR trajectories, and one from 2 synthetic trajectories. With 10% probability, the VR trajectory was swapped out for the synthetic trajectory, and with 10% probability, the real-life trajectory was swapped out for the synthetic trajectory; thus, ∼20% of pairings featured a synthetic trajectory in the comparison. The other ∼80% of pairings featured a comparison between real-life and VR trajectory. Display order was randomized. Finally, there was an optional free-response question at the end: *What characteristics did you use to distinguish between real and fake?*

The survey was distributed via word-of-mouth as well as by university mailing list. Response time for the survey was around 3 minutes per person.

Results

A total of 302 respondents answered the survey for a total of $302 \times 3 = 906$ forced-choice comparisons made by survey respondents.

Consistent with expectations about the "control" variable, evaluators could easily tell the difference between a simple synthetic policy and a genuine human trajectory. 89.3% of real-life / synthetic comparisons were evaluated in favor of the real-life trajectory looking "more real" than the synthetic. The response was similarly high for the VR / synthetic comparisons, with 88.9% evaluated in favor of the virtual over the synthetic (Table 2.4).

Also consistent with expectations, the real-life trajectories were the most-frequently chosen as "more real," with approximately 67% of responses choosing it as more "real" in a pair (Table 2.4). However, VR trajectories do not lag far behind; 59% of responses chose the VR trajectories as "more real" in a pair (Table 2.4). Of particular note is that 36.5% of real-life / VR comparisons were evaluated in favor of the VR trajectory looking "more real" (Table 2.5); these responses claimed that the VR trajectory looked "more real" than a real trajectory.

Though 36.5% is still below 50% (which would mean that real-life trajectories are indistinguishable from VR trajectories), this number still substantiates the claim that VR trajectories can evoke genuine pedestrian responses. First, the results confirm that real-life trajectories are less distinguishable from VR trajectories than they are from fully-synthetic trajectories, supporting the claim that the VR system evokes more natural pedestrian responses than simple policies used by synthetic pedestrian simulators [16, 100]. Second, there could be other reasons evaluators are able to tell the difference between real and VR

Truth Value → Guessed value ↓	Real-life	VR	Synthetic
Real	67.0%	58.6%	10.9%
Fake	33.0%	41.4%	89.1%

(a) Percentages

Truth Value → Guessed value ↓	Real-life	VR	Synthetic
Real	547	482	19
Fake	269	340	155
Total	816	822	174

(b) Counts

Table 2.4: Confusion matrix for survey respondent guesses (real / fake) vs. truth category (real-life, VR, synthetic)

Pairing	Real-life / VR	Real-life / Synthetic	VR / Synthetic
% Correct (N)	64.5% (732)	89.3% (84)	88.9% (90)

Table 2.5: Percent correct of each pairing. “Correct” is defined as selecting real as more “real” when compared with either VR or synthetic, and selecting VR as more “real” when compared to synthetic.

trajectories that have nothing to do with how “real” VR trajectories are. For example, one respondent who evaluated 3 comparisons between real-life and VR trajectories and marked all the real-life ones as more “real” gave the free response answer that “the fake pedestrians seem to cross dangerously close to the vehicles, and sometimes stop in the middle of the road...” Though this respondent labeled all real-life trajectories as “more real” than the corresponding VR ones, their explanation reveals an important insight: the VR environment enabled collection of more aggressive, risk-taking behaviors that are rare but do occur in real life. Sometimes there are jaywalkers who jaywalk even on heavily-trafficked roads. These jaywalkers, not able to find a gap between vehicles, must stop in the middle of the road before they can cross completely. Though rare, “dangerous” jaywalkers do exist in real-life, and it is important that safety-critical AV systems have their data.

2.6 Impact on Trajectory Forecasting Performance

The validation studies above suggest that VR-collected data captures meaningfully realistic pedestrian behavior, though with acknowledged limitations—presence scores indicate moderate-high rather than complete immersion, and third-person evaluators could distinguish VR from real-life trajectories more often than not. The critical test, however, is whether this data improves downstream model performance—a pragmatic measure of utility that complements the perceptual realism assessments. We now demonstrate this practical impact on trajectory forecasting models. This section shows that augmenting training data with CARLA-VR improves model performance, particularly in interactive scenarios—providing direct evidence for the thesis argument that better data coverage leads to more robust generalization.

2.6.1 Experimental Setup

Baseline Model. We use AgentFormer [245] in all experiments for measuring trajectory forecasting performance. AgentFormer is a Transformer-based model that jointly models the time and social dimensions with an agent-aware attention mechanism. The model leverages a sequence representation of multi-agent trajectories by flattening trajectory features across time and agents and using the resulting spatiotemporal attention-based features for trajectory prediction. More details, such as the model architecture and training setup, are available in the original paper. In our experiments, we generate 10 sample 2D trajectories for each agent by using past trajectories, yaw angle information, and a semantic segmentation image of a bird's eye view obtained from CARLA as inputs.

Datasets. We use the following datasets in our experiments:

nuScenes: nuScenes is a widely used public autonomous driving dataset with annotated data, such as position in global coordinates in nuScenes's map, rotation, and bounding box size at 2Hz. nuScenes also provides HD semantic maps with 11 semantic classes.

nuScenes-prediction: We extract the nuScenes prediction dataset from annotated data for the nuScenes prediction challenge. This is used for pre-training of the trajectory prediction model and also for evaluation of prediction performance in general scenes.

nuScenes-interaction: To check the prediction model's performance in rare scenes, we extract interactive scenes from similar situations to our simulation scenarios (e.g., jaywalking) from annotated data on the nuScenes dataset, following the filtering method of [87]. Since this dataset contains only vehicle-pedestrian interaction data that actually occurred in the real world, testing the prediction model with this dataset allows us to evaluate the model's performance in real-world interactive scenes.

CARLA-VR dataset: Our collected dataset containing rare vehicle-pedestrian interactive scene data from the VR simulator described in Section 2.3. We use it for additional training of the trajectory prediction model and also for evaluation of prediction performance in interactive scenes. To align the sampling rate, the CARLA-VR dataset is resampled from 20Hz to 2Hz.

Model Variants and Evaluation Metrics. Our baseline is state-of-the-art AgentFormer trained on the nuScenes prediction dataset, denoted **AgentFormer-B**. To demonstrate the utility of our proposed dataset, we further train AgentFormer-B on CARLA-VR to get **AgentFormer-VR**. We then evaluate both models' performance on nuScenes-prediction, CARLA-VR, and nuScenes-interaction.

We use the following metrics to measure performance:

Marginal *ADE* / *FDE*: *ADE* / *FDE* encompasses Marginal Average Displacement Error (*ADE*) and Marginal Final Displacement Error (*FDE*), commonly used for evaluating how close the predicted trajectory is to the ground truth trajectory. Since AgentFormer generates 10 sample trajectory sets, we evaluate min*ADE* / *FDE*, the top-K minimum error.

Joint *ADE* / *FDE*: Unlike *ADE* / *FDE*, Joint *ADE* / *FDE* (*JADE* / *JFDE*) evaluates scene-level *ADE* / *FDE* [223]. Since this metric calculates the average error over all agents within a sample before selecting the best one, we cannot mix-and-match agents between different samples. This means we can evaluate how close the prediction result is to ground truth while considering social-interaction at the scene-level. We provide the full formal definition and motivation for joint metrics in Chapter 3.

Collision Rate: Collision Rate (CR) evaluates whether the predicted trajectories of each agent collide with each other within the same prediction timestep. (In Chapter 4, we rename this metric to Agent-Agent Collision Rate (ACR) to distinguish it from the environment collision rate introduced there.)

Table 2.6: Evaluation Results: Impact of CARLA-VR Data on Trajectory Forecasting

Test Dataset	Model	Marginal XDE [m] (K=10)		Joint XDE [m] (K=10)		Collision rate
		ADE↓	FDE↓	JADE↓	JFDE↓	CR mean↓
nuScenes-prediction	AgentFormer-B*	**1.2299**	**2.7175**	**2.4023**	**5.9062**	0.1275
	AgentFormer-VR**	1.4408	3.1088	2.7020	6.5288	**0.1186**
CARLA-VR	AgentFormer-B*	1.1404	2.7243	1.9274	5.1474	0.3266
	AgentFormer-VR**	**0.9319**	**2.1491**	**1.6193**	**4.1201**	**0.2856**
nuScenes-interaction	AgentFormer-B*	1.2712	2.8285	2.5995	6.4676	0.3170
	AgentFormer-VR**	**1.1349**	**2.4637**	**2.2680**	**5.3770**	**0.3016**

∗ AgentFormer trained on the nuScenes prediction dataset only

∗∗ AgentFormer trained on the nuScenes prediction dataset and the CARLA-VR dataset

2.6.2 Results

The results of the experiments are listed in Table 2.6. In terms of the evaluation on CARLA-VR dataset and nuScenes-interaction dataset, all metrics improve when incorporating our CARLA-VR dataset. Marginal *ADE* / *FDE* performance improves by 10.7–12.8%, and Joint *ADE* / *FDE* also improves by 12.6–16.9%. Further, the most important metric for safety, collision rate, improves by 4.9%.

In Figure 2.11, we show predicted trajectories from AgentFormer-B (left) and AgentFormer-VR (right). The ground truth trajectories are drawn in red, and the best predicted trajectories are shown with time-varying color. We find that AgentFormer-B, only trained on nuScenes prediction dataset, often predicts trajectories for pedestrians that lead them into direct collision with vehicles. We attribute this to the rarity of dangerous pedestrian-vehicle interactions in the real-world nuScenes dataset. On the other hand, when AgentFormer leverages our safety-critical interaction dataset, we see in the right figure that the pedestrian is predicted to *yield* to the incoming vehicle, better matching the ground truth trajectory. These qualitative visualizations corroborate our quantitative results that the proposed CARLA-VR dataset, containing safety-critical pedestrian-vehicle interactions, better enables trajectory prediction models to model agent behavior in dangerous and rare scenarios.

2.6.3 Discussion

Our results show that the prediction model becomes more robust in real-world interactive scenes through fine-tuning on the CARLA-VR dataset. In particular, min*JADE* / *JFDE* and CR decrease substantially for nuScenes-interaction—the most safety-critical and difficult

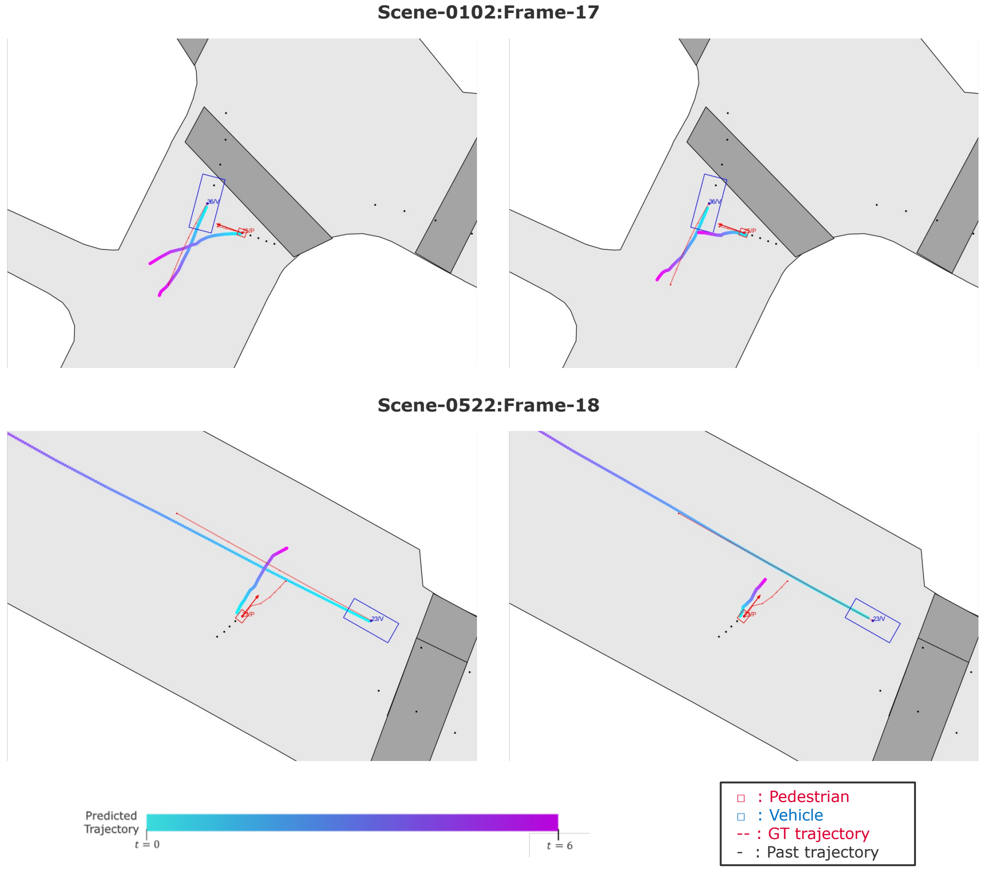


Figure 2.11: Visualization result. Left: predicted trajectories of AgentFormer-B. Right: predicted trajectories of AgentFormer-VR. Light gray areas indicate the drivable area and dark gray areas indicate pedestrian crossings.

scenarios in the nuScenes dataset. Furthermore, AgentFormer-VR improves collision rates across all datasets. This is particularly crucial in evaluating trajectory forecasting models, as the ability to predict plausible trajectories with minimal collisions is important for autonomous driving applications.

While performance in the min*JADE* / *JFDE* metric drops for the nuScenes-prediction test set, we emphasize that the full nuScenes dataset mostly consists of common or simpler driving scenarios [46], and that evaluation on the more complex and interactive driving subset, nuScenes-interaction, is more critical. For these more safety-critical and dynamic scenarios, leveraging our CARLA-VR dataset substantially improves the robustness of interaction-aware motion predictions.

These results support the thesis argument: improving data coverage of rare but important scenarios leads to better generalization in exactly those scenarios where reliable forecasting matters most.

2.7 Discussion and Limitations

2.7.1 Remaining Challenges

Despite the demonstrated benefits of VR-based data collection, several challenges remain:

Confound: Human Realism vs. Domain Alignment. An important caveat is that the observed gains cannot be attributed solely to the realism of VR-collected human behavior. Because CARLA-VR data originates from the CARLA simulator, part of the improvement may reflect domain alignment between CARLA-VR's environment characteristics and the evaluation setup rather than the behavioral realism of the trajectories themselves. A controlled ablation (for example, comparing VR-collected trajectories against scripted CARLA agent trajectories in the same simulated environment) would be needed to disentangle these two sources of improvement, and is left for future work.

Vehicle Behavior Realism. Interview participants I1 and I2 suggested that data from vehicle simulators may be less noisy than real-life data, enlarging the sim-to-real gap. In the VR CARLA environment, vehicles follow simple policies like the Intelligent Driver Model, always driving perfectly along predefined paths. This may not realistically reproduce human driving behavior, which is imperfect and sometimes deviates from the center of the lane. In future work, more complex and noisy driving policies can be implemented into the CARLA environment to create more realistic vehicle behavior. Additionally, a real human driver can be connected to the simulator as an additional vehicle agent, creating vehicle trajectories that pedestrian users may deem more natural.

Under-represented Agent Categories. I4 pointed out that certain object categories were under-represented in their company's large internal dataset, impacting the ability of AV systems to detect them. For example, small children, specific types of signage such as construction zone signs, skateboarders, trikes, and bicycles were lacking. Future work includes inviting children and elderly to collect data with our system, leveraging the low-risk benefits of the VR environment to obtain much-needed behavioral data of vulnerable road users.

Potential Behavioral Biases. Beyond the vehicle behavior and demographic limitations noted above, the study protocol itself may introduce bias: goal markers and verbal instructions direct participants toward specific destinations, which may elicit more goal-directed behavior than naturalistic pedestrian decision-making, where destinations and crossing decisions emerge from internal intent rather than external prompts. Together, these factors (directed protocols, a young able-bodied participant pool, and deterministic vehicle policies) may limit the degree to which models trained on CARLA-VR data generalize to the full diversity of real-world pedestrian populations and traffic conditions, though the improvements on nuScenes-interaction (which contains real-world traffic with diverse road users) provide some evidence of robustness despite these limitations.

System Limitations Identified by Users. The qualitative responses from Section 2.5.1 revealed certain limitations that can be addressed in future work:

- Environmental sound can be added to the simulation (available in CARLA but disabled in our study so participants could hear verbal instructions)
- Missing visual elements (stop signs, walk signals) can be added as Unreal Engine assets
- The driving simulator extension can integrate human drivers for more natural vehicle movement
- Additional pedestrians can be added to the simulation, either as autonomous agents or additional VR participants

2.7.2 Study Design Limitations

One limitation of the third-person evaluative study is that there may not be enough information from a simple 2D schematic for a human evaluator to determine if a trajectory is "real." As shown by the results, there is enough information present in the 2D trajectory schematics to distinguish between a real-life and a simple synthetic linear policy. However, the evaluator has no access to information that could be used to make a better decision, such as body pose. Our data collection environment includes 8 surround GoPro cameras from which full-body 3D skeletal keypoints can be extracted using multi-view pose estimation methods such as Harmony4D [89]. A natural next step would be a side-by-side realism survey in which evaluators compare animated 3D pose sequences from VR-collected and real-world pedestrians, providing a substantially richer basis for judging behavioral realism than 2D trajectory schematics alone.

It is also important to note that just because a VR-collected trajectory is distinguishable from a real trajectory by a human evaluator, that does not necessarily imply that it is unrealistic or unseen in real-life. The VR system specializes in collecting data of uncommon scenarios. This means that even though the VR-collected data may exhibit aggressive behavior, it is behavior that may still be seen in real-life.

2.8 Chapter Summary

This chapter addressed the foundational layer of reliable trajectory forecasting: data quality and coverage. We demonstrated that existing datasets suffer from a fundamental limitation—they are dominated by routine scenarios and lack sufficient representation of rare but safety-critical events. Through interviews with autonomous vehicle professionals, we confirmed that this limitation is a recognized challenge in the field.

To address this gap, we presented JaywalkerVR, a VR-based human-in-the-loop system that enables safe collection of pedestrian-vehicle interaction data in scenarios that would be dangerous or unethical to recreate in the real world. The key insight underlying this approach is that while the environment is simulated, the human behavior within it is real—participants genuinely respond to virtual traffic, producing naturalistic trajectories that capture the complexity of human decision-making.

We validated the realism of VR-collected data through two complementary studies: a first-person user study demonstrating moderate-high sense of presence and behavioral similarity (ratings of 5.3–5.6 on a 7-point scale), and a third-person evaluation showing that VR trajectories are substantially more realistic than synthetic trajectories and approach the realism of real-world data. Finally, we demonstrated that training models with CARLA-VR data leads to significant improvements in forecasting performance on safety-critical scenarios: 10.7% improvement in displacement error and 4.9% reduction in collision rate.

These results directly support the thesis argument that reliable forecasting depends on comprehensive data coverage. However, better data alone is not sufficient—we also need appropriate evaluation protocols to measure model performance meaningfully. The next chapter addresses the second layer by examining how evaluation metrics can obscure or reveal the true capabilities of forecasting models.

Chapter 3

Evaluation: Refining Benchmarking Protocols

The previous chapter addressed the data foundation for reliable forecasting. This chapter turns to the second layer of the thesis pyramid: evaluation. Even with comprehensive data coverage, progress in forecasting research depends critically on how we measure model performance—metrics serve as both optimization targets and comparison tools, and flawed metrics can actively misdirect research. As discussed in the introduction, standard marginal metrics (ADE/FDE) allow "mix-and-match" across prediction samples, crediting models for coherent joint predictions they never actually made. Because researchers optimize toward the metrics they report, this misleading evaluation signal shapes the models the community builds. This chapter provides, to our knowledge, the first comprehensive empirical demonstration of how severe this problem is, and shows that it can be addressed through both better evaluation and better optimization.

We present a comprehensive evaluation of state-of-the-art pedestrian forecasting methods under *joint metrics*, which require that all agents' predictions come from the same sample rather than allowing each agent's best prediction to come from a different hypothetical future. This reveals a striking gap: joint metric performance is typically twice as poor as marginal performance, indicating that current methods are far less capable of modeling multi-agent interactions than standard benchmarks suggest. Furthermore, we demonstrate that optimizing for joint metrics—by modifying loss functions to include joint terms, with no architectural changes—yields a 7% improvement in JADE, 10% improvement in JFDE, and 16% reduction in collision rate, confirming that the metrics we use to benchmark models directly shape the models we build. The work in this chapter was originally presented in [223].

The chapter proceeds as follows. Section 3.1 provides background on evaluation in multi-agent forecasting, explaining how current metrics emerged and why they fall short. Section 3.2 formally defines marginal and joint metrics. Section 3.3 describes how to modify existing methods to optimize for joint performance. Section 3.4 presents experimental evaluation across state-of-the-art methods. Section 3.5 analyzes results and their implications. Finally, Section 3.6 summarizes the chapter's contributions and bridges to the methods chapter.

3.1 Background: Evaluation in Multi-Agent Forecasting

Before defining joint metrics formally, it is helpful to understand how current evaluation practices emerged and why they are insufficient for multi-agent settings.

3.1.1 The Rise of Marginal Metrics

Trajectory forecasting evaluation has its roots in single-agent prediction problems, where the goal is to predict the future path of one target agent given its history. In this setting, displacement error—the Euclidean distance between predicted and ground-truth positions—is a natural and appropriate metric. Average Displacement Error (*ADE*) measures mean error across all future timesteps, while Final Displacement Error (*FDE*) measures error at the final predicted timestep.

As the field progressed to multi-agent forecasting, where models predict futures for all agents in a scene simultaneously, these single-agent metrics were extended in a straightforward way: compute the error for each agent, then average across agents. This approach, which we term *marginal* evaluation, treats each agent's prediction in isolation, ignoring interactions between agent predictions.

The shift to multi-modal prediction introduced an additional complication. Since the future is inherently uncertain, modern methods generate multiple possible futures (typically $K = 20$ samples in the pedestrian forecasting literature) rather than a single prediction. The standard approach is to evaluate using the "best of K" or "top-K minimum" error: for each agent, find the sample with lowest error and use that for evaluation. This optimistic evaluation acknowledges that any of the K futures might be correct.

Combining these conventions yields the standard evaluation protocol: for each agent, find its best prediction across all K samples, compute the displacement error, then average across agents. This is the marginal top-K *ADE* / *FDE* that dominates current benchmarks. Subsequent work has further questioned whether these error-based metrics capture what actually matters for downstream driving: Da et al. [38] propose a scenario-driven evaluation pipeline that dynamically balances accuracy and diversity based on scenario criticality, showing that traditional ADE/FDE rankings correlate poorly with actual autonomous vehicle driving performance in closed-loop evaluation, corroborating the concerns we raise in this chapter.

3.1.2 Why Marginal Metrics Fail for Multi-Agent Forecasting

The marginal evaluation protocol has a critical flaw: it allows “mix-and-match” across samples. Consider a model that generates $K = 20$ joint futures for a scene with two pedestrians. Under marginal evaluation, we might select sample 3 as the best prediction for pedestrian A, and sample 7 as the best prediction for pedestrian B. But samples 3 and 7 represent *different* futures—they cannot both occur simultaneously. By mixing predictions from different samples, marginal evaluation credits the model for predictions that it never actually made as a coherent joint future.

This mix-and-match problem has concrete consequences. A model can achieve excellent marginal *ADE* / *FDE* while consistently failing to predict coherent interactions. For example:

- **Collision predictions:** Two pedestrians walking toward each other might have accurate individual predictions in different samples, but no single sample correctly predicts collision avoidance.
- **Group divergence:** Two people walking together might have accurate predictions in different samples showing them going different directions, but no single sample keeps them together.
- **Inconsistent modes:** At a crosswalk, one pedestrian might be predicted to cross (in their best sample) while another is predicted to wait (in their best sample), even though in reality they would coordinate.

These failure modes matter for real-world deployment. An autonomous vehicle using a forecasting model must make decisions based on coherent predictions of what *will* happen, not an artificial combination of what *might* happen to different agents independently. If the planner receives predictions where pedestrian A crosses and pedestrian B waits, but the model never actually predicted this combination, the planner is operating on fictional information. A recent survey of multi-agent trajectory prediction methods [66] confirms that the vast majority of methods published between 2020 and 2025 continue to evaluate exclusively on marginal metrics, underscoring the persistence of this gap.

3.1.3 Joint Metrics: A Solution

The solution is straightforward: evaluate joint predictions jointly. Rather than allowing mix-and-match across samples, we require that all agents' predictions come from the same sample before computing the error. This is the *joint* evaluation protocol.

Under joint evaluation, a model can only achieve good scores if at least one of its K samples provides accurate predictions for *all* agents simultaneously. This is a strictly harder criterion—any model that performs well under joint evaluation will also perform well under marginal evaluation, but not vice versa.

Joint metrics were first proposed alongside the Waymo Open Dataset [195] under the name “scene-level” *ADE* / *FDE*, but at the time of this work they had gained little traction, especially in pedestrian forecasting. While Sun et al. [195] defined joint metrics, no prior work had systematically analyzed the marginal-joint gap across methods, demonstrated its practical consequences for pedestrian forecasting, or shown that models architecturally designed for interaction modeling do not automatically produce better joint predictions. This chapter provides that analysis.

Some methods have begun to address joint prediction more directly: JFP [132] explicitly models interactive multi-agent futures, and MotionLM [183] uses autoregressive decoding with attention-based interactive modeling to produce scene-consistent trajectories. Subsequent to our work, diffusion-based approaches such as MotionDiffuser [85] and SceneDM [157] have further advanced joint distribution modeling over all agents' futures. Wang et al. [212] explicitly target collision rate reduction through joint multi-agent prediction on Argoverse 2, demonstrating that scene-consistent trajectory generation can significantly reduce collisions compared to marginal baselines. CausalTraj [203] further validates the importance of joint metrics by proposing a temporally causal, likelihood-based model that achieves the best recorded results on joint metrics (minJADE, minJFDE) in multi-agent sports forecasting. Nevertheless, these methods are evaluated primarily in the vehicle or sports forecasting settings; as of this writing, the pedestrian forecasting community has yet to adopt joint evaluation as standard practice.

3.1.4 Related Work on Evaluation Metrics

Multi-modal Joint Trajectory Forecasting. Modern trajectory forecasting methods produce multi-modal predictions using deep generative architectures such as CVAEs [80, 108, 242], GANs [67, 96, 176], and normalizing flows [169, 170]. To model multi-agent interactions, many of these methods incorporate graph neural networks and transformers [80, 142, 177, 245], hierarchical goal-conditioned architectures [134, 228, 252], and social force modeling [70, 246]. Concurrent work has explored low-rank trajectory descriptors [8] as an alternative paradigm for capturing social dynamics in pedestrian settings. In the vehicle forecasting domain, query-centric architectures [259], game-theoretic interaction modeling [78], and autoregressive token-based generation [183] have achieved state-of-the-art results on large-scale benchmarks. Subsequent to our work, further advances include scalable autoregressive generation [230], angle-based social interaction representations [229], language-model-based prediction [9], lightweight scene-aware architectures such as ASTRA [202] that integrate scene context with graph-aware social modeling, and InSyn [247] which explicitly captures diverse interaction patterns through transparent, pattern-aware modeling. Damirchi et al. [39] propose socially-informed reconstruction with a novel social loss for pedestrian trajectory forecasting (WACV 2025), and Li et al. [111] introduce an intention-aware diffusion model that decomposes short-term and long-term motion intentions for pedestrian prediction. Crucially, while these methods explicitly model social interactions through their architectures, the vast majority evaluate only marginal *ADE* / *FDE*, leaving it unknown whether architectural advances in interaction modeling translate to better joint prediction quality.

Joint Evaluation Metrics. Despite the proliferation of interaction-aware architectures described above, the metrics used to evaluate them have not kept pace. Joint *ADE* / *FDE* (*JADE* / *JFDE*) was introduced in Sun et al. [195], but few papers in either vehicle or pedestrian forecasting have evaluated and reported performance on it. At the time of this work, the only papers that reported it were Ettinger et al. [46], Sun et al. [195, 196], all from the vehicle forecasting community. Even the Waymo Open Dataset Challenges, despite introducing these metrics, ranked submissions using marginal metrics such as min*ADE* / min*FDE*, mAP, and miss rate throughout their motion prediction challenges [216, 217, 218, 219], only introducing a dedicated interaction prediction challenge in 2025 [220]. The nuScenes prediction challenge [18] similarly evaluates and ranks submissions using exclusively marginal metrics—minADE$_5$, minADE$_{10}$, minFDE$_1$, miss rate, and off-road rate—with no joint metrics whatsoever; while the inclusion of an off-road rate metric shows awareness that displacement error alone is insufficient, even this metric is computed per-agent and cannot detect socially implausible joint predictions such as two agents colliding with each other. In pedestrian forecasting, *JADE* / *JFDE* has been largely overlooked in favor of *ADE* / *FDE*.

Collision Rate (*CR*) is a joint evaluation metric that has seen greater attention in more recent works as members of the trajectory forecasting community begin to pay more attention to social compliance and effective joint modeling [97, 98, 100, 128, 173, 191].

In the autonomous vehicle setup, other joint evaluation metrics have been proposed, including Driveable Area Compliance [26], Miss Rate [26, 46, 195], and mean Average Precision [46, 195]. These metrics all measure "realism" aspects of predicted trajectories, but have still not become widespread in the pedestrian trajectory forecasting space. Subsequent to our work, Konstantinidis et al. [95] provide a systematic comparison of approaches for converting marginal predictions into joint ones—including post-processing, explicit joint training, and generative methods—evaluating each in terms of accuracy, multi-modality, and inference efficiency (ITSC 2025). While their focus is on comparing conversion strategies in the vehicle domain, our contribution is the empirical demonstration that the marginal-joint gap exists and matters in pedestrian forecasting, and that interaction-aware architectures do not automatically close it. Bahari et al. [11] introduce certified performance metrics for trajectory prediction that provide guaranteed robustness bounds via randomized smoothing (CVPR 2025), complementing the joint evaluation perspective we advocate here.

The NLL metric captures joint-agent performance because it averages over all pedestrians N, but it is insufficient for capturing best-of-K multi-modality because it averages over samples instead of taking the minimum:

$$NLL = \frac{1}{KNT} \sum_{k,t,i} - \log KDE\_PDF(p_{t,n}^{(i)}, p_{t,n}^{*(i)})$$

Thus, when used to report performance on real datasets with only one ground truth future mode, NLL does not reward plausible but unrepresented modes.

Precision / Recall can only be calculated on datasets that report multiple ground truth modes, such as the Forking Paths Dataset [115], so that is a limitation.

This gap between architectural intent and evaluation practice—methods that explicitly model interactions yet evaluate only marginal metrics—is precisely what motivates the joint metrics studied in this chapter.

3.2 Metric Definitions

We now formally define the marginal and joint metrics used throughout this chapter.

3.2.1 Problem Formulation

We formulate the multi-agent trajectory forecasting problem as predicting the future trajectories of N agents conditioned on their past trajectories. For observed history timesteps $t \leq 0$, we represent the state for agent n at timestep t as $x_{t,n} \in \mathbb{R}_d$, which includes the position, velocity, and (in some methods) the heading angle of the agent. We denote the joint observation history for all N agents over all T timesteps as $\mathbf{x} = (x_{1,1}, \ldots, x_{T,N})$.

For future timesteps $t > 0$, we represent the ground-truth positions of agent n at timestep t as $y_{t,n}^* \in \mathbb{R}_2$, which includes a 2D *x*-*y* position. We denote the ground-truth trajectories over all agent-timesteps as $\mathbf{y}^* = (y_{1,1}^*, \ldots, y_{T,N}^*)$. Similarly, we represent the predicted position for agent n at timestep t in prediction sample k as $y_{t,n}^{(k)} \in \mathbb{R}_2$, and the position predictions over all agent-timestep-samples as $\mathbf{y}$.

The standard evaluation setting uses 8 history or observation timesteps and $T = 12$ future timesteps for a total of 20 frames per sequence sampled at 2.5 fps; thus $3.2s$ of history observation and $T = 4.8s$ of future. K is the number of samples, or possible futures, produced by the model for a single 20-frame sequence; the standard used in evaluations is $K = 20$. N is the number of agents, which varies by sequence.

3.2.2 Marginal Metrics (*ADE / FDE*)

Throughout this chapter we use *ADE / FDE* to refer to *top-K minimum error* rather than average error, as this is the standard notation used in multi-modal human trajectory forecasting evaluation.

Average Displacement Error (ADE) measures mean prediction error across all future timesteps, taking the minimum across samples for each agent:

$$ADE(\mathbf{y}, \mathbf{y}^*) = \frac{1}{TN} \sum_{n=1}^{N} \min_{k=1}^{K} \sum_{t=1}^{T} \left\| y_{t,n}^{(k)} - y_{t,n}^* \right\|_2^2 \tag{3.1}$$

Final Displacement Error (FDE) measures prediction error at the final timestep only:

$$FDE(\mathbf{y}, \mathbf{y}^*) = \frac{1}{N} \sum_{n=1}^{N} \min_{k=1}^{K} \left\| y_{T,n}^{(k)} - y_{T,n}^* \right\|_2^2 \tag{3.2}$$

Note that in both metrics, the $\min$ over samples k is taken *inside* the sum over agents n. This is what allows mix-and-match: each agent can have its minimum achieved by a different sample.

3.2.3 Joint Metrics (*JADE / JFDE*)

The difference between marginal and joint metrics is small but significant: swapping the order of taking the minimum over samples k and taking the average over agents n. This means we cannot mix-and-match agents between different samples; rather we must take the average error over all agents within a sample before selecting the best one.

Joint Average Displacement Error (JADE):

$$JADE(\mathbf{y}, \mathbf{y}^*) = \frac{1}{TN} \min_{k=1}^{K} \sum_{n=1}^{N} \sum_{t=1}^{T} \left\| y_{t,n}^{(k)} - y_{t,n}^* \right\|_2^2 \tag{3.3}$$

Joint Final Displacement Error (JFDE):

$$JFDE(\mathbf{y}, \mathbf{y}^*) = \frac{1}{N} \min_{k=1}^{K} \sum_{n=1}^{N} \left\| y_{T,n}^{(k)} - y_{T,n}^* \right\|_2^2 \tag{3.4}$$

Here, the $\min$ over samples k is taken *outside* the sum over agents n. All agents must come from the same sample.

3.2.4 Collision Rate (*CR*)

Collision Rate measures the proportion of agents whose predicted path intersects in time and space (within a threshold) with at least one other predicted agent future in a $T = 12$-frame prediction sample.

If we adopt top-K evaluation and use the minimum-*JADE* sample as an optimistic measure of the method's prediction ability, then the collision rate of that sample provides an optimistic estimate of the model's collision-avoidance ability. Thus, we define CR_{JADE}, the collision rate of the minimum-*JADE* sample:

$$CR_{JADE}(\mathbf{y}) = \frac{1}{N}\sum_{n=1}^{N} \mathbb{1}\left[collision\left(\mathbf{y}_n^{(k)}, \mathbf{y}_{m\neq n}^{(k)}\right)\right]; \tag{3.5}$$
$$k = \arg\min_{k'} \sum_{n=1}^{N}\sum_{t=1}^{T} \left\| y_{t,n}^{(k')} - y_{t,n}^{*} \right\|_2^2$$

where $\mathbf{y}_n^{(k)}$ represents all predictions $y_{t,n}^{(k)}$ for $t = [1, ..., T]$, and $collision$ is a function that returns True if any two line segments formed by $\left(y_{t,n}^{(k)}, y_{t+1,n}^{(k)}\right) \in \mathbf{y}_n^{(k)}$ and $\left(y_{t,m}^{(k)}, y_{t+1,m}^{(k)}\right) \in \mathbf{y}_m^{(k)} \forall\ m \neq n$ come within $2b$ of each other, and False otherwise. For our evaluations, we use an agent radius of $b = 0.1$ meters, as used in Kothari et al. [99]. We note that the choice of agent radius can materially affect absolute collision rate values: a larger radius increases the collision count, while a smaller one decreases it. While 0.1m follows prior work and provides a consistent basis for comparison, absolute collision rate numbers should be interpreted with awareness that different radius choices would shift them. Relative rankings between methods are expected to be more stable across radius choices than absolute values. In Chapter 4, we discuss analogous sensitivity considerations for the environment collision rate (ECR) threshold.

Adversarial examples discounted, CR_{JADE} should be lower than CR_{mean}, the mean collision rate across all samples. This is because the sample with the best *JADE* is that which is closest to the ground-truth, which has few collisions, as seen in the last row of Table 3.1b.

Because proper learning of social interactions should result in low collision rate, we would ideally like all samples produced by a model to avoid collisions. Thus, we also define CR_{mean} as defined in past work Kothari et al. [99], Sohn et al. [191]:

$$CR_{mean}(\mathbf{y}) = \frac{1}{NK} \sum_{k=1}^{K} \sum_{n=1}^{N} \mathbb{1}\left[collision\left(\mathbf{y}_n^{(k)}, \mathbf{y}_{m \neq n}^{(k)}\right) \right] \tag{3.6}$$

As it considers the mean over samples, rather than the min as in top-K evaluation, CR_{mean} provides a holistic rather than optimistic evaluation of a model's collision-avoidance ability.

3.2.5 Illustrative Example: Why the Distinction Matters

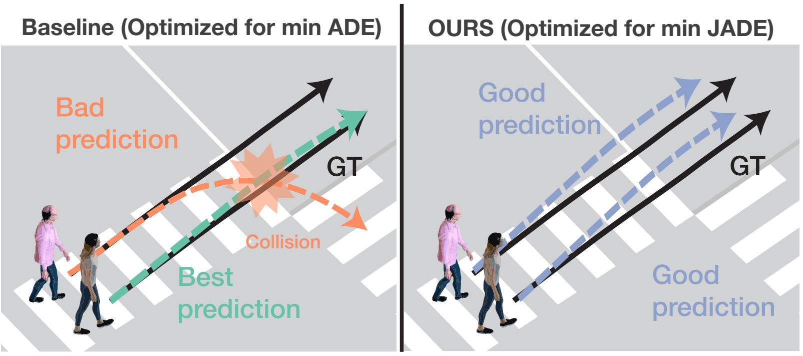


Figure 3.1: Multi-agent trajectory forecasting methods are optimized for single-agent metrics like *ADE* (left panel). As a result, within a single joint future, the method may predict very good trajectories for some agents (e.g. the green agent), but very bad predictions for others (e.g. the orange agent). Optimizing for *JADE* (right panel) encourages the predictions of *all agents* within a joint future to be close to the ground-truth.

Figure 3.1 illustrates the difference between marginal and joint evaluation concretely. Consider two pedestrians crossing a crosswalk together. After crossing, each pedestrian might reasonably continue straight, turn left, or turn right.

A model optimized for marginal *ADE* might predict pedestrian A turning right in sample 1, and pedestrian B going straight in sample 5. Under marginal evaluation, both predictions could be counted as correct. But this combination, where A turns right and B goes straight, might never appear in any single sample, meaning the model never actually predicted this joint future.

If the two pedestrians were walking as a group, they would likely choose the same direction. A model that captures this social structure would predict both going straight in one sample, both turning left in another, and both turning right in a third. Under joint evaluation, such a model would be rewarded for maintaining group coherence.

This example illustrates a general principle: marginal metrics allow models to “hedge” across samples in ways that obscure their actual joint prediction capabilities.

3.3 Joint Optimization of Forecasting Methods

Having established that joint metrics provide a more faithful evaluation of multi-agent forecasting, we now show that optimizing for joint performance, not just evaluating with joint metrics, leads to better interaction modeling. This demonstrates the close connection between evaluation and method development: the metrics we use shape the models we build.

3.3.1 Motivation

In theory, current forecasting architectures may already be capable of modeling joint futures and multi-agent interactions through the use of graph and attention-based architectures such as GNNs and Transformers. These architectures are designed to explicitly model social interactions between agents so as to predict realistic joint futures, and have led to great improvements in *ADE* / *FDE*.

However, increasing studies show that adversarial attacks can cause methods to produce unrealistic joint futures and poor results [22, 173, 249]. For example, Saadatnejad et al. [173] showed that “socially-aware” models may not be as socially-aware as they claim to be, because well-placed attacks can cause predictions with colliding trajectories.

To more realistically assess the performance of multi-agent forecasting, we advocate for the use of joint metrics over marginal metrics in method *evaluation*. Furthermore, we hypothesize that multi-agent architectures fall short in modeling realistic agent interactions because they are optimized with respect to only marginal losses (driven by the field’s focus on marginal metrics). To test this hypothesis, we modify the loss function on popular state-of-the-art methods to include a joint loss term and show that this simple modification makes methods far more accurate at modeling multi-agent interactions.

3.3.2 Joint AgentFormer

AgentFormer [245] is a CVAE structure with transformer layers that factorizes the trajectory forecasting problem into a prediction over a latent space of possible futures $\mathbf{z}$ conditional on the trajectory histories $\mathbf{x}$, and a prediction over decoded possible futures $\mathbf{y}$ conditional on the latent $\mathbf{z}$. We make no modifications to the AgentFormer architecture, and thus refer readers to the original paper for more detail.

AgentFormer training involves a two-step procedure: a first step to learn accurate trajectory decoding, and a second tuning step to learn to produce diverse prediction samples. During the first step, AgentFormer makes use of the negative evidence lower bound (ELBO) loss function to encourage the CVAE model's predictions to match the ground-truth positions while maintaining that the latent space of possible futures adhere to the Gaussian distribution.

$$\mathcal{L}_{elbo} = -\mathbb{E}_{q_\phi}[\log p_\theta(\mathbf{y}|\mathbf{z},\mathbf{x})] + \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{y},\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})] \tag{3.7}$$

This loss can be rewritten as the three terms in black:

$$\begin{aligned}\mathcal{L}_{elbo} = \sum_n ||\mathbf{y}_n^{(0)} - \mathbf{y}^*||^2 + \sum_n \min_{k=1}^{K} ||\mathbf{y}_n^{(k)} - \mathbf{y}_n^*||^2 \qquad (3.8)\\ {\color{blue} + \min_{k=1}^{K} \sum_n ||\mathbf{y}_n^{(k)} - \mathbf{y}_n^*||^2} + \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{y},\mathbf{x}) \; || \; p_\theta(\mathbf{z}|\mathbf{x}))\end{aligned}$$

The first and second terms are reconstruction terms resulting from the first term of Equation 3.7 (the ELBO likelihood term). The first term is a general reconstruction loss that encourages a single predicted future $\mathbf{y}^{(0)}$ to be close to the ground-truth $\mathbf{y}^*$. The second term is a marginal sample reconstruction loss that encourages the min *ADE* prediction for agent n, $\min_k ||\mathbf{y}_n^{(k)} - \mathbf{y}_n^*||$, to be close to the ground-truth $\mathbf{y}_n^*$. The fourth term is equivalent to the second term of Equation 3.7 (the KL divergence term), which encourages the *CVAE Prior* network, p_θ, to learn the latent distribution encoded by the posterior network q_ϕ.

In our joint optimization, we modify the objective function by adding the third term in blue, a joint sample reconstruction loss that encourages the min *JADE* prediction $\min_k \sum_n ||\mathbf{y}_n^{(k)} - \mathbf{y}_n^*||$ to be close to the ground-truth $\mathbf{y}^*$.

While the first training step learns a latent space that maximizes the probability that decoded predictions match the ground-truth, AgentFormer makes use of a second training step designed to encourage the decoded predictions to be diverse. Here, the CVAE weights are fixed, and the *CVAE Prior* network's latent sampler is swapped out for a *DLow Trajectory Sampler*. This new sampler module learns a fixed set of K linear transformations of the latent $\mathbf{z}$ that are trained to be different from one another via a DLow diversity loss $\mathcal{L}_{samp}$ [244]. We signify the new prior network with the *DLow Trajectory Sampler* as r_θ. In our joint optimization, we modify AgentFormer's objective function by adding a joint sample reconstruction term, just as we did in the first training step. The final objective function for the second training step is:

$$\begin{aligned}\mathcal{L}_{samp} = &\sum_n \min_{k=1}^{K} ||\mathbf{y}_n^{(k)} - \mathbf{y}_n^*||^2 + \min_{k=1}^{K} \sum_n ||\mathbf{y}_n^{(k)} - \mathbf{y}_n^*||^2 \\ &+ \mathrm{KL}\big(r_\theta(\mathbf{z}|\mathbf{x}) \;||\; p_\theta(\mathbf{z}|\mathbf{x})\big) \\ &+ \frac{1}{K(K-1)} \sum_{k_1=1}^{K} \sum_{k_1 \neq k_2}^{K} \exp\left(- \frac{||\mathbf{y}^{(k_1)} - \mathbf{y}^{(k_2)}||}{\sigma_d} \right)\end{aligned} \tag{3.9}$$

The first term is the marginal sample reconstruction loss, analogous to the second term of Equation 3.8. The blue second term, our addition, is equivalent to the joint sample reconstruction loss we added to the first training step. The third term is a KL term which encourages the new prior network r_θ (with the new *Trajectory Sampler* module) to be near the distribution of the original prior network p_θ (with the old *CVAE Prior* sampler, learned in the first training step). The fourth term is the diversity loss, which encourages the fixed set of K futures to be diverse from one another.

3.3.3 Joint View Vertically

View Vertically [228] is a simple hierarchical method with two modules: a coarse-level waypoint prediction module, and a fine-level trajectory prediction module. The coarse-level module forecasts future waypoints in the spectrum domain, and the fine-level module interpolates waypoints in the spectrum domain, then decodes full trajectories in coordinate space. We make no modifications to the architecture, and refer readers to the original paper for more detail.

Each module is trained independently. The coarse-level module is optimized according to the loss written in black:

$$
\begin{aligned}
\mathcal{L}_{coarse} = &\frac{1}{N_{way}N_{agents}} \sum_{n} \min_{k=1}^{K} \sum_{m=1}^{N_{way}} ||\mathbf{y}_{m,n}^{(k)} - \mathbf{y}_{m,n}^{*}||^2 \\
&+ \omega \cdot \frac{1}{N_{way}N_{agents}} \min_{k=1}^{K} \sum_{n} \sum_{m=1}^{N_{way}} ||\mathbf{y}_{m,n}^{(k)} - \mathbf{y}_{m,n}^{*}||^2
\end{aligned}
\tag{3.10}
$$

Here, N_{way} is the set of waypoint timesteps used for coarse prediction, optimized as a hyperparameter. Similar to how we do with AgentFormer, to optimize View Vertically with joint metrics, we add the blue second term, the joint optimization term. Here, ω is a weighting hyperparameter that balances how much to consider the marginal term vs. the joint term.

The fine-level module is optimized according to the loss written in black:

$$
\begin{aligned}
\mathcal{L}_{fine} = &\frac{1}{TN} \sum_{n} \min_{k=1}^{K} \sum_{t} ||\mathbf{y}_{t,n}^{(k)} - \mathbf{y}_{t,n}^{*}||^2 \\
&+ \omega \cdot \frac{1}{TN} \min_{k=1}^{K} \sum_{n} \sum_{t} ||\mathbf{y}_{t,n}^{(k)} - \mathbf{y}_{t,n}^{*}||^2
\end{aligned}
\tag{3.11}
$$

We add in the term in blue, analogous to the term we add for the coarse-level module.

3.4 Experimental Setup

3.4.1 Datasets

We evaluate on the commonly-used pedestrian trajectory datasets ETH [154] / UCY [110], used across the field in nearly all pedestrian trajectory forecasting works. ETH / UCY features bird's eye view trajectories of pedestrians in 5 different environment scenes given in real-world coordinates (meters). Following Alahi et al. [4], Gupta et al. [67], Mangalam et al. [134, 135], Salzmann et al. [177], Yuan et al. [245], we use the leave-one-out training and evaluation setup, where we train on all but one of the 5 environment scenes and evaluate on the left-out scene.

We also evaluate on the Stanford Drone Dataset (SDD) [171], which features trajectories captured via drone from a bird's eye viewpoint across 20 different scenes on Stanford University's campus given in image pixel coordinates. Instead of using the raw data, we use the TrajNet split [100] of SDD, a downsampled and smaller subset of the original data considering only pedestrian trajectories, following the training and evaluation setup of Sadeghian et al. [176], Salzmann et al. [177].

3.4.2 Baselines

We compare our approach with current state-of-the-art methods: View Vertically [228], MemoNet [232], Y-Net [134], and AgentFormer [245]; and other standard baselines from recent years: Trajectron++ [177], PECNet [135], and S-GAN [67]. We re-evaluate using pre-trained models if provided, or retrain and re-evaluate using available source code if not.

For fair comparison between methods, we provide the model only ground-truth trajectory histories, and do not provide scene context information such as images or semantic maps. Y-Net is the only baseline in our evaluation that used semantic map information in the original work [135]; our reported results for Y-Net are obtained by retraining without map information.

There are a few other reasons for differences between our results and those reported in the original papers: the numbers originally reported in Trajectron++ were incorrect due to a data snooping bug; thus we re-train and re-evaluate on a corrected version of the code. View Vertically used a different sample rate on the `eth` scene and a slightly different split of the `hotel` scene in the ETH dataset; thus, we retrain and re-evaluate it on the standard split used by most other methods [67, 134, 135, 177, 232, 245]. To ensure fair comparison across methods on SDD, we re-train and re-evaluate methods which either used a different split instead of the TrajNet [99] split (View Vertically used the original raw dataset and downsampled/preprocessed the data themselves) or did not originally train and report results on SDD (AgentFormer and Trajectron++).

3.5 Results and Analysis

3.5.1 The Gap Between Marginal and Joint Performance

JADE / JFDE. Joint metrics (Table 3.1a) perform about 2x worse across the board as compared to marginal metrics (Table 3.1c). This means that while methods can achieve excellent predictions for individual agents across different prediction samples, they perform much worse at producing good predictions for all agents within a single prediction sample. This provides strong evidence that marginal metrics are overly optimistic estimations of trajectory forecasting performance.

Joint AgentFormer achieves the best *JADE* and *JFDE* of all methods: 7% better in *JADE* and 10% better in *JFDE* over AgentFormer, the next-best method on ETH / UCY. Joint View Vertically also achieves a 4% boost in *JADE* and a 7% boost in *JFDE* over vanilla View Vertically. The performance of all methods with respect to *JADE* / *JFDE* is summarized in Table 3.1a.

For both of our methods, there is a significantly large improvement in *JADE* / *JFDE* performance (18% for Joint AgentFormer and 10% for Joint View Vertically) in the `zara2` scene, an environment with plenty of interactions caused by medium-density sequences (about 6 pedestrians per 20-frame sequence). We hypothesize that our optimization method causes increased performance particularly on high-density sequences, in which there exist more interactions than in low-density sequences. Another example is the `univ` scene, the densest scene, in which Joint View Vertically achieves a 20% improvement, and Joint AgentFormer achieves a 5% improvement. It is worth noting that joint optimization results in a smaller improvement margin on scenes with few interactions (`eth`, `hotel`, and `SDD TrajNet`). Perhaps adding consideration of joint dynamics in these simple scenes may confuse the model and impede accurate prediction.

While our method does not achieve the best *ADE* / *FDE* (Table 3.1c), we argue that the decrease in *ADE* / *FDE* is worth the improvement in *JADE* / *JFDE*. As seen in the trajectory visualizations in Figure 3.2, which compares predictions from our method to that of View Vertically, a method may achieve excellent *ADE* / *FDE* yet still fall short at producing natural and socially-compliant trajectories for certain agents.

3.5.2 Collision Rate

Joint AgentFormer performs best across the board with respect to collision rate, as seen in Table 3.1b. We gain an improvement in collision rate as compared with the baselines although we do not optimize explicitly with respect to it, substantiating our claim that optimizing for joint performance also leads naturally to a decrease in collision rate.

3.5.3 Analysis by Interaction Category

We studied the effect of our improvements across different cross sections of the data. Specifically, we heuristically define 3 different categories of interactions present within the data: *group*, *leader-follower*, and *collision-avoidance*. These interaction categories are used to categorize individual agents based on whether that agent interacts in a certain way with at least one other agent within a 20-frame sequence. The categories are not disjoint, so each agent may fall into none, one, or multiple categories. *Group* defines an agent who moves in parallel with another agent; *leader-follower* defines an agent who either is moving behind or before another agent in the same direction as that other agent; *collision-avoidance* defines an agent that comes within a distance threshold to another agent that is not moving in the same direction. Definitions of the heuristics used to create these categories can be found in the supplementary material; we take inspiration from and modify the heuristics used in Kothari et al. [100]. Figure 3.3 shows examples of pedestrians within each category.

Using our defined categories, we examine collision rate performance on the ETH / UCY dataset. We highlight the success of our method with respect to modeling pedestrians involved in interactions, as seen in Table 3.2: a 23% decrease in *group*, a 19% decrease in *collision-avoidance*, and a 24% decrease in *leader-follower*. These results substantiate our claim that our method improves social compliance.

3.5.4 Ablation Studies

We perform ablations to study the effect on joint metric performance of using the marginal reconstruction loss and/or the joint reconstruction loss in both steps of AgentFormer training, as seen in Table 3.3.

An interesting observation is that training AgentFormer with only joint loss during both training steps (line 3 of Table 3.3a) does not result in as good *JADE* / *JFDE* as compared with training with both marginal and joint loss in the first step, and then only joint loss in the second (as we do in Joint AgentFormer, line 4 of Table 3.3a). A possible reason for this is that the joint prediction problem is inherently more difficult than the marginal prediction problem, due to having to optimize for joint performance of multiple agents rather than individual agents independently. As the joint loss function more naturally captures social compliance for the joint prediction problem than the marginal loss function, as previously established, optimizing it acquires the difficulty of the joint prediction problem.

Another point of interest with regard to the two-step training procedure of AgentFormer is that models trained with joint loss during the first AgentFormer training step show greater improvement in mean collision rate (Line 3 and 4 in Table 3.3b). This may be due to the fact that the first training step accounts for the majority of AgentFormer training, as that is when most weights of the CVAE are trained; the second training step only learns the weights of the *Trajectory Sampler*, which account for only a small fraction of the entire network.

3.6 Chapter Summary

This chapter addressed the second layer of reliable trajectory forecasting: good benchmarking and evaluation protocols. We demonstrated that widely used marginal metrics (*ADE* / *FDE*) provide an incomplete and often misleading picture of forecasting model capabilities, allowing methods to achieve excellent scores while producing physically implausible predictions with collisions and inconsistent group behaviors.

The core insight is that trajectory forecasting is inherently a *joint* prediction problem. Standard marginal metrics ignore this structure by allowing “mix-and-match” across prediction samples, crediting models for coherent joint predictions they never actually made. Our comprehensive evaluation revealed a striking gap: joint metric performance is typically twice as poor as marginal metric performance across state-of-the-art methods, indicating that current benchmarks substantially overestimate model capabilities.

To address this, we advocate for the widespread adoption of joint metrics (*JADE* / *JFDE*) in trajectory forecasting evaluation. Joint metrics require that all agents’ predictions come from the same sample, providing a more faithful assessment of whether models can produce coherent multi-agent futures.

Furthermore, we demonstrated that optimizing for joint metrics, not just evaluating with them, leads to substantial improvements. By adding joint loss terms to existing methods (AgentFormer and View Vertically), we achieved:

- 7% improvement in *JADE* and 10% improvement in *JFDE* over baselines
- 16% reduction in collision rate on the ETH/UCY datasets
- Particularly large improvements in high-density, interaction-rich scenarios

These results demonstrate the tight coupling between evaluation and method development. The metrics we use to benchmark models shape the models we build. By adopting metrics that better capture the requirements for real-world deployment, specifically coherent multi-agent predictions that avoid collisions, we can guide the field toward methods that are genuinely more reliable.

Limitations of best-of-K evaluation. While joint metrics address the mix-and-match problem, both marginal and joint best-of-K metrics remain fundamentally optimistic: they select the single best sample out of K and discard the remaining $K - 1$. This means they assess best-case prediction quality but do not reflect the quality distribution across all K samples. For downstream planning applications where the planner may consume all predicted modes (weighted by their probabilities), typical sample quality matters as much as best-case quality. CR_{mean} partially addresses this concern by averaging collision rate across all samples. In Chapter 4, we further address this limitation by evaluating ACR and ECR on the highest-likelihood mode rather than the best-of-K sample, so that collision metrics reflect the model's most confident prediction rather than its luckiest one.

The marginal-joint gap, the optimism of best-of-K selection, and the sensitivity of collision metrics to parameter choices all point toward the same conclusion: a single metric number is insufficient to characterize a forecasting method. We distill these observations into a concrete reporting checklist in Chapter 5.

Since publication, the joint metrics advocated in this chapter have seen growing adoption across the trajectory forecasting community. Several subsequent methods have adopted *JADE*/*JFDE* as primary evaluation metrics: Lin et al. [122] use *JADE*/*JFDE* as their main metrics for evaluating a diffusion-based joint pedestrian prediction framework (IROS 2024), directly benchmarking against our Joint AgentFormer and using our released pretrained checkpoints; Fu et al. [50] report *JADE*/*JFDE* in their flow-matching trajectory prediction model (CVPR 2025); and Teoh [203] extend joint metrics to the sports analytics domain, explicitly crediting this work for highlighting the importance of joint evaluation and building their entire evaluation protocol around *JADE*/*JFDE*. Beyond direct metric adoption, the concerns raised in this chapter have influenced broader methodological directions: Liu et al. [124] incorporate joint evaluation in their diffusion-based multi-modal prediction framework (IJCAI 2024), and Konstantinidis et al. [95] provide a systematic comparison of approaches for converting marginal predictions into joint ones (ITSC 2025), a research direction motivated by the marginal-joint gap we identified. This adoption across venues (ICCV, IROS, CVPR, IJCAI, ITSC), domains (pedestrian, vehicle, sports), and methodological paradigms (diffusion, flow matching, autoregressive) suggests that the joint evaluation perspective introduced in this chapter addresses a genuine and broadly recognized need in the multi-agent forecasting community.

With the foundations of good data coverage (Chapter 2) and good evaluation protocols (this chapter) in place, the next chapter turns to the top layer of the pyramid: developing forecasting methods that can fully leverage rich multi-modal sensor data to produce accurate, socially consistent, and environmentally aware predictions.

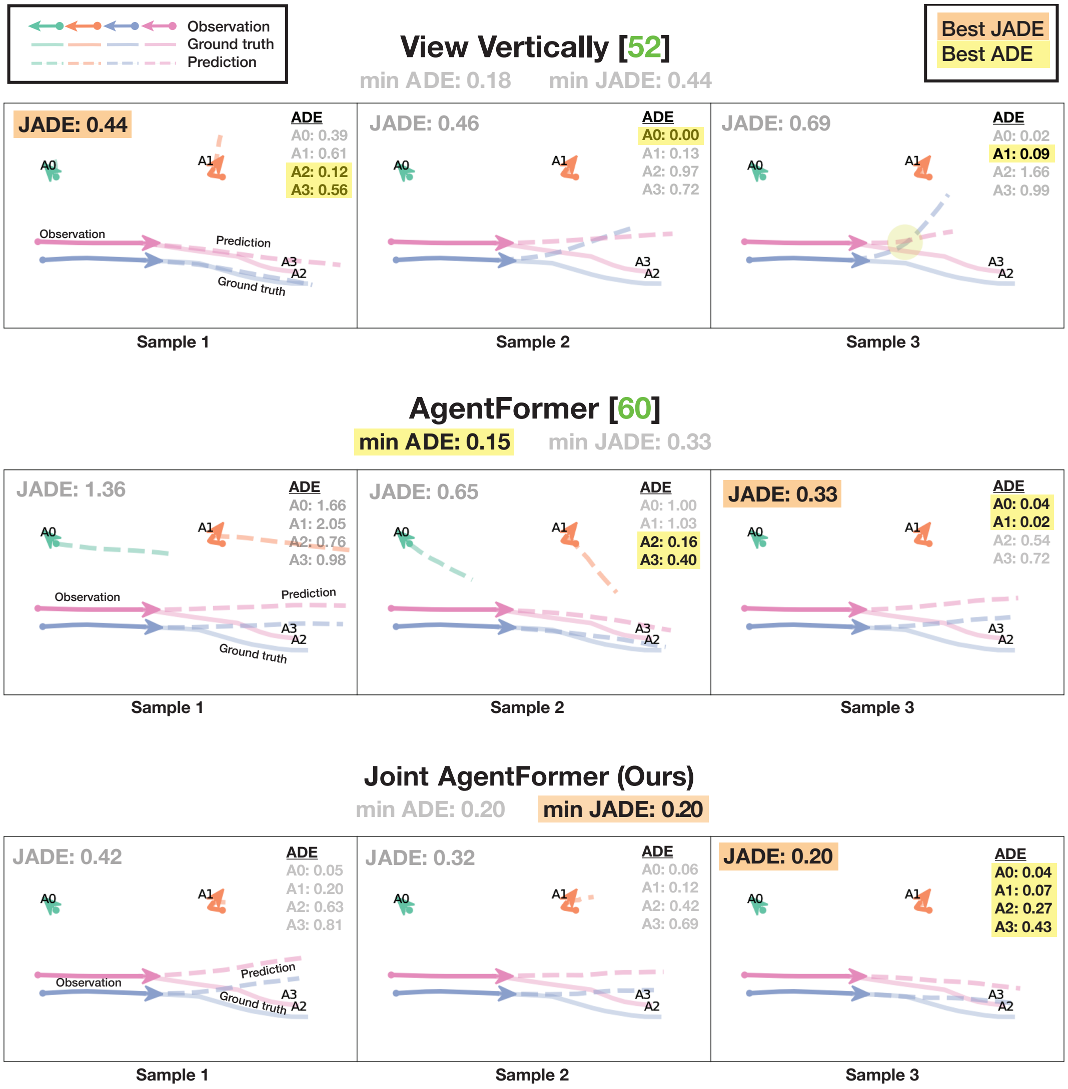


Figure 3.2: Comparison of Joint AgentFormer (last row) with two baselines, View Vertically [228] and AgentFormer [245]. Legend is in the upper-left corner. A# stands for Agent #. Best per-pedestrian *ADE* values are highlighted yellow; best *JADE* values are highlighted orange. View Vertically (1st row) and AgentFormer (2nd row), optimized for *ADE*, achieve a better *ADE* than Joint AgentFormer by mixing and matching pedestrians from different samples. However, they have lower *JADE* than Joint AgentFormer, because no single sample has good *JADE*. On the other hand, our method's best *ADE* is equal to our best *JADE* (bottom right), since our method was optimized to encourage all pedestrians within a sample to have low error. Baseline methods that optimize for *ADE* / *FDE* rather than *JADE* / *JFDE* have several other shortcomings. For example, in the top-right panel, View Vertically predicts colliding trajectories (collision denoted by the light yellow circle). In the top-middle panel, it predicts diverging trajectories for two pedestrians that were clearly walking as a group. In spite of these failures, View Vertically still achieves excellent *ADE*, pointing to a shortcoming in evaluation using only marginal metrics as well as optimization using only marginal loss.

Table 3.1: Baseline evaluations across state-of-the-art methods (first 7 rows) as well as our Joint AgentFormer and Joint View Vertically methods (last rows), on the ETH / UCY datasets (first 6 columns) and the Stanford Drone dataset (last column). The metrics being reported (a.) *JADE / JFDE*, (b.) CR_{JADE} / CR_{mean}, and (c.) *ADE / FDE*) are shown in the title of each table. Sequence density, as given by mean number of agents per 20-frame sequence, is shown in parentheses next to each dataset name. Lower values are better; bolded values show best result, underlined values show second-best. All results are for $K = 20$ prediction samples per sequence.

(a) $\min \textit{JADE}/\textit{JFDE}$

	$\min \text{JADE}_{20}/\text{JFDE}_{20} \downarrow$ (m), $K = 20$ samples						
Dataset (mean # peds) →	ETH (1.4)	HOTEL (2.7)	UNIV (25.7)	ZARA1 (3.3)	ZARA2 (5.9)	ETH / UCY Avg.	SDD TrajNet (1.5)
S-GAN [67]	0.919 / 1.742	0.480 / 0.950	0.744 / 1.573	0.438 / 1.001	0.362 / 0.794	0.589 / 1.212	13.76 / 24.84
Trajectron++ [177]	0.726 / 1.299	0.237 / 0.418	0.609 / 1.316	0.359 / 0.712	0.294 / 0.625	0.445 / 0.874	11.36 / 18.21
PECNet [135]	0.618 / 1.097	0.291 / 0.587	0.666 / 1.417	0.408 / 0.896	0.372 / 0.840	0.471 / 0.967	10.82 / 19.48
Y-Net [134]	0.495 / 0.781	0.205 / 0.386	0.695 / 1.559	0.487 / 1.045	0.492 / 1.101	0.475 / 0.974	9.67 / **16.01**
MemoNet [232]	0.499 / 0.859	0.222 / 0.416	0.686 / 1.466	0.349 / 0.723	0.385 / 0.864	0.428 / 0.866	9.59 / 16.43
View Vertically [228]	0.561 / **0.776**	0.196 / 0.332	0.654 / 1.307	0.328 / 0.654	0.298 / 0.602	0.408 / 0.734	10.75 / 17.45
Joint View Vertically (Ours)	0.652 / 0.839	**0.186 / 0.309**	**0.523 / 1.091**	0.331 / 0.634	0.267 / 0.547	0.392 / 0.684	10.92 / 17.70
AgentFormer [245]	**0.482** / 0.794	0.237 / 0.456	0.622 / 1.310	0.285 / 0.564	0.296 / 0.624	0.384 / 0.749	9.67 / 16.92
Joint AgentFormer (Ours)	0.485 / 0.798	0.186 / 0.320	0.590 / 1.219	**0.271 / 0.513**	**0.252 / 0.509**	**0.357 / 0.672**	**9.56** / 16.59

(b) CR_{JADE} / CR_{mean}

	$\text{CR}_{mean}/\text{CR}_{JADE} \downarrow$ (m), $K = 20$ samples						
Dataset (mean # peds) →	ETH (1.4)	HOTEL (2.7)	UNIV (25.7)	ZARA1 (3.3)	ZARA2 (5.9)	ETH / UCY Avg.	SDD TrajNet (1.5)
S-GAN [67]	0.015 / **0.045**	0.031 / 0.090	0.165 / **0.251**	0.060 / 0.185	0.083 / 0.195	0.071 / 0.153	0.00 / 0.00
Trajectron++ [177]	0.025 / 0.137	0.044 / 0.271	0.281 / 0.489	0.088 / 0.466	0.126 / 0.456	0.113 / 0.364	0.00 / 0.00
PECNet [135]	0.014 / 0.115	0.043 / 0.269	0.218 / 0.409	0.059 / 0.396	0.128 / 0.455	0.092 / 0.329	0.00 / 0.03
Y-Net [134]	0.016 / 0.141	0.039 / 0.250	0.265 / 0.482	0.100 / 0.513	0.134 / 0.480	0.111 / 0.373	0.00 / 0.00
MemoNet [232]	0.014 / 0.160	0.040 / 0.301	0.206 / 0.415	0.065 / 0.445	0.136 / 0.483	0.092 / 0.361	0.00 / 0.00
View Vertically [228]	0.014 / 0.090	0.029 / 0.203	0.212 / 0.428	0.045 / 0.233	0.082 / 0.316	0.077 / 0.254	0.00 / 0.00
Joint View Vertically (Ours)	**0.011** / 0.076	0.026 / 0.168	0.276 / 0.484	0.045 / 0.262	0.081 / 0.349	0.088 / 0.268	0.00 / 0.00
AgentFormer [245]	0.016 / 0.068	0.022 / **0.084**	0.204 / 0.362	0.021 / **0.088**	**0.054 / 0.139**	0.063 / **0.148**	**0.00 / 0.00**
Joint AgentFormer (Ours)	0.013 / 0.064	**0.019** / 0.094	**0.163** / 0.333	**0.021** / 0.100	0.055 / 0.203	**0.054** / 0.159	0.00 / 0.00
Ground Truth	0.000	0.001	0.021	0.000	0.002	0.005	0.00

(c) $\min \textit{ADE}/\textit{FDE}$

	$\min \text{ADE}_{20}/\text{FDE}_{20} \downarrow$ (m), $K = 20$ samples						
Dataset (mean # peds) →	ETH (1.4)	HOTEL (2.7)	UNIV (25.7)	ZARA1 (3.3)	ZARA2 (5.9)	ETH / UCY Avg.	SDD TrajNet (1.5)
S-GAN [67]	0.876 / 1.656	0.461 / 0.920	0.639 / 1.343	0.379 / 0.816	0.285 / 0.600	0.528 / 1.067	12.74 / 22.65
Trajectron++ [177]	0.669 / 1.183	0.185 / 0.283	0.303 / 0.541	0.249 / 0.414	0.175 / 0.319	0.316 / 0.548	10.18 / 15.76
PECNet [135]	0.562 / 0.985	0.192 / 0.332	0.336 / 0.630	0.243 / 0.468	0.179 / 0.345	0.302 / 0.552	9.34 / 16.10
Y-Net [134]	**0.398 / 0.571**	0.123 / 0.189	0.310 / 0.598	0.258 / 0.491	0.198 / 0.389	0.257 / 0.448	8.15 / **12.80**
MemoNet [232]	0.410 / 0.636	**0.113 / 0.173**	**0.244 / 0.433**	0.184 / 0.320	0.143 / 0.248	**0.219 / 0.362**	**7.97** / 12.82
View Vertically [228]	0.569 / 0.691	0.124 / 0.188	0.290 / 0.499	0.202 / 0.356	0.147 / 0.257	0.267 / 0.398	9.34 / 14.67
Joint View Vertically (Ours)	0.700 / 0.792	0.129 / 0.196	0.267 / 0.474	0.219 / 0.359	0.144 / 0.247	0.292 / 0.414	9.62 / 15.07
AgentFormer [245]	0.451 / 0.748	0.142 / 0.225	0.254 / 0.454	**0.177 / 0.304**	**0.140 / 0.236**	0.233 / 0.393	8.01 / 13.24
Joint AgentFormer (Ours)	0.473 / 0.792	0.135 / 0.212	0.285 / 0.505	0.189 / 0.321	0.144 / 0.242	0.245 / 0.414	8.25 / 13.74

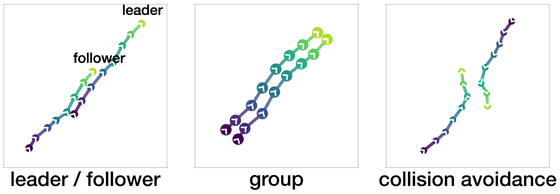


Figure 3.3: Interaction Categories.

Table 3.2: Comparison of collision rate performance of AgentFormer vs. Joint AgentFormer on different interaction categories in the ETH / UCY dataset. In parentheses below each interaction category name shows the proportion of pedestrians belonging to that category across all 20-frame sequences, out of 34161 pedestrians total.

	$CR_{mean} \downarrow$			
Interaction Category (prop. peds) →	group (0.44)	collision avoidance (0.61)	leader-follower (0.03)	ETH/UCY aggregate (1.0)
AgentFormer [245]	0.103	0.201	0.124	0.063
Joint AgentFormer (Ours)	**0.084**	**0.180**	**0.103**	**0.053**
ground-truth	0.010	0.011	0.028	0.005

Table 3.3: Ablation studies on Marginal (M) and Joint (J) loss terms in AgentFormer training. Baseline is original AgentFormer [245].

(a) *JADE / JFDE*

	$\min \text{JADE}_{20}/\text{JFDE}_{20} \downarrow$ (m)				
	step 1		step 2		
	M	J	M	J	Avg.
Baseline	✓		✓		0.384 / 0.749
-	✓			✓	0.365 / 0.694
-		✓		✓	0.386 / 0.734
-	✓	✓		✓	0.358 / 0.672
Ours	✓	✓	✓	✓	**0.357 / 0.672**

(b) Collision Rate

	$\text{CR}_{mean} \downarrow; K = 20$				
	step 1		step 2		
	M	J	M	J	Avg.
Baseline	✓		✓		0.063
-	✓			✓	0.064
-		✓		✓	**0.051**
-	✓	✓		✓	0.053
Ours	✓	✓	✓	✓	0.054

(c) *ADE / FDE*

	$\min ADE_{20}/FDE_{20} \downarrow; K = 20$				
	step 1		step 2		
	M	J	M	J	Avg.
Baseline	✓		✓		**0.233 / 0.393**
-	✓			✓	0.255 / 0.436
-		✓		✓	0.293 / 0.516
-	✓	✓		✓	0.258 / 0.439
Ours	✓	✓	✓	✓	0.245 / 0.414

Chapter 4

Methods

4.1 Introduction

The preceding chapters established the two foundational layers upon which reliable trajectory forecasting methods are built: comprehensive data coverage (Chapter 2) and faithful evaluation protocols (Chapter 3). With these foundations in place, this chapter turns to the top layer of the pyramid: developing forecasting methods that fully leverage the wealth of information present in multi-modal sensor data, moving beyond the trajectory-only approaches that dominate current leaderboards.

As motivated in the introduction (Section 1), pedestrian prediction faces semantic ambiguities largely absent in the vehicle domain—soft spatial preferences, context-dependent affordances, and unstructured social dynamics—that trajectory history alone cannot resolve. Figure 4.1 illustrates examples of these ambiguities. The central technical challenge addressed in this chapter is how to integrate the two complementary information sources identified there—body pose and dense environmental context—into a unified architecture without the noisier or domain-shifted signals corrupting the well-calibrated trajectory representations.

This chapter presents PECT (Pose and Environment-Contextualized Transformer), a multi-modal trajectory prediction framework that explicitly incorporates body pose signals and dense environmental semantics alongside trajectory history. PECT augments a trajectory prediction backbone with a graph-based keypoint encoder and an agent-centric BEV environment encoder, fused through a gated curriculum strategy that prevents the noisier environment signal from corrupting early training. We also introduce the environment collision rate (ECR) metric to capture a failure mode, namely predicted trajectories passing through walls and static structures, that standard displacement metrics cannot detect. PECT demonstrates improvements in both agent-agent and environment collision rates without sacrificing displacement accuracy. The following sections review foundational work in pose-based intent estimation and environment-aware scene understanding (Section 4.2), then present the PECT architecture (Section 4.3), experimental setup and results (Section 4.4), and a discussion of findings (Section 4.5).

4.2 Background and Related Work

4.2.1 3D Pose Estimation

Recovering 3D human pose from visual observations has progressed rapidly over the past decade. Early approaches adopted a two-stage pipeline: a 2D keypoint detector such as OpenPose [24] or HRNet [194] first localizes joints in the image, and a lifting network regresses 3D coordinates from the 2D detections [139, 153]. Temporal lifting methods such as VideoPose3D [153] exploit video-level context through dilated temporal convolutions, improving accuracy on occluded or ambiguous frames. More recent architectures integrate graph convolutions with transformers to jointly model skeletal topology and temporal dynamics [112, 140], achieving strong results on standard benchmarks such as Human3.6M [79]. KTPFormer [155] further improves lifting accuracy by encoding kinematic chain constraints and trajectory priors directly into the attention mechanism. MotionBERT [260] further unifies pose estimation, action recognition, and mesh recovery under a single pretraining framework, demonstrating that learned motion representations transfer broadly across human understanding tasks. In the monocular mesh recovery line, 4DHumans/HMR 2.0 [58] introduced a fully transformer-based architecture for joint human mesh recovery and tracking, and CameraHMR [152] extended this paradigm with perspective-aware camera modeling that improves accuracy at varying subject depths—a property relevant to autonomous driving where pedestrians span a wide range of distances from the ego-vehicle. In the autonomous driving setting, LiDAR and multi-camera rigs enable metric-scale 3D pose recovery without the depth ambiguity inherent to monocular methods. LiDAR-HMR [47] proposed the first point-cloud-to-SMPL pipeline, and

LiveHPS [167] extended LiDAR-based pose to scene-level multi-person estimation in outdoor environments. DAPT [231] addresses the challenge of varying point cloud densities across distances through density-aware pretraining, and a recent survey [54] provides a comprehensive review of LiDAR-based 3D human pose and mesh recovery methods. The recently released JRDB-Pose3D [211] provides multi-person 3D pose and shape annotations in crowded robotic navigation scenes, establishing a benchmark for this setting. For multi-person scenarios, end-to-end architectures have advanced rapidly: WTPose [161] introduces a waterfall transformer module that generates multi-scale feature maps to capture both local and global context in a single pass (WACV 2025), and EVT [106] leverages event camera data alongside video frames to achieve end-to-end 3D pose estimation with improved temporal resolution (WACV 2025). We obtain keypoints using SAM 3D Body [236], a foundation model for single-image full-body 3D human mesh recovery trained on large-scale diverse data, which achieves strong generalization to in-the-wild conditions including the crowded outdoor scenes in our target dataset.

4.2.2 Crossing Intent and Pose-Based Behavioral Cues

Body pose provides a rich signal for anticipating pedestrian behavior: head orientation, arm gestures, and postural shifts often precede observable trajectory changes [160, 165]. The crossing intent prediction literature, built on the JAAD [101, 163] and PIE [164] benchmarks, has shown that visual behavioral cues, particularly head orientation and body pose, are among the strongest predictors of pedestrian crossing decisions [102, 103, 131, 165]. More recent work has extended these findings with multi-modal transformer architectures: PedFormer [166] fuses trajectory, pose, and scene context through cross-modal attention to jointly predict crossing actions and future trajectories, while Rashid et al. [162] demonstrate that 3D keypoints extracted from LiDAR point clouds enable a multi-task framework that simultaneously recognizes crossing actions and predicts trajectories on the Waymo Open Dataset. Emerging approaches leverage vision-language foundation models: MINDREAD [2] introduces cross-modal reasoning that jointly predicts intent and the textual reason behind it, improving accuracy by up to 7% on PIE, and Ahmed et al. [3] show that systematically refined prompts incorporating ego-vehicle dynamics and temporal context enable VLFMs to achieve up to 19.8% accuracy gains on JAAD, PIE, and FU-PIP. While crossing intent is a binary classification task, the broader lesson—that pose reveals intent before it manifests in motion—motivates our use of keypoint features for the more general trajectory forecasting problem.

4.2.3 Pose for Trajectory Prediction

Integrating pose into trajectory prediction has been relatively underexplored, though interest has grown rapidly in recent years. HST [178] demonstrated that 3D skeletal keypoints improve pedestrian trajectory prediction on JRDB, with pose proving most valuable for newly-detected agents with limited trajectory history. However, HST does not incorporate environmental context. Social-Transmotion [174] introduced the concept of promptable trajectory prediction, treating pose keypoints, bounding boxes, and ground-plane coordinates as interchangeable "prompts" that a transformer can flexibly consume; this work showed that pose prompts consistently improve accuracy across datasets and that the model gracefully degrades when prompts are unavailable. Social-Pose [175] introduced a decoupled pose encoder compatible with diverse trajectory prediction architectures, confirming that pose is a broadly valuable complementary signal that generalizes across architectures and datasets. DTDNet [213] combines pose-informed dynamic targets with graph-based social modeling for pedestrian trajectory prediction. Concurrently, Jeong and Jeon [83] proposed a multi-modal knowledge distillation framework that trains a teacher on pose, text, and trajectory modalities and distills this knowledge into a lightweight student that requires only trajectories at inference, demonstrating that pose information can improve predictions even when pose is unavailable at test time. SGNetPose+ [57] integrates skeleton keypoints and body segment angles into a goal-driven architecture for egocentric pedestrian trajectory prediction (WACV 2025), achieving state-of-the-art results on JAAD and PIE. Taketsugu et al. [197] propose Locomotion Embodiment (CVPR 2025), a framework that uses physics simulation to explicitly evaluate the physical plausibility of predicted trajectories conditioned on observed poses, demonstrating that pose-aware plausibility filtering improves even state-of-the-art methods. Kress et al. [105] further confirm the value of predicting trajectories using body pose representations rather than solely Cartesian locations. Closest to our approach, Kress et al. [104] combine body pose with semantic map information for probabilistic VRU trajectory forecasting, demonstrating the value of jointly leveraging pose and scene context; however, their method relies on hand-crafted pose features and a static semantic map rather than learned representations. Our keypoint encoder builds on these insights while additionally integrating scene understanding through a dedicated environment stream with learned BEV features. This combination of pose and dense environmental context is, to our knowledge, unique among existing methods, and as we show in Section 4.5, the two streams address complementary failure modes—pose reduces agent-agent collisions while BEV features reduce environment collisions—providing empirical justification for the three-stream design.

4.2.4 Scene Context Through BEV Representations

Environmental context, including sidewalks, crosswalks, road boundaries, and traversable surfaces, provides critical constraints on feasible pedestrian motion [98, 135]. Several recent methods explicitly incorporate scene features into pedestrian prediction: TSC-Net [90] discretizes the scene into traversability cells and predicts trajectories via cell classification, SceneAware [250] uses LLM-guided walkability maps to enforce scene constraints, and Chen et al. [33] integrate scene features within a sparse graph framework.

Online HD map methods construct intermediate Bird's Eye View (BEV) feature maps by projecting multi-sensor data into a top-down representation [74, 156], then decode them into vectorized polylines. The MapTR family [119, 120] established the dominant paradigm of permutation-equivalent point-set modeling, and subsequent work has rapidly advanced the field: PivotNet [43] introduces pivot-based representations for flexible element shapes, HIMap [257] combines point-level and element-level features through hybrid representation learning, MGMap [126] uses learned masks to highlight informative BEV regions, GeMap [130] leverages geometric priors to surpass 70% mAP on Argoverse 2, and ADMap [72] adds anti-disturbance mechanisms for robustness in complex scenes. StreamMapNet [241] extends the paradigm to temporal streaming with multi-point attention, Mask2Map [121] decodes maps from BEV segmentation masks, and MapExpert [28] distributes sparse experts across diverse map element types. P-MapNet [86] demonstrates that incorporating standard-definition map and HD map priors can improve online construction by up to 18.73 mIoU, while MapDistill [113] uses knowledge distillation to transfer camera-LiDAR fusion knowledge into lightweight camera-only models. While compact, the vectorized output of these methods discards dense semantic information such as surface type, vegetation, and building facades that is particularly relevant for pedestrian behavior prediction.

Recent work has begun to bridge online mapping and trajectory prediction more tightly. Gu et al. [61] showed that exposing map estimation uncertainty to downstream prediction models yields faster convergence and up to 15% better prediction performance. MapBEVPrediction [62] demonstrated that directly accessing intermediate BEV features, bypassing the vectorization bottleneck, improves trajectory prediction by preserving spatial and semantic context while achieving up to 73% faster inference. An emerging line of work pursues map-free trajectory prediction: MFTraj [248] uses behavior-aware graph convolutions to predict without HD maps, and BEVTraj [251] operates directly in BEV space using deformable attention over raw sensor features with sparse goal proposals. Most recently, Gu et al. [63] propose a self-supervised method for learning where map uncertainty is most beneficial for mapless prediction, and DiffSemanticFusion [30] enhances BEV representations by fusing semantic raster and graph-based map formats through an online HD map diffusion module, improving trajectory prediction by 5.1% on nuScenes. Diffusion-based approaches have also begun to incorporate scene context: TrajDiffuse [159] uses map-based guidance within a conditional diffusion model to generate environment-aware trajectories that respect scene constraints, and Leapfrog [137] accelerates diffusion-based trajectory prediction by learning an expressive prior that skips many denoising steps. In the multi-agent setting, hierarchical scene transformers [215] and cooperative trajectory representations [123] model scene-level context jointly with agent interactions. Our environment encoder adopts the BEV feature access principle: we extract agent-centric patches from pretrained BEV feature maps, encoding local scene semantics and spatial constraints that are especially informative in the semantically ambiguous pedestrian environments described above.

In summary, prior work has established that body pose improves trajectory prediction (Section 4.2.3) and that BEV scene features provide valuable environmental context, but no existing method integrates both signals alongside trajectory history. Methods that use pose lack environmental context; methods that use BEV features lack pose. Furthermore, integrating three heterogeneous modalities—trajectory dynamics, body keypoints, and dense BEV features—introduces fusion challenges that two-stream methods do not face, since the BEV signal is extracted from a separately pretrained network with a potential domain gap and encodes static scene context rather than dynamic agent state. PECT addresses both gaps: it is, to our knowledge, the first architecture to combine all three streams, and introduces a gated curriculum fusion strategy to manage the resulting modality heterogeneity.

4.3 PECT: Pose and Environment-Contextualized Transformer

4.3.1 Overview

We introduce PECT (Pose and Environment-Contextualized Transformer), a trajectory prediction framework that augments the HiVT architecture with two additional encoding streams: a *keypoint encoder* that captures fine-grained human pose information, and an *environment encoder* that incorporates rich contextual features from the surrounding scene via BEV representations.

Given a scene with N agents observed over T_{obs} timesteps, our goal is to predict the future trajectories of all agents over a prediction horizon T_{pred}. Let $\mathbf{X}_i = \{\mathbf{x}_i^t\}_{t=1}^{T_{\text{obs}}}$ denote the observed trajectory of agent i, where $\mathbf{x}_i^t \in \mathbb{R}^2$ represents the 2D position at time t. Our model predicts a set of K possible future trajectories $\{\hat{\mathbf{Y}}_i^{(k)}\}_{k=1}^K$ with associated probabilities $\{\pi_i^{(k)}\}_{k=1}^K$ for each agent.

The key insight of PECT is that trajectory prediction for vulnerable road users (pedestrians, cyclists) benefits from two complementary information sources that are typically ignored:

1. **Body pose**: Fine-grained keypoint information reveals intent signals such as head orientation, arm gestures, and body lean that precede motion changes.
2. **Environmental context**: BEV semantic features encode traversability, scene layout, and spatial affordances that constrain feasible motion.

The complete architecture processes three input streams in parallel—vectorized agent and map data, agent keypoints, and BEV environment features—before fusing them through concatenation and joint reasoning via a global interaction encoder.

4.3.2 Trajectory Backbone: HiVT

PECT builds upon HiVT (Hierarchical Vector Transformer) [258] as its trajectory encoding backbone. HiVT decomposes multi-agent motion prediction into two stages through a hierarchical architecture:

- **Local encoder**: Processes individual agent trajectories as sequences of displacement vectors through two sequential steps:
  1. *Agent-agent attention*: A graph transformer layer captures interactions for each timestep independently, where each agent-timestep attends to nearby agents of the same timestep within a spatial radius.
  2. *Temporal attention*: An autoregressive transformer layer captures cross-temporal information, where each timestep attends to past timesteps of the same agent.

  This factorization avoids the quadratic complexity of full all-to-all attention while still capturing both spatial and temporal dependencies. The vector-based representation explicitly encodes motion direction and magnitude, and all polylines are transformed into translation-invariant and rotation-invariant local coordinate frames centered on each agent. HiVT also includes a third parallel transformer layer to process agent-map information; MapBEVPrediction [62] replaces this layer with a ViT-based encoder that directly processes BEV feature maps, which we adopt as the environment stream in PECT (described in Section 4.3.5).
- **Global encoder**: Applies multi-head self-attention across all agents in the scene to capture long-range dependencies and collective dynamics that extend beyond the local neighborhood.
- **Multimodal future decoder**: An MLP receives the local and global representations as input and parameterizes the distribution of future trajectories as a Laplace mixture model. For each agent and each of F mixture components, the decoder outputs a predicted location $\boldsymbol{\mu} \in \mathbb{R}^2$ and associated uncertainty $\mathbf{b} \in \mathbb{R}^2$ per future timestep. A separate MLP followed by a softmax produces the mixing coefficients for each agent. Predictions are made for all agents in a single forward pass.

We chose HiVT as the backbone for its reputation as an established baseline in the AV trajectory forecasting community, its efficiency in handling scenes with many agents, and its modular design that naturally accommodates additional feature streams through concatenation before the global encoding stage.

4.3.3 Input Representation

PECT processes three complementary input modalities, each capturing different aspects of the scene:

Vectorized Agent and Map Data. Following HiVT, we represent agent trajectories as sets of polylines. Each agent's observed trajectory is encoded as a sequence of displacement vectors:

$$\mathbf{v}_i^t = \mathbf{x}_i^{t+1} - \mathbf{x}_i^t, \quad t \in \{1, \ldots, T_{\text{obs}} - 1\} \tag{4.1}$$

Agent Keypoints. For each agent i, we obtain a set of J body keypoints using SAM 3D Body [236]:

$$\mathbf{P}_i^t = \{\mathbf{p}_{i,j}^t\}_{j=1}^J, \quad \mathbf{p}_{i,j}^t = (x_{i,j}^t, y_{i,j}^t, z_{i,j}^t) \tag{4.2}$$

where (x, y, z) are the 3D coordinates of keypoint j. We use the 70 main body keypoints defined by SAM 3D Body's Momentum Human Rig (MHR) pose model (Figure 4.3), which includes major body joints (shoulders, elbows, wrists, hips, knees, ankles), hand and finger keypoints, and head keypoints (nose, eyes, ears) that are particularly informative for gaze and attention estimation. Keypoints are detected for agents with detection confidence $c > 0.5$ and matched to agent tracks via Hungarian algorithm on 2D bounding box IOU.

Environment Features. We extract BEV feature maps $\mathbf{B} \in \mathbb{R}^{H \times W \times C}$ from BEVFusion [129] pretrained on nuScenes, using features from the final observation frame $t = T_{\text{obs}}$. The BEV representation encodes the scene as a top-down grid of spatial resolution $H \times W$, where C feature channels capture semantic information such as drivable area, sidewalks, crosswalks, and vegetation.

4.3.4 Keypoint Encoder

The keypoint encoder transforms raw body keypoints into a compact representation that captures pose configuration over time. We model the human body as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

where nodes $\mathcal{V}$ correspond to keypoints and edges $\mathcal{E}$ represent anatomical connections (e.g., shoulder-elbow, hip-knee).

Spatial Encoding. Each keypoint is first embedded via a linear projection:

$$\mathbf{h}_{i,j}^{t,(0)} = \text{MLP}_{\text{kp}}\left([\mathbf{p}_{i,j}^{t}; c_{i,j}^{t}]\right) \in \mathbb{R}^{d} \tag{4.3}$$

where $[\cdot;\cdot]$ denotes concatenation and d is the hidden dimension. We then apply L layers of graph neural network message passing to capture spatial relationships between body parts:

$$\mathbf{h}_{i,j}^{t,(\ell+1)} = \mathbf{h}_{i,j}^{t,(\ell)} + \text{ReLU}\left(\sum_{k\in\mathcal{N}(j)} \frac{1}{|\mathcal{N}(j)|}\mathbf{W}^{(\ell)}\mathbf{h}_{i,k}^{t,(\ell)}\right) \tag{4.4}$$

where $\mathcal{N}(j)$ denotes the neighbors of keypoint j in the skeleton graph and $\mathbf{W}^{(\ell)} \in \mathbb{R}^{d\times d}$ are learnable weights.

Temporal Encoding. After spatial encoding, we aggregate keypoint features across the body and apply temporal attention to capture pose dynamics over the observation window. First, we compute a per-frame body representation via mean pooling:

$$\mathbf{b}_{i}^{t} = \frac{1}{J}\sum_{j=1}^{J}\mathbf{h}_{i,j}^{t,(L)} \tag{4.5}$$

We then apply multi-head self-attention across timesteps to model temporal dependencies:

$$\tilde{\mathbf{b}}_{i}^{1:T_{\text{obs}}} = \text{TemporalAttn}\left(\mathbf{b}_{i}^{1:T_{\text{obs}}}\right) \tag{4.6}$$

The final keypoint embedding for agent i is obtained by taking the representation at the last observed timestep:

$$\mathbf{e}_{i}^{\text{kp}} = \tilde{\mathbf{b}}_{i}^{T_{\text{obs}}} \in \mathbb{R}^{d} \tag{4.7}$$

This design enables the model to learn pose-based intent signals such as a pedestrian turning their head toward the road before crossing, or a cyclist leaning into a turn.

4.3.5 Environment Encoder

The environment encoder extracts scene context from BEV feature maps. Following MapBEVPrediction [62], we directly access intermediate BEV representations rather than vectorized map outputs, preserving rich spatial and semantic information that would otherwise be lost during decoding. The design is agnostic to the specific BEV backbone, supporting any pretrained mapping or camera-LiDAR fusion model (e.g., BEVFusion, MapTRv2).

Pretrained BEV Map Model

We use BEVFusion [118, 129] pretrained on nuScenes for semantic BEV map segmentation as our BEV feature source. BEVFusion projects multi-camera images and LiDAR point clouds into a shared BEV space through learned depth estimation and voxelization, then fuses the two modalities via learned convolutions. We train the model to produce a feature map of size $(H, W, C) = (200, 200, 256)$ covering a 30m$\times$$30$m area around the ego-vehicle at a resolution of 0.15m per pixel. Rather than decoding this representation into a vectorized map, we extract the intermediate BEV features directly for use in the environment encoder below.

Notably, BEVFusion is pretrained on nuScenes [18], a vehicle-centric driving dataset collected from a car-mounted sensor suite, whereas we apply it to JRDB [138], a pedestrian-centric dataset collected from a ground-level mobile robot on a university campus. This introduces a domain gap across several dimensions: sensor configuration (roof-mounted cameras and LiDAR vs. robot-height sensors with different fields of view and point cloud densities), spatial scale and layout (wide roads, intersections, and highways vs. sidewalks, plazas, and building interiors), and semantic categories (the road-centric ontology of nuScenes, such as drivable surface, lane dividers, and barriers, differs from the pedestrian-relevant categories in JRDB such as grass, benches, doorways, and building facades). We deliberately adopt this cross-domain setup for two reasons. First, it tests whether intermediate BEV features transfer useful spatial and geometric structure even when the source domain's semantic categories do not perfectly align with the target environment—a practically important question, since pretrained BEV models are far more readily available for driving domains than for pedestrian-scale settings. Second, the gated curriculum fusion mechanism (Section 4.3.5) is explicitly designed to handle noisy or partially misaligned environment features: the learned gate can suppress BEV channels that are uninformative or misleading for a given agent, and the epoch-level on-ramp prevents the domain-shifted signal from disrupting early training. We discuss the implications and limitations of this domain gap in Section 4.6.

Agent-Centric Feature Extraction and Encoding

Given the global BEV feature map $\mathbf{B} \in \mathbb{R}^{H\times W\times C}$ at the final observation frame, we first transform it into an agent-centric representation for each agent. For agent i at position $\mathbf{x}_i^{T_{\text{obs}}}$, the BEV map is rotated and cropped to align with the agent's heading, producing an agent-frame feature map $\mathbf{B}_i$. Since the trajectory and keypoint representations are also expressed in agent-centric coordinates, this rotation-aligned transformation maintains consistency across all three input streams and preserves translation and rotation equivariance. Agents whose positions fall outside the BEV perception range are assigned zero embeddings.

The agent-centric BEV map is then processed through a Vision Transformer (ViT). The feature map is divided into a regular grid of non-overlapping patches, and each patch is flattened and linearly projected to produce a sequence of patch tokens. Learned positional embeddings are added to encode spatial location within the grid. The patch token corresponding to the agent's center position serves as the *query* in a cross-attention operation, attending to all other patch tokens as *keys* and *values*. This allows the model to selectively aggregate environmental context from the surrounding area while anchoring the representation at the agent's location. The cross-attention is followed by an optional feed-forward network, with residual connections at each stage. Multiple transformer layers are stacked to enable progressive refinement. An attention mask zeros out patches that fall outside the valid BEV area (e.g., regions introduced by rotation), preventing the model from attending to invalid content.

After the transformer layers, the patch representations are mean-pooled and projected through a linear head to produce the final environment embedding:

$$\mathbf{e}_i^{\text{env}} = \text{MLP}_{\text{head}}\left(\text{MeanPool}\left(\text{ViT}(\mathbf{B}_i)\right)\right) \in \mathbb{R}^d \tag{4.8}$$

The environment embedding $\mathbf{e}_i^{\text{env}}$ encodes local semantic information such as whether the agent is on a sidewalk, approaching a crosswalk, or near a wall—contextual cues that inform feasible future motion.

4.3.6 Feature Fusion and Global Interaction

The three encoding streams produce complementary representations for each agent: the HiVT local encoder output $\mathbf{e}_i^{\text{hvt}} \in \mathbb{R}^d$ capturing trajectory and map context, the keypoint embedding $\mathbf{e}_i^{\text{kp}} \in \mathbb{R}^d$ encoding body pose, and the environment embedding $\mathbf{e}_i^{\text{env}} \in \mathbb{R}^d$ providing scene context.

Gated Environment Fusion. Fusing three fundamentally different modalities—agent trajectories, body pose, and dense environmental semantics—poses a challenge that prior work has not had to confront. Existing methods operate with at most two of these streams: trajectory-only architectures like HiVT [258] and MTR [186] need no fusion at all; pose-augmented methods like HST [178] and Social-Pose [175] concatenate pose with trajectory but do not incorporate environment; and BEV-augmented methods like MapBEVPrediction [62] add scene context to trajectory without pose. In the two-stream case, naïve concatenation or addition suffices because the supplementary modality is typically well-aligned with trajectory in terms of training signal strength and reliability. Adding a third modality breaks this assumption: the BEV environment stream differs qualitatively from trajectory and pose in several ways—it is extracted from a pretrained network with a potential domain gap, it encodes static scene context rather than dynamic agent state, and its informativeness varies sharply across agents (e.g., highly relevant for an agent near a wall, largely redundant for one in open space). Simply concatenating all three streams risks the environment signal either overwhelming the well-calibrated trajectory and pose representations or, conversely, being drowned out entirely.

To address this, we apply a learned gating mechanism specifically to the environment stream:

$$\mathbf{g}_i = \sigma\left(\mathbf{W}_g[\mathbf{e}_i^{\text{hvt}}; \mathbf{e}_i^{\text{env}}] + \mathbf{b}_g\right) \tag{4.9}$$

$$\tilde{\mathbf{e}}_i^{\text{env}} = \alpha(e) \cdot \mathbf{g}_i \odot \mathbf{e}_i^{\text{env}} \tag{4.10}$$

where σ is the sigmoid function, $\mathbf{W}_g \in \mathbb{R}^{d\times 2d}$ and $\mathbf{b}_g \in \mathbb{R}^d$ are learnable parameters, and $\odot$ denotes element-wise multiplication. The scalar $\alpha(e) = \min(e/E_{\text{ramp}},\ 1)$ linearly scales the gated environment features from fully inactive ($\alpha = 0$) to fully active ($\alpha = 1$) over the first E_{ramp} training epochs, where e denotes the current epoch. This on-ramping schedule implements a form of curriculum learning: by initially zeroing out the environment stream, the model first learns basic pedestrian dynamics from trajectory and pose alone, establishing a strong motion prior before gradually incorporating the more complex BEV features. The learned gate $\mathbf{g}_i$ then provides fine-grained, instance-level control, suppressing environment features when they are uninformative or noisy for a particular agent.

Concatenation and Global Encoding. The agent embedding, keypoint embedding, and gated environment embedding are concatenated to form a unified agent representation:

$$\mathbf{z}_i = [\mathbf{e}_i^{\text{hvt}}; \mathbf{e}_i^{\text{kp}}; \tilde{\mathbf{e}}_i^{\text{env}}] \in \mathbb{R}^{3d} \tag{4.11}$$

A linear projection maps this back to dimension d:

$$\mathbf{z}_i' = \mathbf{W}_{\text{proj}}\mathbf{z}_i + \mathbf{b}_{\text{proj}} \in \mathbb{R}^d \tag{4.12}$$

The projected features $\{\mathbf{z}_i'\}_{i=1}^N$ are then passed through HiVT's global interaction encoder (Section 4.3.2), a graph transformer that models relationships between all agents in the scene. The global encoder applies multi-head self-attention over the set of agents:

$$\{\mathbf{z}_i^{\text{global}}\}_{i=1}^N = \text{GlobalTransformer}\left(\{\mathbf{z}_i'\}_{i=1}^N\right) \tag{4.13}$$

This enables each agent's representation to incorporate information about other agents' states, poses, and environmental contexts, facilitating reasoning about multi-agent interactions such as yielding behavior and social navigation.

MLP Decoder. The global agent representations are passed to HiVT's multimodal MLP decoder (Section 4.3.2), which produces K trajectory modes per agent. For each agent i and mode k, the decoder outputs a sequence of predicted positions and associated uncertainties:

$$\hat{\mathbf{Y}}_i^{(k)} = \{(\hat{\mathbf{y}}_{i,t}^{(k)}, \hat{\boldsymbol{\sigma}}_{i,t}^{(k)})\}_{t=1}^{T_{\text{pred}}}, \quad \hat{\mathbf{y}}_{i,t}^{(k)} \in \mathbb{R}^2, \quad \hat{\boldsymbol{\sigma}}_{i,t}^{(k)} \in \mathbb{R}^2 \tag{4.14}$$

where $\hat{\mathbf{y}}_{i,t}^{(k)}$ is the predicted 2D position and $\hat{\boldsymbol{\sigma}}_{i,t}^{(k)}$ parameterizes a Laplace distribution over position uncertainty. A separate MLP head followed by softmax produces mode probabilities $\{\pi_i^{(k)}\}_{k=1}^K$, giving a full Laplace mixture model over future trajectories for each agent. Predictions are made for all agents in a single forward pass.

4.3.7 Training Objective

We train PECT end-to-end using the same multi-task objective as HiVT [258], combining trajectory regression and mode classification.

Regression Loss. The regression loss uses a winner-takes-all strategy: for each agent, the best mode is identified based on L2 distance to ground truth, and a Laplace negative log-likelihood loss is applied to that mode:

$$k_i^* = \underset{k}{\operatorname{argmin}} \sum_{t=1}^{T_{\text{pred}}} \|\hat{\mathbf{y}}_{i,t}^{(k)} - \mathbf{y}_{i,t}^*\|_2 \cdot m_{i,t} \tag{4.15}$$

$$\mathcal{L}_{\text{reg}} = \frac{1}{\sum_i \sum_t m_{i,t}} \sum_{i=1}^{N} \sum_{t=1}^{T_{\text{pred}}} m_{i,t} \cdot \text{NLL}_{\text{Laplace}} \left(\hat{\mathbf{y}}_{i,t}^{(k_i^*)}, \hat{\sigma}_{i,t}^{(k_i^*)}, \mathbf{y}_{i,t}^* \right) \tag{4.16}$$

where $m_{i,t} \in \{0, 1\}$ masks invalid timesteps and the Laplace NLL jointly optimizes position accuracy and uncertainty calibration.

Classification Loss. Soft targets based on negative L2 distances encourage mode probabilities to reflect prediction quality:

$$\mathcal{L}_{\text{cls}} = -\frac{1}{N_{\text{valid}}} \sum_{i:V_i>0} \sum_{k=1}^{K} \tilde{\pi}_i^{(k)} \log \pi_i^{(k)} \tag{4.17}$$

where $\tilde{\pi}_i^{(k)}$ are softmax-normalized negative distances and $V_i = \sum_t m_{i,t}$ is the number of valid timesteps.

Total Loss. The final objective combines both losses with equal weighting: $\mathcal{L} = \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{cls}}$.

4.4 Experimental Setup

4.4.1 JRDB Dataset

We evaluate PECT on the JRDB (JackRabbot Dataset and Benchmark) [138], a large-scale pedestrian-centric dataset collected by the JackRabbot mobile robot navigating indoor and outdoor environments across Stanford University's campus. JRDB provides 64 minutes of multi-modal sensor data, including 360-degree panoramic RGB video and 3D LiDAR point clouds, with over 2.4 million bounding box annotations and 3,500+ time-consistent trajectories spanning diverse pedestrian density scenarios. The dataset's pedestrian-centric perspective, multi-modal sensor suite, and availability of panoptic segmentation labels make it well-suited for evaluating methods that leverage both visual pose information and environmental context. We refer readers to [138] for complete dataset details.

4.4.2 Evaluation Metrics

We evaluate using both marginal and joint trajectory forecasting metrics, as introduced in Chapter 3. **Marginal metrics** (min*ADE*, min*FDE*) evaluate each agent independently by selecting the best of $K = 6$ predicted modes per agent. **Joint metrics** (*JADE*, *JFDE*) require selecting a single mode for all agents simultaneously, providing a more realistic assessment of multi-agent prediction quality (see Chapter 3 for the full motivation and definitions).

Agent-Agent Collision Rate (ACR). In Chapter 3, we introduced this metric as simply CR (Collision Rate), since agent-agent collisions were the only collision type under consideration. Now that we additionally evaluate environment collisions (ECR, below), we rename it to ACR (Agent-Agent Collision Rate) to disambiguate the two. ACR measures the proportion of predicted trajectories that result in collisions between agents, computed on the highest-likelihood mode (the mode with the maximum π value) using an agent radius of 0.2 meters. By evaluating the model's most confident prediction rather than the best-of-K sample, ACR and ECR avoid the over-optimism of oracle selection, which can mask collision-prone behavior by always choosing the luckiest mode.

Environment Collision Rate (ECR). To assess whether predicted trajectories respect environmental constraints, we evaluate an agent-environment collision metric. Prior work has introduced metrics for this purpose: Sohn et al. [191] proposed Environment Collision-Free Likelihood (ECFL) in A2X, which tests whether an agent's center-of-mass trajectory intersects non-navigable cells in a binary navigability map (1 = navigable, 0 = non-navigable). ECFL was subsequently adopted by later forecasting methods such as MUSE-VAE [107]. We use the name ECR (Environment Collision Rate) to distinguish our metric from ECFL; whereas ECFL reports the likelihood of being collision-*free* (higher is better), ECR reports the collision *rate* (lower is better), i.e., $\text{ECR} = 1 - \text{ECFL}$. We adopt the collision-rate framing for both ACR and ECR to maintain a uniform "lower is better" convention across all metrics, consistent with the displacement errors (minADE, minFDE, JADE, JFDE) that dominate trajectory forecasting evaluation.

A key distinction of our approach from that of ECFL is that it does not rely on a pre-labeled binary navigability map. Instead, we leverage the availability of panoptic segmentation labels in the JRDB dataset to construct a semantically-labeled 3D point cloud of the static environment. Specifically, we (1) take JRDB's panoptic 2D camera segmentation labels, (2) project these labels onto the corresponding 3D LiDAR point cloud, and (3) filter for points belonging to certain semantic categories such as ”wall”, ”window”, and other barrier-related categories. We complete an additional heuristic filtering step to clean up the resulting ”wall” point cloud, including filtering out mislabelled points that actually belong to humans using their annotated 3D bounding boxes. Since doorways introduce gaps in the upper portions of walls, we filter for low-lying points to ensure robust wall detection. To account for elevation changes across larger scenes, we tile the scene into 5m × 5m cells in the x-y plane and, within each cell, retain only points in the lower half of the z-range. Given this filtered point cloud, an agent's predicted trajectory is marked as colliding with the environment if any timestep of its highest-likelihood mode (the mode with the maximum π value) falls within 0.05m (5 cm) of at least one wall-category point. The ECR score is the proportion of agents satisfying this condition across the evaluation set.

Because this procedure operates directly on labeled point clouds rather than requiring a hand-annotated pixel map, it generalizes to any dataset with segmentation annotations; in Section 5.5, we discuss how foundation models could automate this labeling step entirely, removing the dependency on dataset-provided annotations. We currently evaluate collisions

Table 4.1: Trajectory prediction on JRDB with $K = 6$ modes. *ADE* / *FDE* and *JADE* / *JFDE* in meters; ACR uses 0.2m agent radius; ECR uses 0.05m wall threshold. Prediction horizon is 4s at 3 Hz (12 waypoints). Percentages indicate relative change from the baseline. Lower is better for all metrics.

Method	**ADE / FDE ↓**	**JADE / JFDE ↓**	**ACR ↓**	**ECR ↓**
HiVT [258]	0.23 / 0.40	0.36 / 0.70	0.020	0.041
+ Keypoints	0.22 (−0.7%) / 0.40 (−1.6%)	0.36 (−0.66%) / 0.69 (−0.86%)	0.019 (−4.6%)	0.039 (−3.4%)
+ BEV [62]	0.23 (+2%) / 0.40 (+2.6%)	0.36 (+1.2%) / 0.69 (+1.6%)	0.019 (−4.2%)	0.038 (−7.9%)
PECT (Ours)	**0.22** (−1.4%) **/ 0.40** (−1%)	**0.36** (−0.94%) **/ 0.69** (−0.56%)	**0.018** (−6.2%)	**0.038** (−8.1%)

Table 4.2: Same setup as Table 4.1, filtered for agents whose last observation step is within 40 cm of another agent. Percentages indicate relative change from the baseline. Lower is better for all metrics.

Method	**ADE / FDE ↓**	**JADE / JFDE ↓**	**ACR ↓**	**ECR ↓**
HiVT [258]	0.13 / 0.21	0.21 / 0.38	0.025	0.024
+ Keypoints	0.12 (−2.1%) / 0.21 (−2%)	0.20 (−1.8%) / 0.38 (−1.4%)	0.021 (−15%)	0.025 (+3.3%)
+ BEV [62]	0.13 (−0.43%) / 0.21 (−1.1%)	0.21 (+0.18%) / 0.38 (+0.16%)	0.023 (−8.7%)	0.024 (+2%)
PECT (Ours)	**0.12** (−2%) **/ 0.21** (−1.5%)	**0.21** (+0.34%) **/ 0.39** (+1%)	**0.021** (−15%)	**0.023** (−3.9%)

against wall-related categories; restricting to these categories may undercount collisions with other static obstacles such as furniture or vegetation, so the current ECR should be interpreted as a lower bound on the total environment collision rate. The 5 cm wall threshold is a pragmatic choice: smaller thresholds risk false positives from LiDAR calibration errors and label noise, while larger thresholds may miss near-wall violations. As with the agent radius sensitivity discussed in Chapter 3, relative rankings between methods are expected to be more stable across threshold choices than absolute metric values. ECR provides a useful complementary signal to ACR, capturing a distinct failure mode, environmental implausibility, that trajectory-only metrics cannot detect.

4.4.3 Evaluation Protocol

Following the evaluation practice used by Salzmann et al. [178], we use an observation horizon of $T_{\text{obs}} = 7$ frames (2.33 seconds at 3 Hz after downsampling from the native 15 Hz) and predict future trajectories over $T_{\text{pred}} = 12$ frames (4 seconds). In contrast to Salzmann et al. [178], we evaluate on the official JRDB test split. We do experiments with

Table 4.3: Same setup as Table 4.1, filtered for agents whose last observation step is within 20 cm of a wall. Percentages indicate relative change from the baseline. Lower is better for all metrics.

Method	**ADE / FDE ↓**	**JADE / JFDE ↓**	**ACR ↓**	**ECR ↓**
HiVT [258]	0.18 / 0.31	0.28 / 0.54	0.025	0.055
+ Keypoints	0.18 (–0.59%) / 0.31 (–0.71%)	0.28 (–0.72%) / 0.54 (–0.66%)	0.020 (–20%)	0.052 (–6.1%)
+ BEV [62]	0.18 (–0.37%) / 0.31 (–1.1%)	0.29 (+0.42%) / 0.54 (+0.29%)	0.021 (–14%)	0.050 (–9.9%)
PECT (Ours)	**0.18** (–1.7%) / **0.30** (–1.3%)	**0.28** (–0.24%) / **0.54** (+0.14%)	**0.022** (–12%)	**0.049** (–10%)

Table 4.4: ADE comparison between PECT (ours) and HST [178] on JRDB with $K = 6$ modes. We retrain HST on the official JRDB split for fair comparison; results differ from the original paper, which used a custom split. Lower is better.

Input features	**HiVT (Ours)**	**HST** [178]
Position only	0.23	0.33
+ Pose	**0.22**	0.25

different numbers of modes K: one with $K = 6$ trajectory modes per agent, and one with $K = 3$ modes to test less-optimistic scenarios.

4.5 Discussion

4.5.1 Improvements in Collision Metrics

PECT improves both agent-agent collision rate (ACR) and environment collision rate (ECR) compared to the HiVT baseline and single-modality ablations (Table 4.1). With $K = 6$ modes, PECT reduces ACR by 6.2% and ECR by 8.1% relative to the position-only baseline. Notably, keypoints alone yield the largest ACR reduction, while the environment stream contributes the largest ECR improvement, suggesting that the two modalities address complementary failure modes.

These improvements are especially pronounced under targeted filtering. In keeping with the evaluation philosophy from earlier chapters of testing on difficult dataset subsets, we filter for agents whose last observed position is within 40 cm of another agent (Table 4.2). The ACR improvement is amplified in this near-agent subset, while other metrics remain steady. This highlights an important point introduced earlier: when evaluating on aggregate metrics dominated by routine scenarios, improvements in difficult interaction-heavy cases can be diluted or hidden. By isolating harder subsets, the advantage becomes clearer.

Similarly, when filtering for agents whose last observed position is within 20 cm of a wall (Table 4.3), the ECR improvement is particularly large, confirming that the environment encoder successfully captures proximity to structural boundaries and produces trajectories that better respect physical constraints.

4.5.2 Maintaining Standard Metric Performance

These collision improvements appear not to come at the cost of standard displacement metrics. As shown in Table 4.1, PECT's *ADE* / *FDE* and *JADE* / *JFDE* remain comparable to or slightly better than the baseline (e.g., −1.4% *ADE* and −1% *FDE* at $K = 6$). Table 4.4 further contextualizes these results against HST [178], the most directly comparable method that also uses skeletal keypoints on JRDB: the HiVT backbone substantially outperforms HST across all input configurations, and adding environment and pose features yields the best *ADE* overall. While these displacement improvements are modest, the key observation is that the collision gains do not appear to degrade displacement accuracy, suggesting that the additional modalities may provide complementary information rather than introducing noise. We note, however, that the displacement differences are small enough that they could partly reflect run-to-run variance; further repetitions would be needed to confirm statistical significance. Indeed, the displacement improvements are modest enough that the BEV stream could partly function as a regularizer, adding parameters and an auxiliary information source that smooths optimization, rather than providing genuinely new spatial information to the model. However, two observations argue against a pure regularization explanation: first, ECR improves disproportionately (7.9–9.9%) compared to displacement metrics, and second, the ECR improvement is concentrated in the subset of agents whose initial positions are near walls and obstacles (as noted above). If the BEV stream were merely regularizing, we would expect uniform improvement across all agents and metrics rather than targeted improvement on environment-proximate predictions. That said, a definitive test would require comparing against a control condition using noisy or spatially shuffled BEV features; if gains persisted with uninformative BEV input, that would confirm a regularization effect. We leave this ablation for future work.

Regardless of whether the BEV stream's benefit is partly regularization or entirely spatial, maintaining displacement accuracy while adding two new modalities is non-trivial. The gated curriculum fusion strategy likely plays a role in this result, and reflects a challenge that may be unique to PECT's three-modality setting. Prior methods that augment trajectories with a single additional stream, whether pose [175, 178] or BEV features [62], can rely on simple concatenation because the auxiliary signal is relatively homogeneous with trajectory in training dynamics. PECT, by contrast, must integrate a third modality (dense BEV features) that differs fundamentally from the other two: it is extracted from a separately pretrained network with a potential domain gap, encodes static scene structure rather than dynamic agent state, and varies widely in relevance across agents. Naïvely fusing all three streams from the start risks the environment signal interfering with the well-calibrated trajectory and pose learning.

The curriculum on-ramping schedule addresses this by linearly scaling the gated BEV features from inactive to fully active over the first several training epochs, so the model first learns basic pedestrian behavior from pose and trajectory, establishing a reliable motion prior, before gradually incorporating the more complex environmental features. This staged approach is intended to prevent the noisy or domain-shifted BEV signal from disrupting early-stage trajectory learning, and to ensure that environment context is layered on top of an already-competent base model rather than competing with it from the start. The learned gate then provides additional instance-level control, suppressing environment features for individual agents when they are uninformative or unreliable. Together, the epoch-level curriculum and instance-level gating form a two-tier fusion strategy that appears to make three-modality integration tractable where simpler approaches might degrade performance.

4.6 Summary

This chapter presented PECT, a three-stream trajectory prediction framework that jointly integrates body pose and dense environmental semantics with trajectory history—at the time of this work, to our knowledge, the first pedestrian prediction method to combine all three modalities in a unified architecture with learned representations. PECT introduces the environment collision rate (ECR) metric for detecting predicted trajectories that violate physical scene boundaries, and a gated curriculum fusion strategy that makes three-modality integration tractable by on-ramping the environment stream gradually during training.

Experiments on JRDB show that the pose and environment streams address complementary failure modes: keypoints primarily reduce agent-agent collisions, while BEV features primarily reduce environment collisions, and their combination improves both without degrading displacement accuracy.

PECT validates the layered structure of this thesis, and the results tables make this dependency concrete. If PECT were evaluated using only the marginal displacement metrics that dominated the field prior to Chapter 3, its primary contribution would be invisible. The *ADE*/*FDE* improvements over the HiVT baseline are modest—on the order of 1% (Table 4.1)—small enough to be dismissed as noise. By contrast, the collision metrics reveal substantial gains: ACR improves by 6% and ECR by 8%. When filtering for near-wall agents (Table 4.3), the ECR improvement is amplified further. PECT's core value proposition—producing trajectories that respect both social and physical constraints—is entirely captured by collision-aware and joint metrics and entirely missed by standard marginal displacement evaluation. This directly demonstrates the upward dependency in the thesis pyramid: the evaluation layer (Chapter 3) is a prerequisite for the methods layer to demonstrate its contributions.

The relationship also flows downward: PECT's need for environment-aware assessment motivated ECR, and its multi-modal design reveals which data modalities future collection efforts should prioritize. Without the data foundation motivating richer sensor inputs (Chapter 2), the case for multi-modal prediction would be weaker.

Future directions. PECT's environment stream currently uses a BEV backbone pretrained on nuScenes (vehicle-centric driving) but evaluated on JRDB (pedestrian-centric campus), introducing a domain gap across sensor platform, environment type, and semantic ontology. That gains persist despite this mismatch establishes a lower bound; domain-adapted pretraining or fine-tuning on pedestrian-centric data would likely strengthen the environment stream. The intermediate BEV features we extract encode lower-level geometric structure, such as occupancy patterns, depth discontinuities, and object boundaries, that transfers more readily than semantic labels, and the gated curriculum fusion is designed to suppress uninformative or noisy channels. Because PECT's modular architecture treats each encoder as an interchangeable component, swapping in an in-domain BEV backbone requires changes only to the feature extraction step. Similarly, advances in monocular 3D pose estimation would directly benefit the keypoint stream, and extending ECR beyond wall categories to include furniture, vegetation, and other obstacles would provide a more comprehensive measure of environmental plausibility.

Chapter 5 synthesizes findings across all three layers, discusses cross-cutting themes including the relationship between multi-modal features and the long-tail data problem, and identifies open problems that will shape the next generation of forecasting systems.

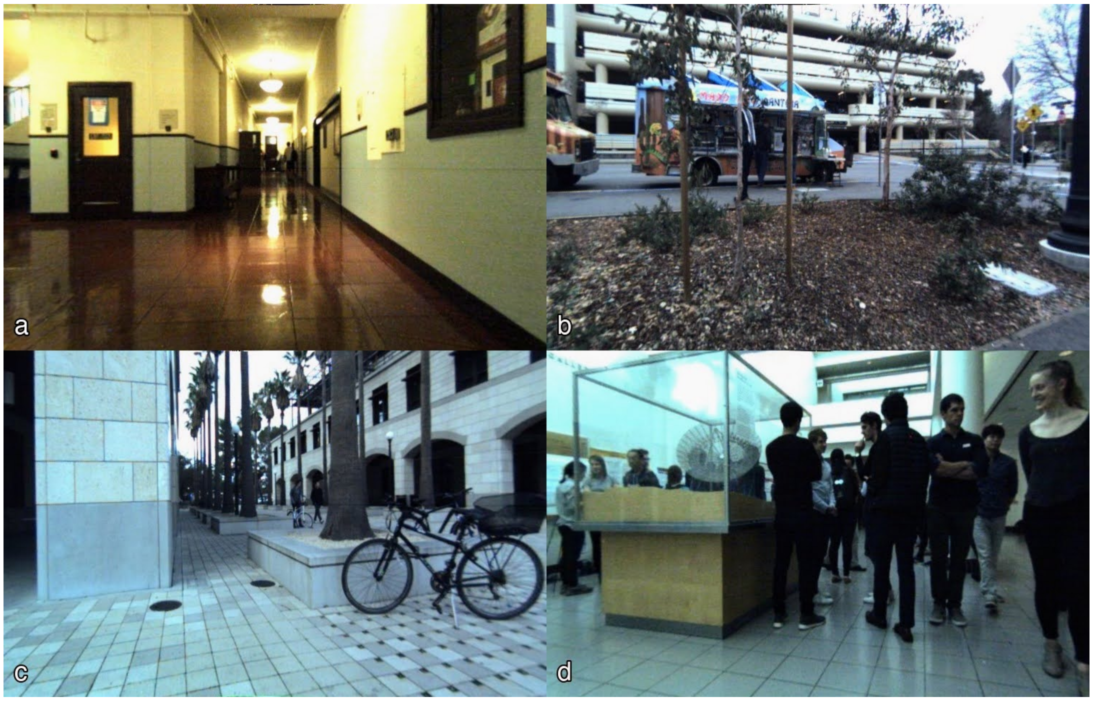


Figure 4.1: Human-centric environments exhibit semantic ambiguities that are largely absent in vehicle-centric scenarios: **(a)** Doorways may visually resemble impassable walls, yet pedestrians interact with them purposefully and pass through them. **(b)** Mulched areas are physically traversable but socially discouraged; thus, traversability is governed by a soft constraint rather than a hard boundary as in road geometry. **(c)** Objects exhibit context-dependent affordances: although a tree and a bicycle may both appear as obstacles to sensors, pedestrians can interact with bicycles while generally avoiding trees. **(d)** Dense crowds lack explicit structural organization, in contrast to vehicles, which remain lane-structured even under high-density conditions. Incorporating human body pose and environmental scene features can help resolve these ambiguities in trajectory prediction for human-centric environments.

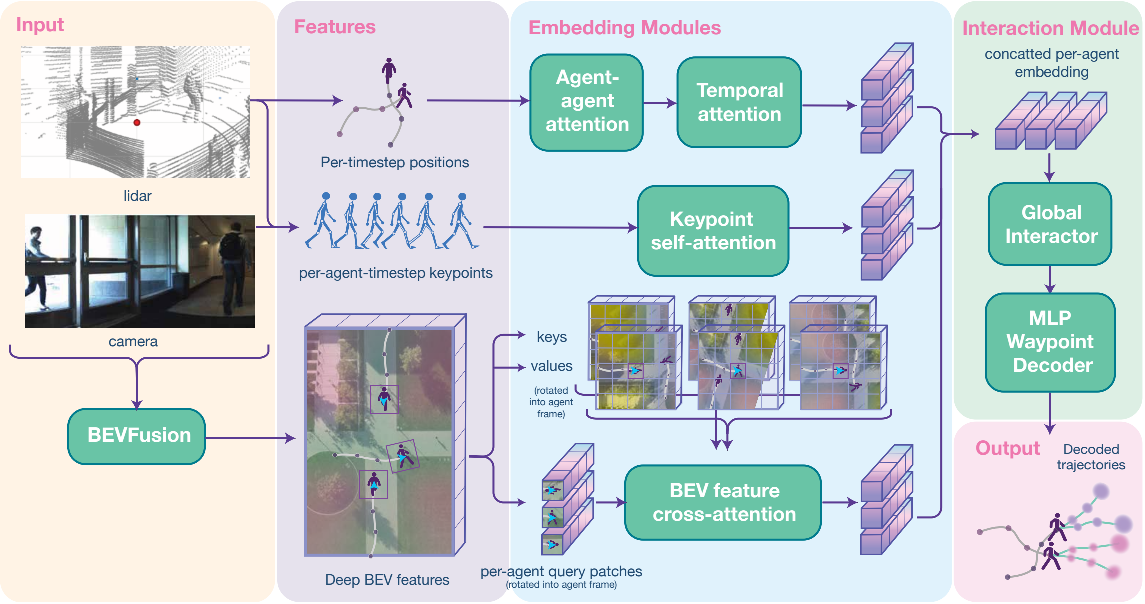


Figure 4.2: Overview of the PECT architecture. Camera and LiDAR inputs are processed by BEVFusion to produce deep BEV features. Three parallel embedding modules encode complementary modalities: (1) per-timestep agent positions through agent-agent attention and temporal attention, (2) per-agent-timestep body keypoints through keypoint self-attention, and (3) agent-centric BEV query patches (rotated into each agent's frame) through cross-attention over the BEV feature map keys and values. The resulting per-agent embeddings are concatenated and passed to a global interactor for scene-level reasoning, followed by an MLP waypoint decoder that outputs the predicted trajectories.

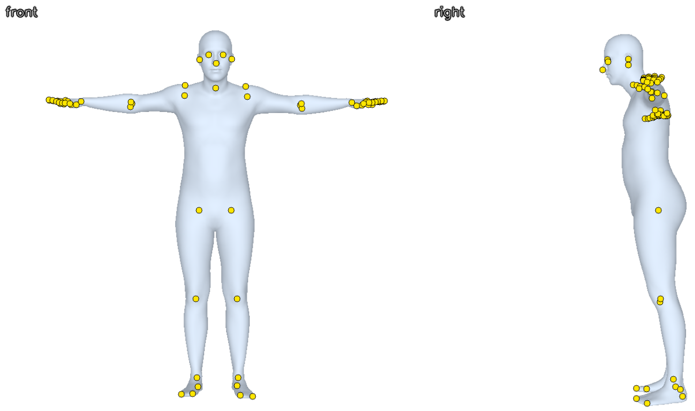
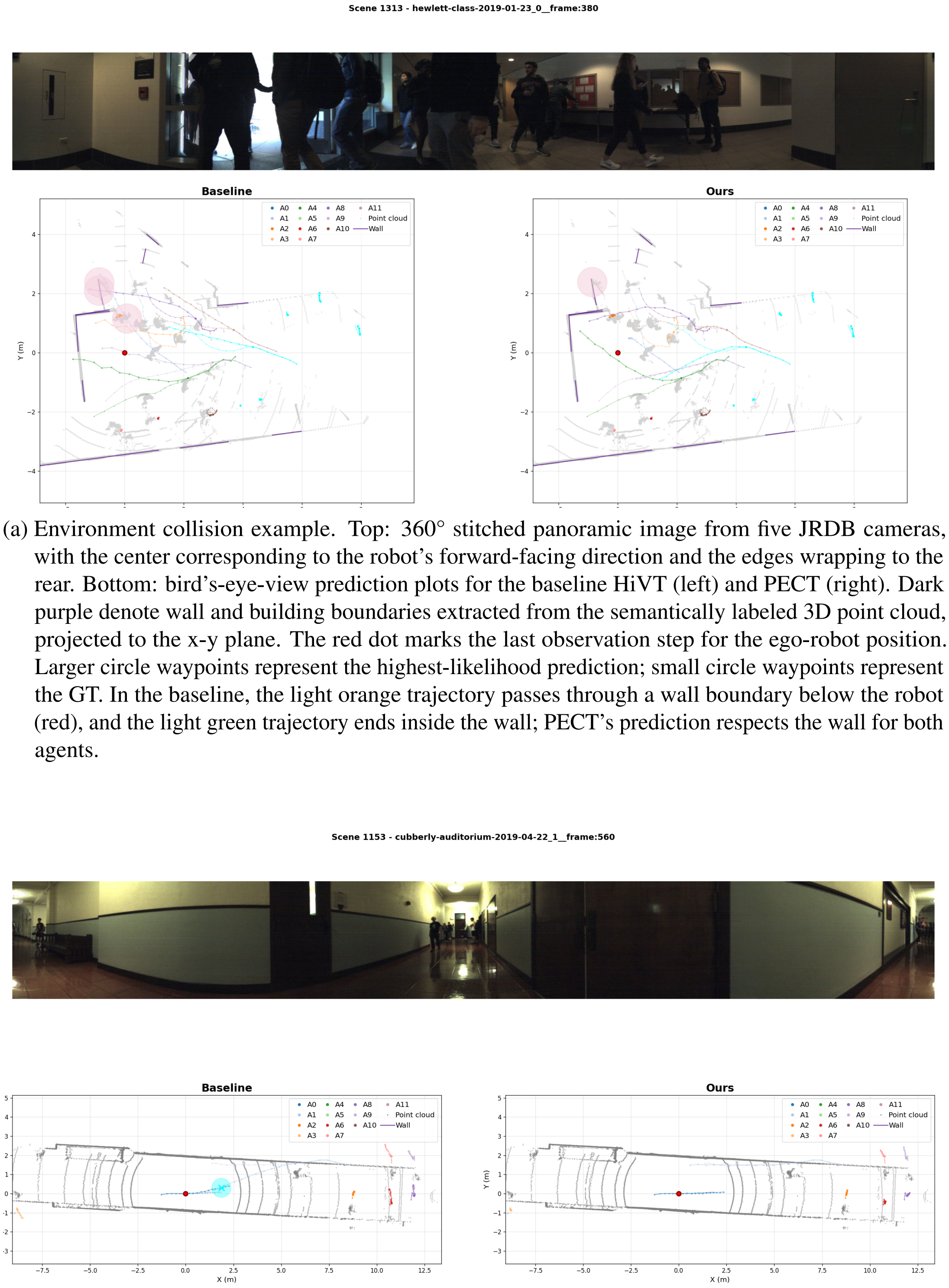


Figure 4.3: The 70 body keypoints from SAM 3D Body's Momentum Human Rig (MHR) pose model, shown on a standard pose.



(a) Environment collision example. Top: 360° stitched panoramic image from five JRDB cameras, with the center corresponding to the robot's forward-facing direction and the edges wrapping to the rear. Bottom: bird's-eye-view prediction plots for the baseline HiVT (left) and PECT (right). Dark purple denote wall and building boundaries extracted from the semantically labeled 3D point cloud, projected to the x-y plane. The red dot marks the last observation step for the ego-robot position. Larger circle waypoints represent the highest-likelihood prediction; small circle waypoints represent the GT. In the baseline, the light orange trajectory passes through a wall boundary below the robot (red), and the light green trajectory ends inside the wall; PECT's prediction respects the wall for both agents.

(b) Agent-agent collision example (same visual format as above). The baseline predicts trajectories for the robot and a nearby pedestrian that intersect, resulting in an agent-agent collision (left). PECT produces socially plausible, collision-free trajectories for the robot and that agent (right).

Figure 4.4: Qualitative comparison of HiVT vs. PECT on real JRDB scenes.

Chapter 5

Discussion and Conclusions

This thesis has argued that reliable egocentric social trajectory forecasting requires a layered foundation: comprehensive data that includes the scenarios where prediction matters most, evaluation protocols built on that data that measure what deployment actually requires, and methods at the top that exploit the full richness of available sensor information. Each layer depends on the ones below it, since methods are only as good as the data they train on and the metrics they optimize toward, though feedback flows downward as well, with methods revealing which data to collect and which metrics to develop. The preceding chapters built this pyramid from the ground up. This final chapter synthesizes what these contributions collectively reveal about the state of the field, distills insights that cut across individual layers, and identifies the open problems that will shape the next generation of forecasting systems. The discussion that follows is organized thematically rather than layer-by-layer, because the most important lessons emerge at the intersections of data, evaluation, and methods.

5.1 Summary of Contributions

Chapter 2 introduced JaywalkerVR, a VR-based human-in-the-loop system for collecting safety-critical pedestrian-vehicle interaction data. The key insight—real human behavior in a simulated environment—bridges the behavioral sim-to-real gap that pure simulation cannot address. The CARLA-VR dataset demonstrated that VR-collected interactions improve forecasting on interactive scenarios (10.7% displacement error, 4.9% collision rate), and validation studies confirmed both the realism of collected data and industry recognition of the long-tail data problem.

Chapter 3 demonstrated that standard marginal metrics provide overly optimistic performance estimates by allowing “mix-and-match” across prediction samples. Joint metrics (*JADE*, *JFDE*) and collision rate revealed a $2\times$ gap between marginal and joint performance. Adding joint loss terms with zero architectural modifications yielded 7% *JADE*, 10% *JFDE*, and 16% collision rate improvements, confirming that the metrics we report directly shape the models we build.

Chapter 4 introduced PECT, a three-stream architecture integrating body pose and dense BEV environmental semantics with trajectory history. PECT introduced the ECR metric and a gated curriculum fusion strategy for three-modality integration, demonstrating improvements in both agent-agent and environment collision rates without degrading displacement accuracy.

5.2 The Data Problem Is Not Solved

Public trajectory forecasting benchmarks have been instrumental in driving progress, but they also shape which problems the field considers solved. The data contribution of this thesis exposes a structural limitation: existing benchmarks are dominated by routine scenarios, and methods that excel on these benchmarks may fail precisely where prediction matters most.

Routine-dominated benchmarks. The top-performing methods on the Waymo Open Motion Dataset leaderboard use only vectorized trajectory history and HD map information [46]. That this input representation suffices to achieve state-of-the-art performance is revealing: it suggests that current benchmarks are largely tractable from road structure alone. When a vehicle approaches a four-way intersection with a stop sign, its future motion is heavily constrained by lane geometry, traffic rules, and the positions of other vehicles. In such scenarios, trajectory-plus-map is arguably sufficient. But pedestrians jaywalking, children running into roads, or wheelchair users navigating construction zones, the scenarios where prediction failures have the most severe consequences, are precisely the scenarios underrepresented in these benchmarks. The Waymo leaderboard measures what it measures; it does not measure deployment readiness.

Tail performance matters more than mean performance. *ADE* averaged over a test set is dominated by the easy majority. A method that improves average *ADE* by 5% but degrades performance on the hardest 10% of scenarios is *worse* for deployment, not better. Our CARLA-VR experiments illustrate this tension directly: AgentFormer-VR improved substantially on nuScenes-interaction (the interactive, safety-critical subset) while showing slightly degraded performance on nuScenes-prediction (the routine-dominated full set). A metric that averages across both subsets would understate the improvement on the scenarios that matter and overstate the degradation on the scenarios where prediction is already adequate. The field needs stratified reporting by scenario difficulty as a standard practice, not an optional supplement.

VR as a validated data collection paradigm. JaywalkerVR is not merely a single system but a validated *methodology* for collecting behavioral data in safety-critical scenarios. The key distinction from pure simulation is that VR addresses the *behavioral* sim-to-real gap rather than the visual one. Neural rendering techniques (NeRF, 3D Gaussian Splatting) are rapidly closing the visual sim-to-real gap by producing photorealistic synthetic environments. VR complements these advances by ensuring that the *behavior* within those environments is generated by real humans rather than scripted policies or learned behavior models. An environment rendered with 3DGS but populated with agents following the Intelligent Driver Model still has a behavioral gap; a VR system with moderate visual fidelity but real human decision-making does not. These are complementary approaches, and combining them, combining photorealistic neural rendering with VR-based behavioral collection, is a natural next step. That said, VR behavioral fidelity is not perfect: presence scores of 5.3–5.6 on a 7-point scale indicate moderate-high rather than complete immersion, and known artifacts include defensive postures and occasional extra risk-taking from participants aware that virtual vehicles pose no real danger. These systematic biases may limit transferability, though whether they meaningfully affect downstream model training remains an open empirical question—and the improvements on nuScenes-interaction suggest that models are at least partially robust to them.

Toward better benchmarks. The path forward requires benchmarks that are explicitly designed around the scenarios where prediction is difficult and consequential. This means: (1) stratified evaluation splits that separate routine from interactive scenarios, (2) long-tail scenario categories as first-class evaluation dimensions rather than afterthoughts, (3) safety-critical interactions such as jaywalking, near-misses, and yielding behavior represented at sufficient scale to produce statistically meaningful results, and (4) reporting conventions that require per-category breakdowns rather than single aggregate numbers.

5.3 Evaluation for Deployment

The evaluation contribution of this thesis reveals that the trajectory forecasting field has a measurement problem, and that this measurement problem is not merely academic—it actively shapes the methods the community builds, and ultimately determines whether those methods are useful to the downstream planners they serve.

Marginal metrics overstate progress. The $2\times$ gap between marginal and joint performance is not just a quantitative discrepancy; it represents a fundamental flaw in how the field measures progress. When a method reports minADE_{20}, it selects the best of 20 samples independently for each agent. In a scene with 10 pedestrians, the reported "prediction" may combine agent 1's trajectory from sample 3, agent 2's trajectory from sample 7, and agent 3's trajectory from sample 15—a joint future that the model never actually predicted. A substantial portion of the "progress" reported on marginal metrics reflects mix-and-matching rather than genuine improvement in prediction quality. The problem is compounded by the generosity of $K = 20$ samples: a downstream planner cannot simultaneously act on 20 different futures, yet the field overwhelmingly reports minADE without examining whether the selected mode is also the highest-probability mode. Mode probability calibration matters as much as mode accuracy. In PECT, we deliberately use $K = 6$ modes, the more operationally realistic setting also used in the vehicle forecasting literature, and this choice itself reveals information about model quality that $K = 20$ obscures.

Evaluation-optimization coupling. The measurement problem is also an *optimization* problem. Evaluation metrics become implicit design specifications: researchers optimize architectures and loss functions to improve the numbers they report, and if those numbers reward mix-and-matching, that is what models learn to exploit. Our experiments demonstrated this directly: swapping in joint loss terms with *zero architectural changes* to AgentFormer yielded a 16% reduction in collision rate. The model's architecture was already capable of producing socially consistent predictions; the marginal training objective simply never asked it to. In this case, the bottleneck appeared to be training signal quality rather than architectural capacity. This tight coupling between metrics and model behavior is a general principle of machine learning: when the pedestrian forecasting field adopted minADE_{20} as its primary metric, it implicitly told researchers to build models that produce diverse marginal samples rather than coherent joint futures. The shift to joint metrics is not merely about more accurate measurement; it is about redirecting optimization pressure toward the properties that matter for deployment.

The reproducibility crisis in evaluation. Even when the community agrees on which metrics to report, inconsistent evaluation practices undermine comparability. In conducting the joint metrics evaluation of Chapter 3, we retrained or re-evaluated every baseline method to ensure fair comparison—and the process revealed how fragile published numbers are. Trajectron++ [177] had a data snooping bug that inflated its originally reported results. View Vertically [228] used a different sampling rate on the `eth` scene and a non-standard split of the `hotel` scene. On SDD, View Vertically preprocessed the raw dataset independently rather than using the standard TrajNet split [100], while AgentFormer and Trajectron++ had never been trained on SDD at all. Beyond splits, preprocessing choices that are rarely documented—whether to retain sequences with only a single agent, how to handle tracks with missing timesteps, what smoothing or interpolation to apply—can shift results by margins comparable to the differences between methods. The result is that published leaderboard comparisons often reflect evaluation setup differences as much as genuine methodological improvements. Standardized evaluation toolkits with locked preprocessing, fixed splits, and mandatory joint metric reporting would substantially improve the field's ability to measure real progress.

What planners actually need. Trajectory prediction exists in service of downstream planning, yet the prediction and planning communities remain largely separate. An accurate-but-uncalibrated prediction, one that produces diverse, low-error samples without reliable mode probabilities, forces a planner into worst-case reasoning across all modes, producing overly conservative behavior. A slightly-less-accurate-but-well-calibrated prediction, where the highest-probability mode is reliably the most likely future, may yield substantially better planning outcomes. A planner consuming $K = 20$ modes per agent faces a combinatorial explosion: with 10 agents, there are $20^{10} \approx 10^{13}$ possible joint futures, and no real-time system can reason over this space. In practice, planners use the top-1 or top-3 modes, making smaller K with better calibration more operationally useful than larger K with uncalibrated probabilities.

What metrics, then, should complement displacement error? Collision rate and environment collision rate capture failure modes that *ADE* is blind to and may even reward: the shortest path between two points often passes through other agents or walls. Beyond collision metrics, the planning community cares about time-to-collision, comfort (bounded acceleration and jerk), and route efficiency—dimensions that the prediction community has largely ignored. The vehicle planning community has already converged on multi-dimensional evaluation: nuPlan [20, 88] evaluates planners on composite scores aggregating collision avoidance, drivable area compliance, time-to-collision, comfort, and progress, while NAVSIM [40] introduced the Predictive Driver Model Score (PDMS), a weighted composite that multiplicatively penalizes safety violations and correlates far more strongly with closed-loop planning performance than displacement-based open-loop metrics, and Bench2Drive [84] provides a closed-loop benchmark evaluating multi-ability driving competence across diverse scenarios. The collision rate and ECR metrics introduced in this thesis represent initial steps toward bringing this philosophy to pedestrian prediction, but a full planning-aware evaluation framework, one that scores predictions by their downstream utility rather than their geometric accuracy, remains an open challenge. It is also worth acknowledging that even the joint metrics introduced in this thesis are still fundamentally displacement-based: they ensure predictions are internally consistent across agents but do not capture social norm compliance, planning utility, or behavioral plausibility beyond collision avoidance. Furthermore, all experiments in this thesis use academic-scale datasets containing thousands to tens of thousands of trajectories, while industry datasets (Waymo Open Motion, Argoverse 2) contain millions. Whether the magnitude of the marginal-joint gap, the benefits of VR data augmentation, and the gains from multi-modal fusion hold at industry scale is an open question. We expect the qualitative findings to transfer, but validating them at scale will likely require collaboration with industry partners who have access to the data volumes that academic groups cannot independently collect.

Toward prediction-planning co-design. The contributions of this thesis point toward a tighter integration of prediction and planning. Joint metrics reward predictions that are internally consistent—exactly the property a planner needs. Joint optimization improved collision rate as a side effect of better-aligned training objectives, without any explicit collision loss term, suggesting that properly formulated objectives implicitly encode the social and physical constraints that planners care about. Multi-modal inputs provide information that a planner also needs: body pose's temporal lead time (head turns and body leans precede trajectory changes by 0.5–2 seconds) gives a planner extra reaction cycles, and environmental context (traversability, obstacles) directly constrains the planner's own action space. Building prediction models that explicitly produce planner-consumable outputs, including calibrated mode probabilities, environment-aware predictions, and temporally leading intent signals, is a natural extension of the multi-modal approach developed in this thesis.

Evaluation takeaways. The findings above suggest a two-tier agenda for improving trajectory forecasting evaluation.

Immediate reporting practices. Papers should report, at minimum:

1. Both marginal and joint displacement metrics (ADE/FDE and JADE/JFDE), so the marginal-joint gap is always visible.
2. Collision rate on both the best-JADE sample (CR_{JADE}) and averaged across samples (CR_{mean}), with the agent radius explicitly stated.
3. Multiple K values: K=20 for comparability and K≤6 for operational realism.
4. Stratified splits separating routine from interactive/safety-critical scenarios.
5. Per-category breakdowns rather than single aggregate numbers.

Next-step planning-aware metrics. Three lightweight extensions would bridge toward planning-aware evaluation without requiring a full planner integration:

1. Top-1 and top-3 joint collision rate, reflecting the predictions a planner would actually act on.
2. Expected collision under predicted mode probabilities ($\sum_k p_k \cdot CR_k$), penalizing models that assign high probability to collision-prone modes.

3. Mode probability calibration: how often the top-1 predicted mode is actually closest to the ground truth.

These require only predicted mode probabilities and ground-truth trajectories, making them straightforward to compute alongside standard metrics.

5.4 The Multi-Modal Imperative

PECT's three-stream architecture is motivated by a simple observation: humans use multiple information sources to predict other humans' motion, and prediction models should too. But the specific findings from Chapter 4 reveal why this is harder than it sounds and what it means in practice.

Why trajectory alone is insufficient for pedestrians. For vehicles, trajectory history plus HD map often suffices—a claim supported by the success of trajectory-plus-map methods on the Waymo leaderboard. Vehicles are constrained to lanes, obey (mostly) predictable traffic rules, and have limited degrees of freedom. Pedestrians are fundamentally different: they can move in any direction, are not constrained to marked paths, and make decisions based on social context, intent, and environmental affordances that trajectory history alone cannot capture. The distinction is not merely that pedestrian prediction is harder; it is that the *information requirements* are qualitatively different.

Pose as a leading indicator. Body pose reveals pedestrian intent 0.5–2 seconds before observable trajectory changes [160, 165]. A head turn toward the road precedes a step off the curb; a body lean precedes a direction change; raised arms signal an intention to stop or gesture. For a planner operating at 10 Hz, 1 second of early warning translates to 10 additional planning cycles—the difference between a smooth deceleration and an emergency brake. This temporal lead time is pose's most practically valuable property for autonomous vehicle safety, and it is information that trajectory-only models fundamentally cannot access because the intent has not yet manifested in observable motion.

Environment as affordance, not just constraint. Vehicle prediction treats the environment as a hard constraint: vehicles must stay on roads. Pedestrian prediction requires a more nuanced view. Pedestrians have *soft spatial preferences*: grass is traversable but generally unpreferred; a doorway is both a barrier and a potential destination; a bench is an obstacle but also an attractor. BEV features encode this richer information—not just binary navigability but spatial affordances that influence where pedestrians are likely to go. The PECT experiments further revealed that *intermediate* BEV features transfer better across domains than decoded semantic labels: a wall on a campus and a barrier on a highway produce different semantic labels but similar geometric signatures in the feature space. This suggests that environment encoding for pedestrian prediction should operate at the representation level rather than the label level.

Multi-modal features and the long tail. The connection between richer input modalities and the long-tail data problem from Section 2.7 deserves emphasis. Routine scenarios, such as a pedestrian walking steadily along a sidewalk, are well-predicted by trajectory history alone because the past motion is a reliable predictor of the future. Safety-critical scenarios are precisely the cases where trajectory history is *insufficient*: the pedestrian's past motion does not yet reflect the dangerous action they are about to take. This is where additional modalities become essential, and where they provide not just better accuracy but also better *explainability*: the ability to attribute a prediction to interpretable input signals rather than opaque trajectory extrapolation.

Consider three examples. First, a child standing at a curb who suddenly darts into the road: trajectory history shows a stationary agent and provides no warning, but body pose may reveal a shift in weight or a turn of the head toward the street—signals that a crossing is imminent. A multi-modal model that flags the pose change gives both a better prediction and an interpretable reason for it. Second, a pedestrian walking parallel to a road who abruptly jaywalks between parked cars: trajectory-only models extrapolate continued parallel motion, but environment features encoding the gap between parked vehicles, combined with a head turn checking for traffic, reveal an affordance the pedestrian is likely to exploit. The prediction can be attributed to a specific spatial feature (the gap) and a specific pose signal (the head check). Third, a group of pedestrians approaching a narrow doorway who must transition from parallel walking to single-file: trajectory history suggests continued parallel motion, but environment features encoding the doorway constraint, combined with pose signals showing the group beginning to reorganize, enable the model to predict the bottleneck behavior. In each case, the additional modalities provide information that trajectory alone cannot, and crucially, that information is *human-*

interpretable; an engineer debugging a failure or a safety auditor reviewing a near-miss can trace the prediction back to specific pose and scene features rather than an inscrutable learned representation.

Domain gap as the silent challenge. Not all modalities carry equal domain risk, since body keypoints generalize across settings because human anatomy is invariant, but environment representations are vulnerable. PECT's BEV encoder is pretrained on nuScenes but evaluated on JRDB, and the fact that the environment stream still improves collision metrics despite this mismatch establishes a useful lower bound. The broader design principle is that modular systems with graceful degradation, such as gated curriculum fusion that suppresses uninformative environment features, are more practical than monolithic systems requiring perfect in-domain pretraining. But graceful degradation is a workaround, not a solution; closing the domain gap entirely is where foundation models enter the picture.

5.5 Future Directions: Foundation Models and Emerging Paradigms

Vision foundation models trained on internet-scale data offer features that are domain-agnostic by construction, and their emergence is poised to address the domain gap challenge head-on while reshaping the trajectory forecasting landscape more broadly.

From task-specific to foundation model features. PECT's environment collision rate metric currently relies on JRDB's panoptic segmentation labels to construct semantically labeled point clouds. A vision foundation model such as DINOv2 [150] or SAM [94] could automate this entirely: labeling obstacles from camera images and projecting them onto point clouds using available calibration parameters, removing the dependency on dataset-provided annotations. More directly, PECT's BEV encoder could be replaced with a foundation-model-based encoder that provides domain-agnostic spatial features, eliminating the nuScenes-to-JRDB domain gap that currently requires careful gated fusion to manage—the very challenge identified above. Monocular depth foundation models such as Depth Anything V2 [235] could further improve BEV construction by providing metric-scale depth estimates from camera images alone, reducing reliance on LiDAR for geometric reasoning. Foundation models offer features that are more robust to domain shift by construction, having been trained on vastly more diverse data than any single driving dataset. The broader autonomous driving community is already moving in this direction: UniAD [73] demonstrates end-to-end perception-prediction-planning with unified representations, while vision-language models such as DriveVLM [204] and language-based decision-making approaches like LanguageMPC [184] suggest that foundation model features can capture semantic context far richer than task-specific encoders.

Distillation for deployment. Foundation models are large—DINOv2-giant has over 1 billion parameters—while autonomous vehicle perception pipelines have strict latency requirements (typically $<$100 ms end-to-end). Knowledge distillation [14, 71] offers a path forward: a compact student network trained to mimic foundation model features can inherit much of the quality while meeting deployment constraints. BEVDistill [34] has already demonstrated that cross-modal distillation of BEV features is effective for 3D object detection, Hydra-MDP [114] shows that multi-teacher distillation from both human and rule-based planners can scale end-to-end driving, and similar approaches could transfer foundation model representations into lightweight encoders suitable for real-time prediction. Distillation is also a natural domain adaptation mechanism: a student trained on in-domain data to reproduce foundation model representations learns domain-adapted features without requiring foundation model fine-tuning. This approach could simultaneously address PECT's computational overhead and domain gap challenges.

Implications for each thesis contribution. Each layer of the pyramid interacts with foundation models differently. *Data*: foundation models combined with neural rendering could scale VR-based data collection by generating photorealistic environments and automating annotation, reducing the manual effort currently required. *Evaluation*: joint metrics and collision rate remain relevant regardless of model architecture—they measure properties of predictions, not properties of models. If anything, as models become more capable, rigorous evaluation becomes *more* important to distinguish genuine progress from benchmark saturation. *Methods*: foundation features could replace PECT's keypoint and BEV encoders with representations that have better out-of-distribution generalization, but the gated fusion strategy remains necessary. The fundamental challenge of integrating heterogeneous information sources, namely trajectory dynamics, body pose signals, and environmental context, does not disappear when those sources are encoded by foundation models rather than task-specific networks.

5.6 Closing Remarks

The three contributions of this thesis form a pyramid. Data is the foundation: without coverage of rare, safety-critical scenarios, neither evaluation nor methods can address the cases where prediction matters most. Evaluation is the middle layer: built on that data, joint metrics and collision rate reveal which methods genuinely improve interaction modeling and which merely exploit the permissiveness of marginal evaluation. Methods sit at the top: multi-modal architectures that leverage pose and environmental context can only be trained on sufficiently rich data and are only meaningfully assessed through the richer evaluation protocols below them.

The dependence is primarily upward, with methods resting on evaluation, which rests on data, but feedback flows downward as well. PECT's need for environment-aware assessment motivated the ECR metric, a case where a method drove evaluation innovation. Method failures on safety-critical scenarios motivated the VR data collection effort, a case where methods revealed data gaps. These descending feedback loops are real and valuable, but they are not symmetric with the foundational dependencies: one can have good data without good methods, but one cannot have good methods without good data.

The field stands at an inflection point. Foundation models, neural rendering, and end-to-end learning are reshaping the technical landscape of autonomous perception and prediction. The specific architectures and training procedures in this thesis will evolve; the principles will not. Targeted collection of tail-distribution data, evaluation protocols that align with deployment requirements, and methods that exploit complementary information sources are necessary regardless of whether the underlying models are transformers, diffusion models, or whatever comes next.

Safe autonomous navigation in human environments requires more than accurate trajectory extrapolation. It requires understanding *why* people move as they do—the intents revealed by body language, the affordances offered by the environment, the social dynamics that govern multi-agent interaction—and understanding *what the consequences are* of getting it wrong. The contributions of this thesis provide tools for each of these requirements: data that captures the scenarios where consequences are highest, metrics that measure whether models understand the joint structure of multi-agent futures, and methods that access the rich information streams needed to predict human behavior in all its complexity.

Bibliography

[1] Sander Ackermans, Debargha Dey, Peter Ruijten, Raymond H Cuijpers, and Bastian Pfleging. The effects of explicit intention communication, conspicuous sensors, and pedestrian attitude in interactions with automated vehicles. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–14, New York, NY, USA, April 2020. Association for Computing Machinery. 13

[2] Aditya Agarwal, Ankur Manglik, Sayan Banerjee, and Sriparna Saha. Can reasons help improve pedestrian intent estimation? a cross-modal approach. 2025. 69

[3] Mohd Faizan Ahmed, Naveed Syed, and Amir Rasouli. Pedestrian intention prediction via vision-language foundation models. 2025. 69

[4] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human Trajectory Prediction in Crowded Spaces. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 961–971, Las Vegas, NV, USA, June 2016. IEEE. ISBN 978-1-4673-8851-1. doi: 10.1109/CVPR.2016.110. URL http://ieeexplore.ieee.org/document/7780479/. 56

[5] Ferran Alet, Erica Weng, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Neural Relational Inference with Fast Modular Meta-learning. In *Advances in Neural Information Processing Systems*, page 12, 2019. 10

[6] Elmira Amirloo, Amir Rasouli, Peter Lakner, Mohsen Rohani, and Jun Luo. LatentFormer: Multi-Agent Transformer-Based interaction modeling and trajectory prediction. March 2022. 10

[7] Silvia Arias, Axel Mossberg, Daniel Nilsson, and Jonathan Wahlqvist. A study on evacuation behavior in physical and virtual reality experiments. *Fire Technol.*, 58(2): 817–849, March 2022. 13

[8] Inhwan Bae, Jean Oh, and Hae-Gon Jeon. Eigentrajectory: Low-rank descriptors for multi-modal trajectory forecasting. In *ICCV*, 2023. 47

[9] Inhwan Bae, Junoh Park, and Hae-Gon Jeon. Can language beat numerical regression? language-based multimodal trajectory prediction. In *CVPR*, 2024. 47

[10] Mohammadhossein Bahari, Vahid Zehtab, Sadegh Khorasani, Sana Ayromlou, Saeed Saadatnejad, and Alexandre Alahi. SVG-Net: An SVG-based Trajectory Prediction Model. *arXiv:2110.03706 [cs]*, October 2021. URL http://arxiv.org/abs/2110.03706. arXiv: 2110.03706. 10

[11] Mohammadhossein Bahari, Saeed Saadatnejad, Amirhossein Askari Farsangi, Seyed-Mohsen Moosavi-Dezfooli, and Alexandre Alahi. Certified human trajectory prediction. In *CVPR*, 2025. 11 and 48

[12] Mehmet Ilker Berkman and Ecehan Akan. Presence and immersion in virtual reality. In Newton Lee, editor, *Encyclopedia of Computer Graphics and Games*, pages 1–10. Springer International Publishing, Cham, 2019. 13 and 25

[13] Florian Berton, Anne-Hélène Olivier, Julien Bruneau, Ludovic Hoyet, and Julien Pettre. Studying gaze behaviour during collision avoidance with a virtual walker: Influence of the virtual reality setup. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 717–725, March 2019. 13

[14] Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. June 2021. 103

[15] Rajaram Bhagavathula, Brian Williams, Justin Owens, and Ronald Gibbons. The reality of virtual reality: A comparison of pedestrian behavior in real and virtual environments. *Proc. Hum. Fact. Ergon. Soc. Annu. Meet.*, 62(1):2056–2060, September 2018. 13

[16] Abhijat Biswas, Allan Wang, Gustavo Silvera, Aaron Steinfeld, and Henny Admoni. SocNavBench: A grounded simulation testing framework for evaluating social navigation. February 2021. 12 and 33

[17] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qual. Res. Psychol.*, 3(2):77–101, January 2006. 15

[18] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. March 2019. 1, 3, 11, 16, 48, and 77

[19] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *CoRR*, abs/1903.11027, 2019. URL http://arxiv.org/abs/1903.11027. 11

[20] Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric Wolff, Alex Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. nuPlan: A closed-loop ML-based planning benchmark for autonomous vehicles. In *ICRA 2021 Workshop on Safe Robot Learning*, 2021. 98

[21] Fanta Camara, Patrick Dickinson, Natasha Merat, and Charles Fox. Examining pedestrian-autonomous vehicle interactions in virtual reality. 09 2019. 18

[22] Yulong Cao, Chaowei Xiao, Anima Anandkumar, Danfei Xu, and Marco Pavone. AdvDO: Realistic adversarial attacks for trajectory prediction. In *Computer Vision – ECCV 2022*, pages 36–52. Springer Nature Switzerland, 2022. 53

[23] Zhangjie Cao, Erdem Bıyık, Guy Rosman, and Dorsa Sadigh. Leveraging smooth attention prior for Multi-Agent trajectory prediction. March 2022. 10

[24] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1):172–186, 2021. 68

[25] Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. In *Proceedings of the Conference on Robot Learning*, volume 100 of *Proceedings of Machine Learning Research*, pages 86–99. PMLR, 30 Oct–01 Nov 2020. URL https://proceedings.mlr.press/v100/chai20a.html. 10

[26] Ming-Fang Chang, Deva Ramanan, James Hays, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, and Simon Lucey. Argoverse: 3D Tracking and Forecasting With Rich Maps. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8740–8749, Long Beach, CA, USA, June 2019. IEEE. ISBN 978-1-72813-293-8. doi: 10.1109/CVPR.2019.00895. URL https://ieeexplore.ieee.org/document/8953693/. 3, 11, 16, and 48

[27] Changan Chen, Sha Hu, Payam Nikdel, Greg Mori, and Manolis Savva. Relational Graph Learning for Crowd Navigation. *arXiv:1909.13165 [cs]*, August 2020. URL http://arxiv.org/abs/1909.13165. arXiv: 1909.13165. 10

[28] Dapeng Chen, Bo Jiang, Ruimao Qian, and Liang Li. Mapexpert: Online hd map construction with simple and efficient sparse map element expert. 2024. 71

[29] Wenxiang Chen, Qianni Jiang, Xiangling Zhuang, and Guojie Ma. Comparison of pedestrians' gap acceptance behavior towards automated and human-driven vehicles. In *Engineering Psychology and Cognitive Ergonomics. Cognition and Design*, Lecture notes in computer science, pages 253–261. Springer International Publishing, Cham, 2020. 13

[30] Yifan Chen, Xunjiang Gu, Guanyu Song, Igor Gilitschenski, Marco Pavone, and Boris Ivanovic. Semantic raster bev fusion for autonomous driving via online hd map diffusion. 2025. 72

[31] Yu Fan Chen, Miao Liu, Michael Everett, and Jonathan P How. Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning.

September 2016. 12

[32] Yun Chen, Frieda Rong, Shivam Duggal, Shenlong Wang, Xinchen Yan, Sivabalan Manivasagam, Shangjie Xue, Ersin Yumer, and Raquel Urtasun. GeoSim: Realistic video simulation via Geometry-Aware composition for Self-Driving. January 2021. 12

[33] Yuxuan Chen, Hao Wang, and Yunpeng Li. Where do you go? pedestrian trajectory prediction using scene features. *arXiv preprint arXiv:2501.13848*, 2025. 71

[34] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. BEVDistill: Cross-modal BEV knowledge distillation for multi-view 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8602–8612, 2022. 103

[35] Jie Cheng, Xiaodong Mei, and Ming Liu. Forecast-mae: Self-supervised pre-training for motion forecasting with masked autoencoders. In *ICCV*, 2023. 10

[36] Luigi Filippo Chiara, Pasquale Coscia, Sourav Das, Simone Calderara, Rita Cucchiara, and Lamberto Ballan. Goal-driven self-attentive recurrent networks for trajectory prediction. pages 2518–2527, April 2022. 10

[37] Mark Colley and Enrico Rukzio. A design space for external communication of autonomous vehicles. In *12th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '20, pages 212–222, New York, NY, USA, September 2020. Association for Computing Machinery. 13

[38] Longchao Da, David Isele, Hua Wei, and Manish Saroya. Measuring what matters: Scenario-driven evaluation for trajectory predictors in autonomous driving. 2025. 45

[39] Haleh Damirchi, Ali Etemad, and Michael Greenspan. Socially-informed reconstruction for pedestrian trajectory forecasting. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025. 47

[40] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, Andreas Geiger, and Kashyap Chitta. NAVSIM: Data-driven non-reactive autonomous vehicle simulation and benchmarking. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2024. 98

[41] Koen de Clercq, Andre Dietrich, Juan Pablo Núñez Velasco, Joost de Winter, and Riender Happee. External Human-Machine interfaces on automated vehicles: Effects on pedestrian crossing decisions. *Hum. Factors*, 61(8):1353–1370, December 2019. 13

[42] Shuchisnigdha Deb, Daniel W Carruth, and Christopher R Hudson. How communicating features can help pedestrian safety in the presence of self-driving vehicles:

Virtual reality experiment. *IEEE Trans. Hum. Mach. Syst.*, 50(2):176–186, April 2020. 13

[43] Wenjie Ding, Limeng Qiao, Xi Qiu, and Chi Zhang. Pivotnet: Vectorized pivot learning for end-to-end hd map construction. In *ICCV*, 2023. 71

[44] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. November 2017. 4, 12, and 14

[45] Tobias Drey, Michael Rietzler, and Enrico Rukzio. Questionnaires and qualitative feedback methods to measure user experience in mixed reality. April 2021. 13

[46] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles Qi, Yin Zhou, Zoey Yang, Aurelien Chouard, Pei Sun, Jiquan Ngiam, Vijay Vasudevan, Alexander McCauley, Jonathon Shlens, and Dragomir Anguelov. Large scale interactive motion forecasting for autonomous driving : The waymo open motion dataset. April 2021. 1, 3, 11, 38, 48, and 94

[47] Siqi Fan, Lijun Wang, Jisoo Shin, and In So Park. 3d human mesh recovery from LiDAR. In *CVPR*, 2024. 68

[48] Ilja T Feldstein and Georg N Dyszak. Road crossing decisions in real and virtual environments: A comparative study on simulator validity. *Accid. Anal. Prev.*, 137: 105356, March 2020. 13

[49] Yan Feng, Dorine Duives, Winnie Daamen, and Serge Hoogendoorn. Data collection methods for studying pedestrian behaviour: A systematic review. *Build. Environ.*, 187(107329):107329, January 2021. 13

[50] Yuxiang Fu, Qi Yan, Lele Wang, Ke Li, and Renjie Liao. MoFlow: One-step flow matching for human trajectory forecasting via implicit maximum likelihood estimation based distillation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 62

[51] Tanja Fuest, Anna Sophia Maier, Hanna Bellem, and Klaus Bengler. How should an automated vehicle communicate its intention to a pedestrian? – a virtual reality study. In *Human Systems Engineering and Design II*, pages 195–201. Springer International Publishing, 2020. 13

[52] Ryo Fujii, Hideo Saito, and Ryo Hachiuma. Realtraj: Towards real-world pedestrian trajectory forecasting. 2025. 11

[53] Ryo Fujii, Hideo Saito, and Ryo Hachiuma. Towards predicting any human trajectory in context. In *NeurIPS*, 2025. 11

[54] Salma Galaaoui, Eduardo Valle, David Picard, and Nermin Samet. 3D human pose and shape estimation from LiDAR point clouds: A review. *arXiv preprint arXiv:2509.12197*, 2025. 69

[55] A Geiger, P Lenz, C Stiller, and R Urtasun. Vision meets robotics: The KITTI dataset. *Int. J. Rob. Res.*, 32(11):1231–1237, September 2013. 11

[56] Darren George and Paul Mallery. *SPSS for Windows Step by Step: A Simple Guide and Reference, 11.0 Update*. Allyn & Bacon, Boston, 4th edition, 2003. 28

[57] Akshat Ghiya, Ali K. AlShami, and Jugal Kalita. SGNetPose+: Stepwise goal-driven networks with pose information for trajectory prediction in autonomous driving. In *IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 2025. 70

[58] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *CVPR*, 2024. 68

[59] Mar Gonzalez-Franco and Tabitha C Peck. Avatar embodiment. towards a standardized questionnaire. *Front Robot AI*, 5:74, June 2018. 26

[60] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014. 10

[61] Xunjiang Gu, Guanyu Song, Igor Gilitschenski, Marco Pavone, and Boris Ivanovic. Producing and leveraging online map uncertainty in trajectory prediction. In *CVPR*, 2024. 72

[62] Xunjiang Gu, Guanyu Song, Igor Gilitschenski, Marco Pavone, and Boris Ivanovic. Accelerating online mapping and behavior prediction via direct BEV feature attention, 2024. URL https://arxiv.org/abs/2407.06683. 72, 74, 77, 79, 84, 85, and 87

[63] Xunjiang Gu, Guanyu Song, Igor Gilitschenski, Marco Pavone, and Boris Ivanovic. Delving into mapping uncertainty for mapless trajectory prediction. 2025. 72

[64] Jiaqi Guan, Ye Yuan, Kris M Kitani, and Nicholas Rhinehart. Generative hybrid representations for activity forecasting with no-regret learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 173–182, 2020. 10

[65] Ke Guo, Dawei Wang, Tingxiang Fan, and Jia Pan. VR-ORCA: Variable responsibility optimal reciprocal collision avoidance. *IEEE Robotics and Automation Letters*, 6 (3):4520–4527, July 2021. 12

[66] Ziming Guo et al. Recent advances in multi-agent human trajectory prediction. *arXiv preprint arXiv:2506.14831*, 2025. 46

[67] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social GAN: Socially Acceptable Trajectories With Generative Adversarial Networks. pages 2255–2264, 2018. URL https:

//openaccess.thecvf.com/content_cvpr_2018/html/Gupta_Social_GAN_Socially_CVPR_2018_paper.html. 1, 10, 47, 56, 57, and 64

[68] Sandra G Hart and Lowell E Staveland. Development of NASA-TLX (task load index): Results of empirical and theoretical research. *Human mental workload.*, 382: 139–183, 1988. 26

[69] Tilo Hartmann, Werner Wirth, Holger Schramm, Christoph Klimmt, Peter Vorderer, André Gysbers, Saskia Böcking, Niklas Ravaja, Jari Laarni, Timo Saari, Feliz Gouveia, and Ana Maria Sacau. The spatial presence experience scale (SPES): A short self-report measure for diverse media settings. *Journal of Media Psychology: Theories, Methods, and Applications*, 28(1):1–15, 2016. 26

[70] Dirk Helbing and Peter Molnar. Social Force Model for Pedestrian Dynamics. *Physical Review E*, 51(5):4282–4286, May 1995. ISSN 1063-651X, 1095-3787. doi: 10.1103/PhysRevE.51.4282. URL http://arxiv.org/abs/cond-mat/9805244. arXiv: cond-mat/9805244. 47

[71] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. March 2015. 103

[72] Haotian Hu, Fanyi Liu, Kunhong Gao, Guoqiang Liao, Zuofan Jiang, Shaohua Xu, and Bing Liu. Admap: Anti-disturbance framework for reconstructing online vectorized hd map. 2024. 71

[73] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiagang Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17853–17862, 2023. 102

[74] Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. In *arXiv preprint arXiv:2112.11790*, 2021. 71

[75] Renhao Huang, Hao Xue, Maurice Pagnucco, Flora Salim, and Yang Song. Multimodal trajectory prediction: A survey, 2023. 10

[76] Yanjun Huang, Jiatong Du, Ziru Yang, Zewei Zhou, Lin Zhang, and Hong Chen. A survey on trajectory-prediction methods for autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 7(3):652–674, 2022. doi: 10.1109/TIV.2022.3167103. 10

[77] Zhiyu Huang, Xiaoyu Mo, and Chen Lv. Multi-modal motion prediction with transformer-based neural network for autonomous driving. September 2021. 10

[78] Zhiyu Huang, Haochen Liu, and Chen Lv. Gameformer: Game-theoretic modeling and learning of transformer-based interactive prediction and planning for autonomous

driving. In *ICCV*, 2023. 10 and 47

[79] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 2014. 68

[80] Boris Ivanovic and Marco Pavone. The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2375–2384, 2019. 1, 10, and 47

[81] Omar Janeh, Gerd Bruder, Frank Steinicke, Alessandro Gulberti, and Monika Poetter-Nerger. Analyses of gait parameters of younger and older adults during (Non-)Isometric virtual walking. *IEEE Trans. Vis. Comput. Graph.*, 24(10):2663–2674, October 2018. 13

[82] Suresh Kumaar Jayaraman, Chandler Creech, Lionel P Robert, Jr, Dawn M Tilbury, X Jessie Yang, Anuj K Pradhan, and Katherine M Tsui. Trust in AV. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, New York, NY, USA, March 2018. ACM. 13

[83] Jaewoo Jeong and Hae-Gon Jeon. Multi-modal knowledge distillation-based human trajectory forecasting. In *CVPR*, 2025. 70

[84] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. In *NeurIPS*, 2024. 98

[85] Chiyu Max Jiang, Andre Cornman, Cheolho Park, Benjamin Sapp, Yin Zhao, and Dragomir Anguelov. Motiondiffuser: Controllable multi-agent motion prediction using diffusion. In *CVPR*, 2023. 46

[86] Zhou Jiang, Zhenxin Zhu, Pengfei Li, Huan-ang Gao, Tianyuan Yuan, Yongliang Shi, Hang Zhao, and Hao Zhao. P-mapnet: Far-seeing map generator enhanced by both sdmap and hdmap priors. In *ECCV*, 2024. 71

[87] Arash Kalatian and Bilal Farooq. A context-aware pedestrian trajectory prediction framework for automated vehicles. *CoRR*, abs/2104.08123, 2021. URL `https://arxiv.org/abs/2104.08123`. 36

[88] Napat Karnchanachari, Dimitris Geromichalos, Kok Seang Tan, Nanxiang Li, Christopher Eriksen, Shakiba Yaghoubi, Noushin Mehdipour, Gianmarco Bernasconi, Whye Kit Fong, Yiluan Guo, and Holger Caesar. Towards learning-based planning: The nuPlan benchmark for real-world autonomous driving. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024. 98

[89] Rawal Khirodkar, Jyun-Ting Song, Jinkun Cao, Zhengyi Luo, and Kris Kitani. Harmony4d: A video dataset for in-the-wild close human interactions. In *NeurIPS Datasets and Benchmarks Track*, 2023. 22 and 41

[90] Jaewoo Kim, Hanjun Lee, and Hae-Gon Jeon. Prediction of pedestrian trajectories by trajectory-scene-cell classification. In *ICLR*, 2025. 71

[91] Wonhui Kim, Manikandasriram Srinivasan Ramanagopal, Charles Barto, Ming-Yuan Yu, Karl Rosaen, Nick Goumas, Ram Vasudevan, and Matthew Johnson-Roberson. PedX: Benchmark dataset for metric 3D pose estimation of pedestrians in complex urban intersections. September 2018. 11

[92] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 10

[93] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International Conference on Machine Learning*, pages 2688–2697. PMLR, 2018. 10

[94] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. April 2023. 102

[95] Fabian Konstantinidis, Ariel Dallari Guerreiro, Raphael Trumpp, Moritz Sackmann, Ulrich Hofmann, Marco Caccamo, and Christoph Stiller. From marginal to joint predictions: Evaluating scene-consistent trajectory prediction approaches for automated driving. In *ITSC*, 2025. 48 and 62

[96] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian D Reid, Hamid Rezatofighi, and Silvio Savarese. Social-bigat: multimodal trajectory forecasting using bicycle-gan and graph attention networks. In *Advances in Neural Information Processing Systems 2019*. Neural Information Processing Systems (NIPS), 2019. 10 and 47

[97] Parth Kothari and Alexandre Alahi. Safety-compliant generative adversarial networks for human trajectory forecasting. September 2022. 10 and 48

[98] Parth Kothari, Sven Kreiss, and Alexandre Alahi. Interpretable social anchors for human trajectory forecasting in crowds. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15556–15566, 2021. 48 and 71

[99] Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human Trajectory Forecasting in Crowds: A Deep Learning Perspective. *arXiv:2007.03639 [cs]*, January 2021. URL `http://arxiv.org/abs/2007.03639`. arXiv: 2007.03639. 12, 51, 52, and 57

[100] Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human trajectory forecasting in crowds: A deep learning perspective. *IEEE Trans. Intell. Transp. Syst.*, 23(7): 7386–7400, July 2022. 12, 33, 48, 57, 59, and 97

[101] Iuliia Kotseruba, Amir Rasouli, and John K. Tsotsos. Joint attention in autonomous driving (JAAD). *CoRR*, abs/1609.04741, 2016. URL `http://arxiv.org/abs/1609.04741`. 69

[102] Iuliia Kotseruba, Amir Rasouli, and John K Tsotsos. Benchmark for evaluating pedestrian action prediction. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1258–1268, 2021. 69

[103] Roland Kress, Stefan Zernetsch, Konrad Doll, and Bernhard Sick. Pip-net: Pedestrian intention prediction in the wild. *arXiv preprint arXiv:2402.12810*, 2024. 69

[104] Viktor Kress, Fabian Jeske, Stefan Zernetsch, Konrad Doll, and Bernhard Sick. Pose and semantic map based probabilistic forecast of vulnerable road users' trajectories. June 2021. 70

[105] Viktor Kress et al. Enhancing trajectory prediction with human body pose. *arXiv preprint arXiv:2507.22742*, 2025. 70

[106] Bo Lang and Mooi Choo Chuah. Event-guided video transformer for end-to-end 3d human pose estimation. In *WACV*, 2025. 69

[107] Mihee Lee, Samuel S. Sohn, Seonghyeon Moon, Sejong Yoon, Mubbasir Kapadia, and Vladimir Pavlovic. MUSE-VAE: Multi-scale VAE for environment-aware long range trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2221–2230, 2022. 83

[108] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 336–345, 2017. 10 and 47

[109] Yee Mun Lee, Ruth Madigan, Jorge Garcia, Andrew Tomlinson, Albert Solernou, Richard Romano, Gustav Markkula, Natasha Merat, and Jim Uttley. Understanding the messages conveyed by automated vehicles. In *Proceedings of the 11th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '19, pages 134–143, New York, NY, USA, September 2019. Association for Computing Machinery. 13

[110] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. *Comput. Graph. Forum*, 26(3):655–664, September 2007. 1, 11, and 56

[111] Haoran Li et al. Intention-aware diffusion model for pedestrian trajectory prediction. *arXiv preprint arXiv:2508.07146*, 2025. 47

[112] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13147–13156, 2022. 68

[113] Xiaoshuai Li, Yifan Gu, Kun Jiang, and Diange Yang. Mapdistill: Boosting efficient camera-based hd map construction via camera-lidar fusion model distillation. 2024. 71

[114] Zhenxin Li, Kailin Li, Shihao Wang, Shiyi Lan, Zhiding Yu, Yishen Ji, Zhiqi Li, Ziyue Zhu, Jan Kautz, Zuxuan Wu, Yu-Gang Jiang, and Jose M. Alvarez. Hydra-MDP: End-to-end multimodal planning with multi-target hydra-distillation. In *NeurIPS*, 2024. 103

[115] Junwei Liang, Lu Jiang, Kevin Murphy, Ting Yu, and Alexander Hauptmann. The garden of forking paths: Towards Multi-Future trajectory prediction. December 2019. 12, 13, and 49

[116] Junwei Liang, Lu Jiang, and Alexander G. Hauptmann. Simaug: Learning robust representations from 3d simulation for pedestrian trajectory prediction in unseen cameras. *CoRR*, abs/2004.02022, 2020. URL `https://arxiv.org/abs/2004.02022`. 12

[117] Ming Liang, Bin Yang, Rui Hu, Yun Chen, Renjie Liao, Song Feng, and Raquel Urtasun. Learning lane graph representations for motion forecasting. July 2020. 10

[118] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. BEVFusion: A simple and robust LiDAR-Camera fusion framework. May 2022. 77

[119] Bencheng Liao, Shaoyu Chen, Xinggang Wang, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Chang Huang. MapTR: Structured modeling and learning for online vectorized HD map construction. In *International Conference on Learning Representations (ICLR)*, 2023. 71

[120] Bencheng Liao, Shaoyu Chen, Yunchi Zhang, Bo Jiang, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. MapTRv2: An end-to-end framework for online vectorized HD map construction. *International Journal of Computer Vision*, 2024. 71

[121] Sehwan Liao, Xunjiang Gu, Guanyu Song, Igor Gilitschenski, Marco Pavone, and Boris Ivanovic. Mask2map: Vectorized hd map construction using bird's eye view segmentation masks. In *ECCV*, 2024. 71

[122] Haotian Lin, Yixiao Wang, Mingxiao Huo, Chensheng Peng, Zhiyuan Liu, and Masayoshi Tomizuka. Joint pedestrian trajectory prediction through posterior sampling. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024. 61

[123] Hongzhi Liu, Ruimin Zhang, Jifeng Wang, and Deheng Qian. Learning cooperative trajectory representations for motion forecasting. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 72

[124] Jianwei Liu, Hao Ding, Changyin Li, and Yuxuan Yang. DifTraj: Diffusion inspired by intrinsic intention and extrinsic interaction for multi-modal trajectory prediction. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2024. 62

[125] Minghuan Liu, Tairan He, Minkai Xu, and Weinan Zhang. Energy-Based Imitation Learning. *arXiv:2004.09395 [cs, stat]*, April 2021. URL http://arxiv.org/abs/2004.09395. arXiv: 2004.09395. 10

[126] Xiaolu Liu, Song Liu, Zhaokai Lin, Jing Ye, Hai-Rui Tan, and Hengtao Ding. Mgmap: Mask-guided learning for online vectorized hd map construction. In *CVPR*, 2024. 71

[127] Yuejiang Liu, Riccardo Cadei, Jonas Schweizer, Sherwin Bahmani, and Alexandre Alahi. Towards Robust and Adaptive Motion Forecasting: A Causal Representation Perspective. *arXiv:2111.14820 [cs]*, November 2021. URL http://arxiv.org/abs/2111.14820. arXiv: 2111.14820. 10

[128] Yuejiang Liu, Qi Yan, and Alexandre Alahi. Social NCE: Contrastive Learning of Socially-aware Motion Representations. *arXiv:2012.11717 [cs]*, August 2021. URL http://arxiv.org/abs/2012.11717. arXiv: 2012.11717. 10 and 48

[129] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2774–2781. IEEE, 2023. 75 and 77

[130] Zhixin Liu, Ziyang Liu, Yujiang Zhu, Yifan Lu, Hengshuang Luo, and Hang Zhao. Online vectorized hd map construction using geometry. In *CVPR*, 2024. 71

[131] Javier Lorenzo, Iñigo Alonso, Rubén Izquierdo, Augusto Luis Ballardini, Alvaro H. Saz, David F. Llorca, and Miguel A. Sotelo. CAPformer: Pedestrian crossing action prediction using transformer. In *Sensors*, volume 21, 2021. 69

[132] Wenjie Luo, Cheolho Park, Andre Cornman, Benjamin Sapp, and Dragomir Anguelov. JFP: Joint future prediction with interactive Multi-Agent modeling for autonomous driving. December 2022. 46

[133] Karthik Mahadevan, Elaheh Sanoubari, Sowmya Somanath, James E Young, and Ehud Sharlin. AV-Pedestrian interaction design using a pedestrian mixed traffic simulator. In *Proceedings of the 2019 on Designing Interactive Systems Conference*, DIS '19, pages 475–486, New York, NY, USA, June 2019. Association for Computing Machinery. 13

[134] Karttikeya Mangalam, Yang An, Harshayu Girase, and Jitendra Malik. From Goals, Waypoints & Paths To Long Term Human Trajectory Forecasting. *arXiv:2012.01526 [cs]*, December 2020. URL http://arxiv.org/abs/2012.01526. arXiv: 2012.01526. 1, 10, 47, 56, 57, and 64

[135] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *European Conference on Computer Vision*, pages 759–776. Springer, 2020. 10, 56, 57, 64, and 71

[136] Sivabalan Manivasagam, Shenlong Wang, Kelvin Wong, Wenyuan Zeng, Mikita Sazanovich, Shuhan Tan, Bin Yang, Wei-Chiu Ma, and Raquel Urtasun. LiDARsim: Realistic LiDAR simulation by leveraging the real world. 2020. 4

[137] Weibo Mao, Chenxin Xu, Qi Zhu, Siheng Chen, and Yanfeng Wang. Leapfrog diffusion model for stochastic trajectory prediction. In *CVPR*, 2023. 72

[138] Roberto Martin-Martin, Mihir Patel, Hamid Rezatofighi, Silvio Savarese, et al. Jrdb: A dataset and benchmark for visual perception for navigation in human environments. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2021. 77 and 82

[139] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2640–2649, 2017. 68

[140] Soroush Mehraban, He Meng, Yiqing Peng, Luc Van Gool, and Xi Chen. Motionagformer: Enhancing 3d human pose estimation with a transformer-gcn approach. In *arXiv preprint arXiv:2405.11039*, 2024. 68

[141] Leonel Merino, Magdalena Schwarzl, Matthias Kraus, Michael Sedlmair, Dieter Schmalstieg, and Daniel Weiskopf. Evaluating mixed and augmented reality: A systematic literature review (2009-2019). October 2020. 13

[142] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction. *arXiv:2002.11927 [cs]*, March 2020. URL `http://arxiv.org/abs/2002.11927`. arXiv: 2002.11927 version: 3. 10 and 47

[143] Kenta Mukoya, Erica Weng, Rohan Choudhury, and Kris Kitani. JaywalkerVR: A VR system for collecting safety-critical pedestrian-vehicle interactions. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024. 17

[144] Niels Christian Nilsson, Stefania Serafin, and Rolf Nordahl. The effect of head mounted display weight and locomotion method on the perceived naturalness of virtual walking speeds. In *2015 IEEE Virtual Reality (VR)*, pages 249–250, March 2015. 13

[145] Niels Christian Nilsson, Tabitha Peck, Gerd Bruder, Eri Hodgson, Stefania Serafin, Mary Whitton, Frank Steinicke, and Evan Suma Rosenberg. 15 years of research on redirected walking in immersive virtual environments. *IEEE Comput. Graph. Appl.*, 38(2):44–56, March 2018. 13

[146] Nahal Norouzi, Kangsoo Kim, Myungho Lee, Ryan Schubert, Austin Erickson, Jeremy Bailenson, Gerd Bruder, and Greg Welch. Walking your virtual dog: Analysis of awareness and proxemics with simulated support animals in augmented reality. In *2019 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 157–168, October 2019. 13

[147] J Pablo Nuñez Velasco, Haneen Farah, Bart van Arem, and Marjan P Hagenzieker. Studying pedestrians' crossing behavior when interacting with automated vehicles using virtual reality. *Transp. Res. Part F Traffic Psychol. Behav.*, 66:1–14, October 2019. 13

[148] Nami Ogawa, Takuji Narumi, Hideaki Kuzuoka, and Michitaka Hirose. Do you feel like passing through walls?: Effect of Self-Avatar appearance on facilitating realistic behavior in virtual environments. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–14, New York, NY, USA, April 2020. Association for Computing Machinery. 13

[149] Anne-Helene Olivier, Julien Bruneau, Richard Kulpa, and Julien Pettre. Walking with virtual people: Evaluation of locomotion interfaces in dynamic environments. *IEEE Trans. Vis. Comput. Graph.*, 24(7):2251–2263, July 2018. 13

[150] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. April 2023. 102

[151] Tina Øvad. Likert scale & semantic differential scale. https://preely.com/likert-scale-semantic-differential-scale/, 2023. Accessed: 2023-8-29. 25

[152] Priyanka Patel and Michael J. Black. CameraHMR: Aligning people with perspective. In *3DV*, 2025. 68

[153] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7753–7762, 2019. 68

[154] S Pellegrini, A Ess, K Schindler, and L van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*. IEEE, September 2009. 1, 11, and 56

[155] Jihua Peng, Yanghong Zhou, and P. Y. Mok. Ktpformer: Kinematics and trajectory prior knowledge-enhanced transformer for 3d human pose estimation. In *CVPR*,

2024. 68

[156] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision (ECCV)*, pages 194–210, 2020. 71

[157] Jonah Philion, Xue Bin Sun, Wenjie Luo, Benjamin Sapp, and Dragomir Anguelov. Scene-level multi-agent trajectory generation with consistent diffusion models. 2024. 46

[158] Iana Podkosova and Hannes Kaufmann. Co-presence and proxemics in shared walkable virtual environments with mixed colocation. In *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology*, number Article 21 in VRST '18, pages 1–11, New York, NY, USA, November 2018. Association for Computing Machinery. 13

[159] Qingze, Liu, Danrui Li, Samuel S. Sohn, Sejong Yoon, Mubbasir Kapadia, and Vladimir Pavlovic. Trajdiffuse: A conditional diffusion model for environment-aware trajectory prediction, 2024. URL `https://arxiv.org/abs/2410.10804`. 72

[160] Ricardo Quintero, Jorge Almeida, Boris Vintimilla, and Miguel Torres-Torriti. Pedestrian path prediction based on body language and action classification. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–6, 2014. 69 and 100

[161] Navin Ranjan, Bruno Artacho, and Andreas Savakis. Waterfall transformer for multi-person pose estimation. In *WACV*, 2025. 69

[162] Munir Rashid, Sungbin Bae, Wenjie Luo, Benjamin Sapp, and Dragomir Anguelov. Pedestrian crossing action recognition and trajectory prediction with 3d human keypoints. In *ICRA*, 2024. 69

[163] Amir Rasouli, Iuliia Kotseruba, and John K. Tsotsos. Are they going to cross? A benchmark dataset and baseline for pedestrian crosswalk behavior. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2017. 69

[164] Amir Rasouli, Iuliia Kotseruba, Toni Kunic, and John K Tsotsos. Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6262–6271, 2019. 11 and 69

[165] Amir Rasouli, Iuliia Kotseruba, and John K Tsotsos. Pedestrian action anticipation using contextual feature fusion in stacked rnns. *British Machine Vision Conference (BMVC)*, 2020. 69 and 100

[166] Amir Rasouli, Mohsen Rohani, and Jun Luo. Pedformer: Pedestrian behavior prediction via cross-modal attention modulation and gated multitask learning. In

ICRA, 2023. 69

[167] Yiming Ren, Xiao Zhao, Zhengxin Liu, Qian Li, Siqi Yuan, Bowen Jiang, Chenglu Wen, and Cheng Wang. LiveHPS: LiDAR-based scene-level human pose and shape estimation in free environment. In *CVPR*, 2024. 69

[168] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pages 1530–1538. PMLR, 2015. 10

[169] Nicholas Rhinehart, Kris M Kitani, and Paul Vernaza. R2p2: A reparameterized pushforward policy for diverse, precise generative path forecasting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 772–788, 2018. 10 and 47

[170] Nicholas Rhinehart, Rowan McAllister, Kris Kitani, and Sergey Levine. Precog: Prediction conditioned on goals in visual multi-agent settings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2821–2830, 2019. 10 and 47

[171] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *Computer Vision – ECCV 2016*, pages 549–565. Springer International Publishing, 2016. 1, 11, and 57

[172] Lloyd Russell, Anthony Hu, Lorenzo Bertoni, George Fedoseev, Jamie Shotton, Elahe Arani, and Gianluca Corrado. Gaia-2: A controllable multi-view generative world model for autonomous driving. 2025. 12

[173] Saeed Saadatnejad, Mohammadhossein Bahari, Pedram Khorsandi, Mohammad Saneian, Seyed-Mohsen Moosavi-Dezfooli, and Alexandre Alahi. Are socially-aware trajectory prediction models really socially-aware? *arXiv:2108.10879 [cs]*, August 2021. URL `http://arxiv.org/abs/2108.10879`. arXiv: 2108.10879. 48 and 53

[174] Saeed Saadatnejad, Yang Gao, Kaouther Messaoud, and Alexandre Alahi. Social-transmotion: Promptable human trajectory prediction. In *ICLR*, 2024. 70

[175] Saeed Saadatnejad, Taylor Schaub, Simin Ghasemi, and Alexandre Alahi. Social-pose: Trajectory prediction with body pose. *arXiv preprint arXiv:2507.22742*, 2025. 70, 79, and 87

[176] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1349–1358, 2019. 1, 10, 47, and 57

[177] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-Feasible Trajectory Forecasting With Heterogeneous Data. *arXiv:2001.03093 [cs]*, January 2021. URL `http://arxiv.org/abs/2001.03093`. arXiv: 2001.03093. 1, 10, 47, 56, 57, 64, and 97

[178] Tim Salzmann, Lewis Chiang, Markus Ryll, Dorsa Sadigh, Carolina Parada, and Alex Bewley. Robots that can see: Leveraging human pose for trajectory prediction. 2023. xii, 70, 79, 84, 85, 86, and 87

[179] Gayani Samaraweera, Rongkai Guo, and John Quarles. Latency and avatars in virtual environments and the effects on gait for persons with mobility impairments. In *2013 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 23–30, March 2013. 13

[180] Neal Schmitt. Uses and abuses of coefficient alpha. *Psychological Assessment*, 8(4): 350–353, 1996. doi: 10.1037/1040-3590.8.4.350. 28

[181] Paul Schmitt, Nicholas Britten, Jihyun Jeong, Amelia Coffey, Kevin Clark, Shweta Sunil Kothawade, Elena Corina Grigore, Adam Khaw, Christopher Konopka, Linh Pham, Kim Ryan, Christopher Schmitt, Aryaman Pandya, and Emilio Frazzoli. nureality: A VR environment for research of pedestrian and autonomous vehicle interactions. *CoRR*, abs/2201.04742, 2022. URL `https://arxiv.org/abs/2201.04742`. 14

[182] Sonja Schneider and Klaus Bengler. Virtually the same? analysing pedestrian behaviour by means of virtual reality. *Transp. Res. Part F Traffic Psychol. Behav.*, 68:231–256, January 2020. 13

[183] Ari Seff, Brian Cera, Dian Chen, Mason Ng, Aurick Zhou, Nigamaa Nishi, Pei Peng, and Dragomir Anguelov. Motionlm: Multi-agent motion forecasting as language modeling. In *ICLR*, 2024. 10, 46, and 47

[184] Hao Sha, Yao Mu, Yuxuan Jiang, Li Chen, Chenfeng Xu, Ping Luo, Shengbo Eben Li, Masayoshi Tomizuka, Wei Zhan, and Mingyu Ding. Languagempc: Large language models as decision makers for autonomous driving. *arXiv preprint arXiv:2310.03026*, 2023. 103

[185] Azmeh Shahid, Kate Wilkinson, Shai Marcu, and Colin M Shapiro. Stanford sleepiness scale (SSS). In Azmeh Shahid, Kate Wilkinson, Shai Marcu, and Colin M Shapiro, editors, *STOP, THAT and One Hundred Other Sleep Scales*, pages 369–370. Springer New York, New York, NY, 2012. 26 and 27

[186] Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion transformer with global intention localization and local movement refinement. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 6 and 79

[187] Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. MTR++: Multi-agent motion prediction with symmetric scene modeling and guided intention querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 6

[188] Gustavo Silvera, Abhijat Biswas, and Henny Admoni. DReyeVR: Democratizing virtual reality driving simulation for behavioural & interaction research. January 2022. 14

[189] SimScale Authors. Simscale: Learning to drive via real-world simulation at scale. 2025. 12

[190] Mel Slater and Sylvia Wilbur. A framework for immersive virtual environments five: Speculations on the role of presence in virtual environments. *Presence: Teleoper. Virtual Environ.*, 6(6):603–616, December 1997. 13

[191] Samuel S. Sohn, Mihee Lee, Seonghyeon Moon, Gang Qiao, Muhammad Usman, Sejong Yoon, Vladimir Pavlovic, and Mubbasir Kapadia. A2X: An Agent and Environment Interaction Benchmark for Multimodal Human Trajectory Prediction. In *Motion, Interaction and Games*, pages 1–9, Virtual Event Switzerland, November 2021. ACM. ISBN 978-1-4503-9131-3. doi: 10.1145/3487983.3488302. URL `https://dl.acm.org/doi/10.1145/3487983.3488302`. 48, 52, and 83

[192] Kay M Stanney, Mansooreh Mollaghasemi, Leah Reeves, Robert Breaux, and David A Graeber. Usability engineering of virtual environments (VEs): identifying multiple criteria that drive effective VE system design. *Int. J. Hum.-Comput. Stud.*, 58(4):447–481, April 2003. 13

[193] Yuchao Su, Jie Du, Yuanman Li, Xia Li, Rongqin Liang, Zhongyun Hua, and Jiantao Zhou. Trajectory Forecasting Based on Prior-Aware Directed Graph Convolutional Neural Network. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–13, 2022. ISSN 1558-0016. doi: 10.1109/TITS.2022.3142248. URL `10.1109/TITS.2022.3142248`. Conference Name: IEEE Transactions on Intelligent Transportation Systems. 10

[194] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for visual recognition. *TPAMI*, 2019. 68

[195] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2443–2451, Seattle, WA, USA, June 2020. IEEE. ISBN 978-1-72817-168-5. doi: 10.1109/CVPR42600.2020.00252. URL `https://ieeexplore.ieee.org/document/9156973/`. 46, 47, and 48

[196] Qiao Sun, Xin Huang, Junru Gu, Brian C Williams, and Hang Zhao. M2I: From factored marginal trajectory prediction to interactive prediction. February 2022. 48

[197] Hiromu Taketsugu, Takeru Oba, Takahiro Maeda, Shohei Nobuhara, and Norimichi Ukita. Physical plausibility-aware trajectory prediction via locomotion embodiment. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 70

[198] Kojiro Takeyama, Yimeng Liu, and Misha Sra. Divr: Incorporating context from diverse vr scenes for human trajectory prediction. 2025. 14

[199] Kojiro Takeyama, Yimeng Liu, and Misha Sra. Locovr: Multiuser indoor locomotion dataset in virtual reality. In *ICLR*, 2025. 14

[200] Chek Tien Tan, Leon Cewei Foo, Adriel Yeo, Jeannie Su Ann Lee, Edmund Wan, Xiao-Feng Kenan Kok, and Megani Rajendran. Understanding user experiences across VR walking-in-place locomotion methods. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, number Article 517 in CHI '22, pages 1–13, New York, NY, USA, April 2022. Association for Computing Machinery. 13

[201] Yichuan Charlie Tang and Ruslan Salakhutdinov. Multiple futures prediction. *arXiv preprint arXiv:1911.00997*, 2019. 10

[202] Izzeddin Teeti, Aniket Thomas, Munish Monga, Sachin Kumar, Uddeshya Singh, Andrew Bradley, Biplab Banerjee, and Fabio Cuzzolin. Astra: A scene-aware transformer-based model for trajectory prediction. 2025. 47

[203] Wei Zhen Teoh. Coherent multi-agent trajectory forecasting in team sports with CausalTraj. *arXiv preprint arXiv:2511.18248*, 2025. 46 and 62

[204] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Chenxu Hu, Yang Wang, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. DriveVLM: The convergence of autonomous driving and large vision-language models. *arXiv preprint arXiv:2402.12289*, 2024. 103

[205] Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and Christoffer Petersson. NeuRAD: Neural rendering for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 4

[206] Tram Thi Minh Tran, Callum Parker, and Martin Tomitsch. A review of virtual reality studies on autonomous Vehicle–Pedestrian interaction. *IEEE Transactions on Human-Machine Systems*, 51(6):641–652, December 2021. 13

[207] Nico Uhlemann, Yipeng Zhou, Tobias Simeon Mohr, and Markus Lienkamp. Snapshot: Towards application-centered models for pedestrian trajectory prediction in urban traffic environments. In *WACV*, 2025. 12

[208] Martin Usoh, Ernest Catena, Sima Arman, and Mel Slater. Using presence questionnaires in reality. *Presence*, 9(5):497–503, October 2000. 26 and 28

[209] Jur van den Berg, Stephen J Guy, Ming Lin, and Dinesh Manocha. Reciprocal n-body collision avoidance. In *Springer Tracts in Advanced Robotics*, Springer tracts in advanced robotics, pages 3–19. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. 12

[210] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL `https://papers.nips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html`. 11

[211] Edward Vendrow, Roberto Martin-Martin, Silvio Savarese, and Hamid Rezatofighi. Jrdb-pose3d: A multi-person 3d human pose and shape estimation dataset for robotics. 2025. 69

[212] Mingyi Wang, Hongqun Zou, Yifan Liu, You Wang, and Guang Li. A joint prediction method of multi-agent to reduce collision rate. *arXiv preprint arXiv:2411.07612*, 2025. 46

[213] Shuai Wang, Yunpeng Li, and Dongdong Zhang. Dtdnet: Dynamic target driven network for pedestrian trajectory prediction. *Frontiers in Neuroscience*, 18:1346374, 2024. 70

[214] Yichen Wang, Yiyi Zhang, Xinhao Hu, Li Niu, Jianfu Zhang, Yasushi Makihara, Yasushi Yagi, Pai Peng, Wenlong Liao, Tao He, Junchi Yan, and Liqing Zhang. Pedestrian motion reconstruction: A large-scale benchmark via mixed reality rendering with multiple perspectives and modalities. In *International Conference on Learning Representations (ICLR)*, 2025. 14

[215] Zikang Wang, Jingke Li, and Zikang Zhou. Hstr: Hierarchical scene transformer for multi-agent trajectory prediction. *arXiv preprint arXiv:2406.11741*, 2024. 72

[216] Waymo. Waymo Open Dataset Challenge: Motion prediction. `https://waymo.com/open/challenges/2021/motion-prediction/`, 2021. 48

[217] Waymo. Waymo Open Dataset Challenge: Motion prediction. `https://waymo.com/open/challenges/2022/motion-prediction/`, 2022. 48

[218] Waymo. Waymo Open Dataset Challenge: Motion prediction. `https://waymo.com/open/challenges/2023/motion-prediction/`, 2023. 48

[219] Waymo. Waymo Open Dataset Challenge: Motion prediction. `https://waymo.com/open/challenges/2024/motion-prediction/`, 2024. 48

[220] Waymo. Waymo Open Dataset Challenge: Interaction prediction. `https://waymo.com/open/challenges/2025/interaction-prediction/`, 2025. 48

[221] Waymo Research. Waymo open dataset for end-to-end driving in challenging long-tail scenarios. 2025. 12

[222] Florian Weber, Ronee Chadowitz, Kathrin Schmidt, Julia Messerschmidt, and Tanja Fuest. Crossing the street across the globe: A study on the effects of eHMI on pedestrians in the US, germany and china. In *HCI in Mobility, Transport, and Automotive Systems*, pages 515–530. Springer International Publishing, 2019. 13

[223] Erica Weng, Hana Hoshino, Deva Ramanan, and Kris Kitani. Joint metrics matter: A better standard for trajectory forecasting, 2023. 36 and 44

[224] Erica Weng, Kenta Mukoya, Deva Ramanan, and Kris Kitani. Evaluating a VR system for collecting safety-critical vehicle-pedestrian interactions. In *Data Generation for Robotics Workshop at RSS*, 2024. 23

[225] Xinshuo Weng, Ye Yuan, and Kris Kitani. Joint 3d tracking and forecasting with graph neural network and diversity sampling. *arXiv preprint arXiv:2003.07847*, 2020. 10

[226] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for Self-Driving perception and forecasting. January 2023. 11 and 16

[227] Bob G Witmer and Michael J Singer. Measuring presence in virtual environments: A presence questionnaire. *Presence*, 7(3):225–240, June 1998. 25, 26, and 28

[228] Conghao Wong, Beihao Xia, Ziming Hong, Qinmu Peng, Wei Yuan, Qiong Cao, Yibo Yang, and Xinge You. View vertically: A hierarchical network for trajectory prediction via fourier spectrums. In *Computer Vision – ECCV 2022*, pages 682–700. Springer Nature Switzerland, 2022. ix, 1, 10, 47, 55, 57, 63, 64, and 97

[229] Conghao Wong, Beihao Xia, Ziming Zou, Yuxin Wang, and Xinge You. Socialcircle: Learning the angle-based social interaction representation for pedestrian trajectory prediction. In *CVPR*, 2024. 47

[230] Wei Wu, Xiaoxin Feng, Ziyan Gao, and Yongkang Kan. Smart: Scalable multi-agent real-time motion generation via next-token prediction. In *ICLR*, 2025. 10 and 47

[231] Lisheng Xiong, Ziad Al-Halah, and Kristen Grauman. Pre-training a density-aware pose transformer for robust LiDAR-based 3D human pose estimation. *arXiv preprint arXiv:2412.13454*, 2024. 69

[232] Chenxin Xu, Weibo Mao, Wenjun Zhang, and Siheng Chen. Remember intentions: Retrospective-Memory-based trajectory prediction. March 2022. 10, 57, and 64

[233] Xiaolong Xu et al. Towards intelligent transportation with pedestrians and vehicles in-the-loop: A surveillance video assisted federated digital twin framework. *arXiv*

preprint arXiv:2503.04170, 2025. 12

[234] Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street Gaussians: Modeling dynamic urban scenes with Gaussian splatting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024. 4

[235] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything V2. *NeurIPS*, 2024. 102

[236] Xitong Yang, Devansh Kukreja, Don Pinkus, Anushka Sagar, Taosha Fan, Jinhyung Park, Soyong Shin, Jinkun Cao, Jiawei Liu, Nicolas Ugrinovic, Matt Feiszli, Jitendra Malik, Piotr Dollár, and Kris Kitani. SAM 3d body: Robust full-body human mesh recovery. 2025. Meta Superintelligence Labs. 69 and 75

[237] Yang Yang, Liang Qiao, Yanlun Gu, Mingjie Guo, and Ming Liu. Smartpretrain: Model-agnostic and dataset-agnostic representation learning for motion prediction. In *ICLR*, 2025. 10

[238] Ze Yang, Yun Chen, Jingkang Wang, Sivabalan Manivasagam, Wei-Chiu Ma, Anqi Joyce Yang, and Raquel Urtasun. UniSim: A neural closed-loop sensor simulator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 4 and 12

[239] Yu Yao, Ella Atkins, Matthew Johnson-Roberson, Ram Vasudevan, and Xiaoxiao Du. BiTraP: Bi-Directional pedestrian trajectory prediction with Multi-Modal goal estimation. *IEEE Robotics and Automation Letters*, 6(2):1463–1470, April 2021. 10

[240] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-Temporal graph transformer networks for pedestrian trajectory prediction. In *Computer Vision – ECCV 2020*, pages 507–523. Springer International Publishing, 2020. 10

[241] Tianyuan Yuan, Yicheng Liu, Yue Wang, Yilun Wang, and Hang Zhao. StreamMapNet: Streaming mapping network for vectorized online HD map construction. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024. 71

[242] Ye Yuan and Kris Kitani. Diverse trajectory forecasting with determinantal point processes. *arXiv preprint arXiv:1907.04967*, 2019. 1, 10, and 47

[243] Ye Yuan and Kris Kitani. DLow: Diversifying latent flows for diverse human motion prediction. March 2020. 10

[244] Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision*, pages 346–364. Springer, 2020. 55

[245] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris Kitani. AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting. *arXiv:2103.14023*

[cs], October 2021. URL http://arxiv.org/abs/2103.14023. arXiv: 2103.14023. ix, xi, 1, 10, 35, 47, 54, 56, 57, 63, 64, and 65

[246] Jiangbei Yue, Dinesh Manocha, and He Wang. Human trajectory prediction via neural social physics. July 2022. 10 and 47

[247] Kaiyuan Zhai, Juan Chen, Chao Wang, Zeyi Xu, and Guoming Tang. Trustworthy pedestrian trajectory prediction via pattern-aware interaction modeling. 2025. 47

[248] Haicheng Zhang, Yonghao Chen, Wenjia Ding, Zhenning Li, and Chen Lv. Mftraj: Map-free, behavior-driven trajectory prediction for autonomous driving. In *IJCAI*, 2024. 72

[249] Qingzhao Zhang, Shengtuo Hu, Jiachen Sun, Qi Alfred Chen, and Z Morley Mao. On adversarial robustness of trajectory prediction for autonomous vehicles. pages 15159–15168, January 2022. 53

[250] Wei Zhang, Ming Liu, and Xiaolin Chen. Sceneaware: Scene-constrained pedestrian trajectory prediction with llm-guided walkability. *arXiv preprint arXiv:2506.14144*, 2025. 71

[251] Zhijian Zhang, Zhaoyang Zhu, Hang Gao, and Hang Zhao. Map-free end-to-end trajectory prediction in bird's-eye view with deformable attention and sparse goal proposals. 2025. 72

[252] Hang Zhao, Jiyang Gao, Tian Lan, Chen Sun, Benjamin Sapp, Balakrishnan Varadarajan, Yue Shen, Yi Shen, Yuning Chai, Cordelia Schmid, Congcong Li, and Dragomir Anguelov. TNT: Target-driveN Trajectory Prediction. *arXiv:2008.08294 [cs]*, August 2020. URL http://arxiv.org/abs/2008.08294. arXiv: 2008.08294. 1, 10, and 47

[253] He Zhao and Richard P Wildes. Where are you heading? dynamic trajectory prediction with expert goal examples. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7629–7638. IEEE, October 2021. 10

[254] Tianyang Zhao, Yifei Xu, Mathew Monfort, Wongun Choi, Chris Baker, Yibiao Zhao, Yizhou Wang, and Ying Nian Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12126–12134, 2019. 10

[255] Zhengqiu Zheng et al. An immersive digital twin framework for exploring human-autonomy coexistence in urban transportation systems. *arXiv preprint arXiv:2406.05465*, 2025. 14

[256] Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. DrivingGaussian: Composite Gaussian splatting for surrounding dynamic autonomous driving scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 4

[257] Yi Zhou, Hui Zhang, Jiaqian Yu, Yifan Yang, Sangil Gu, Seung-In Oh, Byung-In Sun, and Liangjun Li. Himap: Hybrid representation learning for end-to-end vectorized hd map construction. In *CVPR*, 2024. 71

[258] Zikang Zhou, Luyao Ye, Jianping Wang, Kui Wu, and Kejie Lu. Hivt: Hierarchical vector transformer for multi-agent motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8823–8833, 2022. 74, 79, 81, 84, and 85

[259] Zikang Zhou, Jianping Wang, Yung-Hui Li, and Yu-Kai Huang. Query-centric multi-agent motion prediction. In *CVPR*, 2024. 10 and 47

[260] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. *ICCV*, 2023. 68

[261] Zhizheng Zhu et al. Learning to generate diverse pedestrian movements from web videos with noisy labels. In *International Conference on Learning Representations (ICLR)*, 2025. 14

[262] Katja Zibrek and Rachel McDonnell. Social presence and place illusion are affected by photorealism in embodied VR. In *Proceedings of the 12th ACM SIGGRAPH Conference on Motion, Interaction and Games*, number Article 13 in MIG '19, pages 1–7, New York, NY, USA, October 2019. Association for Computing Machinery. 13