

LookOut: Diverse Multi-Future Prediction and Planning for Self-Driving

Alexander Cui ^{*1,2}, Sergio Casas ^{*1,2}, Abbas Sadat ^{*1}, Renjie Liao ³, Raquel Urtasun^{1,2}

^{*}Waabi¹, University of Toronto², Google Brain³

{acui, sergio, asadat, urtasun}@waabi.ai, rjliao@google.com

Abstract

In this paper, we present LOOKOUT, a novel autonomy system that perceives the environment, predicts a diverse set of futures of how the scene might unroll and estimates the trajectory of the SDV by optimizing a set of contingency plans over these future realizations. In particular, we learn a diverse joint distribution over multi-agent future trajectories in a traffic scene that covers a wide range of future modes with high sample efficiency while leveraging the expressive power of generative models. Unlike previous work in diverse motion forecasting, our diversity objective explicitly rewards sampling future scenarios that require distinct reactions from the self-driving vehicle for improved safety. Our contingency planner then finds comfortable and non-conservative trajectories that ensure safe reactions to a wide range of future scenarios. Through extensive evaluations, we show that our model demonstrates significantly more diverse and sample-efficient motion forecasting in a large-scale self-driving dataset as well as safer and less-conservative motion plans in long-term closed-loop simulations when compared to current state-of-the-art models.

1. Introduction

Self-driving vehicles (SDVs) have the potential to enhance considerably the safety of our roads as, unlike humans, they can constantly scan the surrounding environment without getting distracted or being impaired while driving. Key to the success of a self-driving vehicle is its ability to perceive its surroundings and predict the future trajectory of the traffic participants, particularly those that might affect its decision making. These predictions are then exploited by the motion planning module to plan a safe and comfortable maneuver towards the goal.

Forecasting the behavior of traffic participants is very challenging as humans do not always follow the rules of the road and sometimes exhibit erratic behaviors. Furthermore, the scene might unroll in many possible ways in the

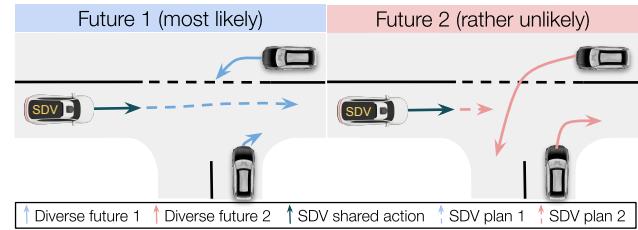


Figure 1. We illustrate the fact that the future is highly uncertain and multimodal by showing 2 distinct futures at the scene-level. In such scenario, LOOKOUT plans a short-term executable action that leads to 2 different contingent plans to stay safe in both cases.

future, depending heavily on the interactions between actors (e.g., at a merge, either actor A yields to actor B or vice versa). While most works predict each actor's future independently [11, 10, 29, 8, 4], recent approaches model actor interactions and can produce samples that explain the full scene in a consistent manner [2, 33, 38, 5]. However, they require prohibitively large numbers of samples to cover the long-tails of the distribution. This is problematic since these long tails are critical for safety, as failing to take them into account might result in an accident (e.g., an impaired driver running a red traffic light perpendicularly to the SDV's intended trajectory). Thus, there is a need to develop prediction systems that can efficiently sample the diverse set of possible futures. Unfortunately, existing approaches [46, 47] are not sample efficient as they trivially encourage diversity in euclidean space, thus utilizing samples to cover irrelevant actors or actions that do not impact the SDV's behavior.

Furthermore, existing motion planners cannot take advantage of prediction systems that produce scene-consistent samples [53, 13, 1, 36]. Instead, they optimize the expected cost by sampling the marginal distribution of each actor independently, thus ignoring the fact that some of these futures cannot happen at the same time (e.g., either the horizontal or vertical traffic can flow at a 4-way stop, but not both). These planners also assume that the SDV must commit to a single long-term trajectory, when in practice it can execute a short-term action and re-plan as newer sensor observations become available. As a consequence, they result in suboptimal and overly conservative trajectories [50, 39],

^{*}Denotes equal contribution

[†]This work was done by all authors while at Uber ATG

e.g., the SDV braking prematurely to react to an unlikely future instead of maintaining its speed as long as it is able to stop safely and comfortably later.

In this paper we propose LOOKOUT, a full end-to-end autonomy system that detects actors in the scene, predicts a diverse set of consistent futures with high sample efficiency, and plans an action that behaves defensively to potential hazards while not overreacting to low probability dangers far into the future. In particular, to address the sample inefficiency and limited mode coverage in motion forecasting we formulate this task as a *diverse set prediction problem*, where each element in the set reflects one possible future at the scene level. To enable this set to cover the future modes that matter for our decision making, we directly optimize the diversity of the downstream ego-vehicle motion plans. Then, a scenario scoring module estimates the probability of each future in the set, enabling our planner to account for unlikely but safety-critical scenarios without being overly conservative. Finally, we propose a novel contingency planner that is able to leverage multiple consistent futures by planning separate long-term responses for each future, while sharing an initial short-term action that behaves non-conservatively with respect to the futures and avoids immediate collision. Fig. 1 shows an example of two diverse futures and the corresponding shared action and contingent plans.

We demonstrate the effectiveness of our approach in large-scale open-loop and closed-loop experiments that comprise a wide variety of complex scenarios. Our extensive experiments show that LOOKOUT’s driving is significantly safer as well as less conservative than previous state-of-the-art approaches. Furthermore, there exists a trade-off between diversity and reconstruction quality of the forecasts; our approach can produce much better reconstruction than other methods with similar diversity and higher diversity with similar reconstruction capability.

2. Related Work

The autonomy pipeline composed of cascading detection, motion forecasting, and motion planning modules offers great advantages over black-box end-to-end models [30, 3, 9, 20, 26] such as safety, interpretability, error tracing, and data efficiency. Moreover, [23, 6, 48, 35, 49, 7, 40] have recently shown that learning this interpretable pipeline end-to-end is beneficial. Because of this, we focus our literature review on interpretable approaches. For object detection, we simply leverage recent advances in 3D voxel-based object detection from LiDAR point clouds [22, 12, 44, 52, 43], which have been shown to achieve great speed-accuracy tradeoffs. In the following paragraphs, we dive deep into recent advances in motion forecasting and motion planning, given that the main contributions of our work reside on these modules.

Motion Forecasting: A common approach for actor modeling has been to independently predict the trajectory of each actor [32, 11, 4, 8, 19, 29, 51]. These predictions can be represented as closed-form gaussian distributions [11, 4, 8], a classification or energy over a discrete grid/graph/set structure [19, 51, 29, 49], or trajectory samples of a stochastic model [32, 17]. One approach to tractably model the traffic multimodality jointly across actors is to stochastically sample one possible future scenario at a time, by sampling latent variables that encode the joint scene dynamics, and then decode the future trajectories [33, 38, 5]. These are mainly divided into autoregressive models [33, 38], and implicit latent variable models [5]. However, these methods require a high number of samples to characterize the scene. In contrast, work in diverse motion forecasting has focused on achieving high sample-efficiency to cover the main modes of the distribution. This is especially important in self-driving as SDVs need to be able to anticipate rare or dangerous behavior by other actors on the road in order to plan safe responses. Recent work [46, 47] has explored how to encourage more diverse predictions from pretrained variational inference models [37]. They train new encoders that output a fixed number of jointly diverse samples of latent variables. The formulation in [46] directly outputs the set of latent codes, and evaluates their diversity based on determinantal point processes (DPP). In the work of [47], a set of multivariate gaussian distributions are sampled jointly via reparameterization trick with a shared noise, and a diversity loss based on the L2-distance between motion forecast samples is used to increase diversity in the predictions. Alternatively, [18] trains a conditional GAN to output diverse samples using Farthest Point Sampling on the latent space to spread out over more modes of the latent space. Finally, [28] trains their trajectory samples to stay within the drivable area, allowing for greater diversity while retaining admissibility. While these works achieve greater diversity and accuracy in motion prediction, it is unclear how these improvements translate into better motion planning for autonomous agents.

Planning: In motion planning the goal is to generate a trajectory for the self-driving vehicle to drive safely, comfortably, and progressing toward the goal [27]. A popular approach to achieve this task is to design a cost function that encodes all the objectives above and find a minimum-cost trajectory. Such optimizations have been solved using continuous-optimization [53], sampling [36, 35], or search [1]. These methods achieve safety by including a collision cost in the objective function which is computed with respect to the predicted trajectories of actors in the scene. However, in probabilistic settings where the predictions take the form of trajectory distributions, the above methods compute the collision cost in expectation, minimizing

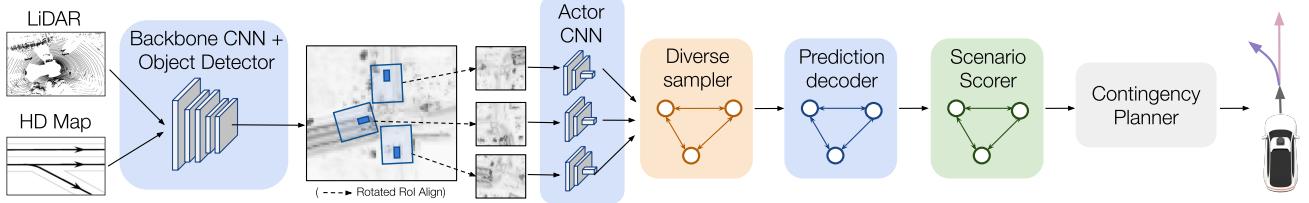


Figure 2. LOOKOUT inference. For the learnable components, the colors denote different training stages. The [backbone CNN](#), [actor CNN](#) and [prediction decoder](#) are first trained (Section 3.1), the [diverse sampler](#) next (Section 3.2), and lastly the [scenario scorer](#) (Section 3.3)

the expected cost over all future predicted scenarios with a single plan. Given a set of joint diverse predictions, it is possible that the SDV will need to plan for a greater number of unlikely, but safety-critical future scenarios that will require it drive defensively (e.g. yield, change lanes). However, it is undesirable to always brake preemptively for such rare scenarios, or ignore them altogether. Work in optimization-based motion planning [45, 15, 50, 39, 31] plan for such rare scenarios by picking trajectory plans that ensure it can react to them safely, while also optimizing for objectives such as progress and comfort. [15] splits the planned trajectory into an initial shared section, and a set of branched plans that could be taken from the end of the shared section. [50, 39] predict the probability that a defensive maneuver will be necessary in the near future, and decide whether to postpone the decision to the future (where more information will be available). The concurrent work of [31] learns an autoregressive contingency planner that branches flexibly, and models the surrounding agents as reactive. However, the online policy optimization could converge to an unsafe, local optima.

3. Diverse Prediction and Planning

In this section, we break down the autonomy problem of mapping sensor data to an executable action into several modules which provide interpretability of the SDV decision making. Towards this goal, we first learn a joint perception and future prediction model that detects relevant objects and estimates the joint distribution over all actors’ future trajectories with an implicit latent variable model [5] (Section 3.1). Despite its sample inefficiency, such generative model allows us to learn a very powerful and efficient trajectory decoder from latent samples. Next, we leverage this decoder to learn a diverse latent sampler that achieves high sample efficiency from the planner’s perspective (Section 3.2). Then, we estimate the probability of each future realization in the set (Section 3.3). Finally, we design a novel contingency planner that plans a safe trajectory for each possible future without being overly cautious (Section 3.4). Fig. 2 depicts an overview of our approach.

3.1. Joint Perception and Motion Forecasting

In order to extract features useful for both detection and motion forecasting, we employ a convolutional *backbone network* inspired by [44, 6], which takes as input a history of voxelized LiDAR sweeps and a raster HD map, both in

bird’s eye view (BEV) centered around the SDV. We then perform multi-class object detection with a shallow convolutional header to recognize the presence, BEV pose and dimensions of vehicles, pedestrians and bicyclists, and apply rotated ROI align [24] to extract small feature crops from the scene context around each actor’s location. Finally, an *actor CNN* with max-pooling reduces the feature map of each actor n into a feature vector, x_n^{local} . Since this local context lacks global information about the actor’s pose with respect to the rest of the scene, we include the BEV centroid and rotation relative to the SDV $x_n^{global} = \{c_{x,n}, c_{y,n}, a_n\}$ as additional features, obtaining the final actor context $x_n = [x_n^{local}, x_n^{global}] \in \mathbb{R}^D$, where $[\cdot, \cdot]$ denotes channel-wise concatenation. We refer to the set of all the detected actors’ contexts as $X = \{x_1, x_2, \dots, x_N\}$. The details about the LiDAR and map parameterization as well as the backbone network, object detector header, and actor CNN are left for the supplementary materials as they are not the focus of our work and are highly inspired by previous literature [23, 6, 4].

We parameterize the trajectory of each actor with a temporal series of the actor centroid in 2D Euclidean space, i.e., $y_n \in \mathbb{R}^{2T}$, where each trajectory is predicted in the actor’s relative coordinate frame in Bird’s Eye View (BEV) defined by its centroid and heading. Our latent variable model then characterizes the joint distribution over actors’ trajectories as follows:

$$p(Y|X) = \int_Z p(Y|X, Z)p(Z|X)dZ, \quad (1)$$

where $Z = \{z_1, z_2, \dots, z_N\}$ is a set of continuous latent variables that capture latent scene dynamics, and $Y = \{y_1, y_2, \dots, y_N\}$ is the future trajectories of all actors. We assume a fixed prior $p(Z|X) \approx p(Z) = \prod_{n=1}^N p(z_n)$, where $z_n \sim \mathcal{N}(0, I) \in \mathbb{R}^L$. Following [5], we adopt an implicit¹ decoder $Y = f_\theta(X, Z)$, where f_θ is a deterministic function parameterized by a spatially-aware Graph Neural Network (GNN) [4]. Since from observational data we only obtain (X, Y) pairs, a posterior or encoder function q_ϕ is introduced to approximate the true posterior distribution $p(Z|X, Y)$ during training [37], also parameterized by a GNN. This encoder function helps this model learn a powerful decoder, since given only X there could be many feasible Y due to the inherent multimodality and uncertainty of the future.

¹“Implicit” means $p(Y|X, Z)$ does not have analytical form.

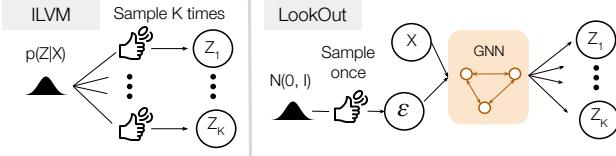


Figure 3. Obtaining K latent samples from an implicit latent variable model (ILVM) implies sampling K times independently from the prior. In contrast, our diverse sampler exploits a GNN mapping to predict K latent samples from a single noise (in parallel).

The backbone network, detection header, actor CNN, encoder, and decoder are trained jointly for the tasks of object detection and motion forecasting. We use binary cross-entropy with hard negative mining per class for the presence of an actor, and Huber loss for the regression targets (i.e., pose and dimension) [44]. See the supplementary materials for more details. We use the CVAE framework [37] for the latent variable model, which optimizes the evidence lower bound (ELBO) of the log-likelihood $\log p(Y|X)$. Because the deterministic decoder leads to an implicit distribution over Y , we use Huber loss ℓ_δ as the reconstruction loss [5], and reweight the KL term with β as proposed by [16]:

$$\mathcal{L}_{\text{forecast}} = \frac{1}{NT} \sum_n^N \sum_t^T \ell_\delta(y_n^t - y_{n,GT}^t) + \beta \cdot \text{KL}(q_\phi(Z|X, Y = Y_{GT}) \| p(Z)), \quad (2)$$

where the first term minimizes the reconstruction error between the trajectory samples $Y = \{y_n^t | \forall n, t\} = f_\theta(X, Z)$, $Z \sim q_\phi(Z|X, Y = Y_{GT})$ and their corresponding ground-truth Y_{GT} , and the second term brings the privileged posterior $q_\phi(Z|X, Y = Y_{GT})$ and the prior $p(Z)$ closer.

So far we have learned a powerful model of the future from which we can generate scene consistent samples for all actors in the scene, largely similar to [5]. In particular, inference in this model works as follows: First, we encode the sensor data into actor contexts X . Then, we sample K times from the prior $\{Z_k \sim p(Z) | \forall k\}$, and decode the scene latent samples deterministically in parallel to obtain each of the K futures $\{Y_k = f_\theta(X, Z_k) | \forall k\}$. Despite the high expressivity of this model and its attractive parallel sampling, it has two major drawbacks: (i) *sample inefficiency*, and (ii) *no closed-form likelihood*; respectively causing increased reaction times and lack of downstream probabilistic risk assessment. In the following, we address (i) by learning to sample jointly a diverse set of latent codes that map into a covering distribution over trajectories and, (ii) by learning a categorical distribution over the diverse futures in the set.

3.2. Planning-Centric Diverse Sampler

The goal here is to remediate the sample inefficiency of the scene-level generative model presented in Section 3.1 while exploiting its expressivity. To do so, we learn a *diverse sampling function* $\mathcal{M} : X \mapsto \mathbf{Z}$ that maps the actor contexts X coming from sensor data around each actor into a compact set of scene latent samples $\mathbf{Z} = \{Z_1, \dots, Z_K\}$

whose decoded trajectories \mathbf{Y} achieve good coverage. This sampler will then replace the Monte Carlo sampling from the prior $p(Z)$ during inference, as illustrated in Fig. 3.

To leverage the decoder trained in Sec. 3.1, which was trained to decode samples from a Gaussian approximate posterior, the distribution over the set of latents induced by the diverse sampler should also be Gaussian to reduce the distributional shift [47]. Thus, we assume $p(\mathbf{Z}|X) = \prod_{k=1}^K p(Z_k|X)$ where $p(Z_k|X) = \mathcal{N}(\mu_k, \Sigma_k)$, $\mu_k \in \mathbb{R}^{NL}$, and $\Sigma_k \in \mathbb{R}^{NL \times NL}$. To sample a set of latents \mathbf{Z} that are distinct enough from each other such that they will be decoded into a set of diverse futures, we use the reparameterization trick [21] to map a shared noise $\varepsilon \sim \mathcal{N}(0, I) \in \mathbb{R}^{NL}$ across K latent mappings $\{\mathcal{M}_{\eta_k} | k \in 1 \dots K\}$:

$$Z_k = \mathcal{M}_{\eta_k}(X, \varepsilon) = b_{\eta_k}(X) + A_{\eta_k}(X)\varepsilon, \quad (3)$$

where $\eta = \{\eta_k | \forall k\}$ is the set of learnable parameters, $\mu_k = b_{\eta_k}(X)$, and $\Sigma_k = A_{\eta_k}(X)A_{\eta_k}(X)^T$.

To handle the fact that the input $X \in \mathbb{R}^{ND}$ can vary in size (i.e. the number of actors N varies from scene to scene), we parameterize \mathcal{M} with a pair of GNNs: one to generate the means and another to generate the covariances. Both GNNs assume a fully connected graph where each node is anchored to an actor, and initialize the node states as $\{x_n\}$. Then, we perform message passing to aggregate information over the whole scene at each node. Finally, each node in the first GNN predicts $a_n \in \mathbb{R}^{KL}$ via an MLP. Then, we can easily extract $A_{\eta_k}(X) = \text{diag}([a_1^{kL:(k+1)L}, \dots, a_N^{kL:(k+1)L}])$. Similarly, each node in the second GNN predicts $b_n \in \mathbb{R}^{KL}$ via another MLP, and $b_{\eta_k}(X) = [b_1^{kL:(k+1)L}, \dots, b_N^{kL:(k+1)L}]$.

The diverse latent codes \mathbf{Z} can then be deterministically decoded via $Y_k = f_\theta(X, Z_k)$ with the decoder learned in Section 3.1. Through sampling and decoding, we obtain a set of K future trajectory realizations of all actors in the scene $\mathbf{Y} = \{Y_1, \dots, Y_K\}$. This process is parallel since it is performed by leveraging a pair of GNNs that perform all K latent mapping in a single round of message passing $\mathbf{Z} \sim \mathcal{M}(X, \varepsilon; \eta)$. Then, we can batch the K latent samples to decode them in parallel $\mathbf{Y} = f_\theta(\mathbf{Z}, X)$.

The objective of this diverse sampler is to be able to generate a set of futures \mathbf{Y} that are diverse while recovering well the ground-truth observations Y_{gt} , which we can express through an energy $E(\mathbf{Y}, Y_{gt})$. Moreover, to encourage minimal distribution shift to the inputs of the pretrained decoder f_θ , we also minimize the KL divergence between all the diverse latent distributions $p(Z = Z_k|X)$ and the prior distribution $p(Z)$. In practice, this term makes the learning much more stable. To find the right balance between these two objectives, we add a hyperparameter β . Overall, the minimization can be formulated as:

$$\min_{\eta} E(\mathbf{Y}, Y_{gt}) + \beta \sum_{k=1}^K \text{KL}(p(Z_k|X) \| p(Z)), \quad (4)$$

where $\mathbf{Y} = \{Y_1, \dots, Y_K\}$, $Y_k = f_\theta(X, Z_k)$, $Z_k = \mathcal{M}_{\eta_k}(X, \varepsilon)$ and the minimization is with respect to learnable parameters of the pair of GNNs η . Note that the decoder is fixed, i.e., θ is not optimized. Note that learning the decoder jointly would compromise the realism to maximize planning diversity (e.g., by having agents suddenly appear in front of the SDV in the future, forcing it to change its plan). Our energy function is composed of a few terms that promote the diversity while preserving data reconstruction:

$$E(\mathbf{Y}, Y_{\text{gt}}) = E_r(\mathbf{Y}, Y_{\text{gt}}) + E_p(\mathbf{Y}) + E_d(\mathbf{Y}). \quad (5)$$

We now define the energy terms in more details.

Reconstruction Energy: This term encourages that what happened in reality at the time the log was recorded to be captured by at least one sample:

$$E_r(\mathbf{Y}) = \min_k \ell_2(Y_k - Y_{\text{gt}}). \quad (6)$$

Planning Diversity Energy: Increased prediction diversity helps anticipate distinct future scenarios that require different SDV plans (e.g., a vehicle cuts in front of the SDV vs. keeps driving on its original lane). Thus, we promote diverse samples that matter for the downstream task of motion planning by maximizing the following reward function:

$$R(\mathbf{Y}) = \frac{1}{K} \sum_{i=1}^K \sum_{j \neq i}^K \ell_2(\tau_i - \tau_j), \quad (7)$$

where $\tau_i = \tau(Y_i)$ refers to the SDV trajectory planned for predicted scene sample Y_i by our contingency motion planner outlined in Section 3.4. Since the optimal planned trajectory for each scene τ_i is not differentiable with respect to Y_i , we leverage the REINFORCE gradient estimator to express the energy E_p as a function of the log-likelihood under the diverse sampler

$$\begin{aligned} E_p(\mathbf{Y}) &= -\mathbb{E}_Y[R(\mathbf{Y})] \approx -\log p(\mathbf{Z}|X)R(\mathbf{Y}) \\ &= \frac{1}{K(K-1)} \sum_{i=1}^K \sum_{j \neq i}^K -\log p(Z_i, Z_j) \ell_2(\tau_i - \tau_j), \end{aligned} \quad (8)$$

where $\log p(Z_i, Z_j) = \log p(Z_j) + \log p(Z_i)$. The approximation comes from a Monte Carlo estimation of the marginalization over \mathbf{Z} .

General Diversity Energy: Since the signal from the planning-based diversity can be sparse for scenes that do not have any actors interacting with the SDV, we additionally encourage diversity in the behaviors of all actors:

$$E_d(\mathbf{Y}) = \frac{1}{K(K-1)} \sum_{i=1}^K \sum_{j \neq i}^K \exp\left(-\frac{\ell_2(Y_i - Y_j)}{\sigma_d}\right). \quad (9)$$

With our proposed diverse sampler, each \mathbf{Y} induced by a different noise ε efficiently covers well the distribution over futures. Thus, during inference we can simply take the set induced by the mode $\varepsilon = 0$ to eliminate all randomness. We note that determinism is important in self-driving for safety, verification, and reproducibility.

3.3. Scenario Probability Estimation

The diverse set of K future realizations $\mathbf{Y} = \{Y_1, \dots, Y_K\}$ provides the coverage needed for safe motion planning. However, for accurate risk assessment we need to estimate the probability distribution over each future realization in the set. To achieve this goal, we augment our model to also output a score for all future realizations $l = s_\psi(X, \mathbf{Y})$, where s_ψ is a GNN that takes as input the actor features and all K sample future scenarios. We can then easily recover a distribution over such scores by re-normalization. Thus, the probability of each sample is

$$p_\psi(Y_k|X) = \frac{\exp(l_k)}{\sum_{k'} \exp(l_{k'})}. \quad (10)$$

Since we only have access to a single ground truth realization (i.e., the one that occur in the training log), we train the scoring function s_ψ to match the approximate categorical distribution over future scenarios $q(Y_k|X)$ under the $\text{KL}(p_\psi \| q)$ divergence. We define this approximate distribution as follows:

$$q(Y_k|X) = \frac{\exp(-\alpha \ell_2(Y_k - Y_{GT}))}{\sum_{k'} \exp(-\alpha \ell_2(Y_{k'} - Y_{GT}))}, \quad (11)$$

where $\alpha = 10$ is a temperature hyperparameter we chose empirically. The research of better distance functions that can capture the relevance of each actor with respect to the SDV are left as future work.

3.4. Contingency Planner

The goal of the motion planning module is to generate safe, comfortable and not overly conservative trajectories for the SDV to execute. We achieve this through *Model Predictive Control*, where a trajectory is planned considering a finite horizon, and is executed until a new trajectory is replanned upon availability of a new LiDAR sweep. Most planning frameworks in the literature [27, 14, 1, 36] take an optimization-based approach where the trajectory that minimizes the expected cost is selected for execution:

$$\tau_{0:T}^* = \arg \min_{\tau_{0:T} \in \mathcal{T}_{0:T}(\mathbf{x}_0)} \mathbb{E}_{P(Y)} c(\tau_{0:T}, Y), \quad (12)$$

where $\mathcal{T}_{0:T}(\mathbf{x}_0)$ denotes the set of possible trajectories starting from SDV state \mathbf{x}_0 up to the horizon T , and c denotes the planner cost function. Note that the expectation is over the distribution of possible future realizations of all actors $P(Y)$. However, the above formulation does not exploit the fact that only one of the predicted scenarios will happen in the future and is conversely optimizing for a single trajectory that is "good" in expectation. Note that if we change the expectation in Eq. 12 to the *max* operator, the planner will optimize for the worst-case scenario regardless of its likelihood. Consequently the planner will become over-conservative, e.g., it will apply a hard-break for a very low probability scenario where a vehicle crosses SDV lane, as shown in [50, 39].

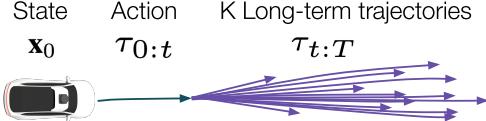


Figure 4. **Contingency planning paradigm.** The cost-to-go of a short term ego-action is captured by the ability to react to the K diverse predicted futures with the K most suitable ego-trajectories.

In this paper, we take a different approach where instead of finding a single motion plan for multiple futures, we generate a single common immediate action, followed by a set of future trajectories, one for each future realization of the scene, as shown in Fig. 4. This *contingency planning* paradigm finds an immediate action $\tau_{0:t}$ that is safe with respect to all the possible realizations in Y and comfortably bridges into a set of contingent trajectories, where each is specifically planned for a single future realization. Such decision-postponing avoids over-conservative behaviors while staying safe until more information is obtained. Importantly, this is only possible if the set of predicted future scenarios is diverse, and covers possible realizations, including low likelihood events.

Specifically, we plan a short-term trajectory that is safe with respect to all possible futures and allows a proper contingent plan for each future realization:

$$\tau_{0:t}^* = \arg \min_{\tau_{0:t} \in \mathcal{T}_{0:t}(x_0)} \left(\overbrace{\max_Y c(\tau_{0:t}, Y)}^{\text{action cost}} + \overbrace{\sum_{Y_i \in Y} p(Y_i) g(x_t, Y_i)}^{\text{cost-to-go}} \right) \quad (13)$$

where $g(x, Y) = \min_{\tau_{t:T} \in \mathcal{T}_{t:T}(x)} c(\tau_{t:T}, Y)$ represents the minimum cost trajectory from time t to T starting from the state x and assuming a single future realization Y .

Cost Function: The planner cost function $c(\cdot) = \sum_i w_i s_i(\cdot)$ is a linear combination of various carefully crafted subcosts s_i that encode different aspects of driving including safety, comfort, traffic-rules and the route. Here, $w = \{w_i | \forall i\}$ is a set of learnable parameters. However, learning these parameters in the contingency planning paradigm (Eq. 13) is an open problem since we only have expert demonstrations for the future that occurred at the time of the log. Thus, we leave this for future work, and leverage the weights learned through Eq. 12 by [36]. Regarding the subcosts, collision and safety-distance subcosts penalize SDV trajectories that overlap with the predicted trajectories of other actors or have high speed in close distance to them. Similarly, trajectories that violate a headway buffer to the lead vehicle are penalized. Other subcosts promote driving within the lane and road boundaries, and penalize trajectories that go above speed-limit or violate a red-traffic light. Finally, motion jerk, high forward acceleration, deceleration, and lateral acceleration of the trajectories are penalized to promote comfortable maneuvers. The details of all the subcosts can be found in the supplementary materials.

Inference: We take a sampling approach to solve the minimization in Eq. 13. Specifically, we generate a set of pairs $\{(\tau_{0:t}, \mathcal{T}_{t:T}(\tau_t))\}$, which include possible short-term trajectories $\tau_{0:t}$ and their possible subsequent set of trajectories $\mathcal{T}_{t:T}(\tau_t)$. It is important to consider a dense set of initial actions such that the final executed trajectory is smooth and comfortable. Similarly, a dense set of long-term trajectories enables the planner to find a proper contingent plan for the future and thus obtain a more accurate cost-to-go for the initial action. In order to manage the complexity of the search space above, we take the following sampling strategy: (i) first a set of (spatial) paths are generated, (ii) for each path, a set of initial velocity profiles are sampled, creating the set of short-term trajectories, (iii) conditioned on the end state of these initial trajectories, another set of velocity profiles are sampled for the rest of the planning horizon assuming the SDV follows the same path. In total, the sample set contains ≈ 240 actions and for each action there are ≈ 260 long-term trajectories. The above path and velocity generation are done in Frenet-frame of the desired lane center line, by sampling lateral and longitudinal profiles [42, 36]. For more details see the supplementary materials.

4. Experiments

In this section, we present empirical results to validate the hypotheses that our diverse sampler improves the diversity/reconstruction tradeoff in the motion forecasts, and that when tied with the contingency planner it can drive more safely and less conservatively than prior art.

4.1. Experimental Setup

Dataset: ATG4D is composed of over one million frames of LiDAR, HD maps with very accurate object tracks. It was collected with careful expert drivers in several North American cities. All models are trained to predict 5-second trajectories, given 1 second of LiDAR history. We evaluate motion forecasting in the test set of this dataset.

Closed-loop simulator: We use a simulated LiDAR environment [25] for closed-loop experiments where we evaluate the quality of our end-to-end driving model, recreated from real static environments and actors. These scenarios are curated from real driving logs to be particularly challenging, and they do not overlap with those in ATG4D in order to evaluate generalization. When replaying the scenario, the actors switch to reactive actors [41] if the scenario diverges from the original one due to SDV actions. The simulation is unrolled for ~ 18 seconds at intervals of 100 milliseconds, which is the same time it takes to acquire a new LiDAR sweep in the data collection vehicle. We note that all training happens on real offline data, but it transfers well to the simulated environment due to its high realism.

Model	CR(%)	$\frac{\text{Progress}}{\text{collision}}(m)$	Progress(m)	Jerk($\frac{m}{s^3}$)	Lat.Acc.($\frac{m}{s^2}$)	Acc($\frac{m}{s^2}$)	Decel($\frac{m}{s^2}$)
CVAE-DPP[46]	17.07	123.97	21.17	11.99	0.06	1.11	0.80
CVAE-DLow[47]	14.63	377.07	55.18	5.22	0.15	0.84	0.54
MultiPath[8]	12.20	394.37	48.09	12.92	0.13	1.24	0.80
CVAE[37]	8.54	655.22	55.93	7.22	0.15	0.96	0.62
ESP[33]	11.59	464.44	53.81	6.52	0.15	0.89	0.57
ILVM[5]	10.98	553.96	60.80	5.50	0.16	0.86	0.56
LOOKOUT	7.93	790.37	62.65	4.69	0.37	0.79	0.53

Table 1. **End-to-end driving results in closed-loop simulation.** All motion forecasting baselines use the PLT planner [36] (Eq. 12) as they don't propose a motion planner. Please see our supplementary materials for results when they are paired with our planner (Eq. 13).

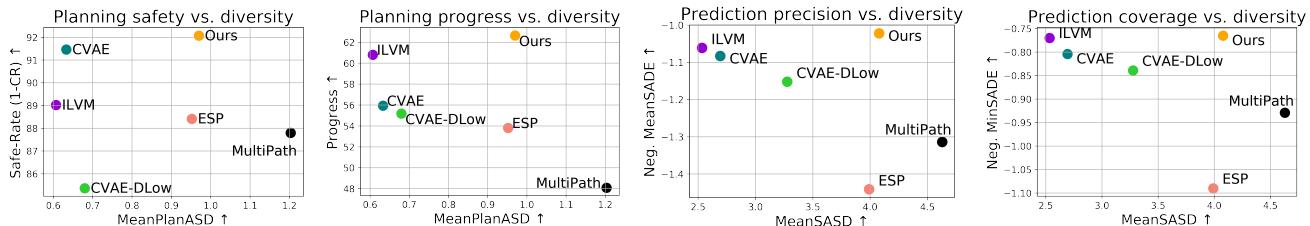


Figure 5. **Planning quality and prediction reconstruction as a function of diversity.** More diversity is not always better. We do not include CVAE-DPP in these visualizations for clarity, as it has much lower performance than other models and would be off-the-charts.

Baselines: For motion forecasting, we use state-of-the-art baselines in multimodal and diverse prediction, all of them trained end-to-end with the same backbone network and object detector architectures for a fair comparison, following the experimental setup in [4, 5]. *MultiPath* [8], *CVAE* [37], *CVAE-DPP* [46] and *CVAE-DLow* [47] model the distribution over each actor's future trajectories independently. Thus, to construct a scene sample for these baselines we sample a random trajectory for each actor, following [5]. For approaches that model the joint distribution over all actors' future trajectories, we benchmark against *ESP* [33] and *ILVM* [5], which performed the best when compared in [5]. To compare LOOKOUT's driving to the baselines, we use the state-of-the-art PLT planner [36] (Eq. 12) for those motion forecasting models that did not propose a planner.

End-to-end driving metrics (closed-loop): We measure the *collision rate (CR)* to reflect the driving safety. This is the percentage of simulations in which there is at least 1 collision between the SDV and another actor. We also evaluate the *progress* made by the SDV on its desired route throughout the simulation horizon, measured in meters from the starting location, as well as the progress per collision, giving an idea of the ratio between non-conservativeness and safety. Finally, we measure the mean jerk and acceleration applied as a metric of the driving comfort. As the autonomy unrolls its own actions for long time periods, potentially diverging from the path the expert-driver executed, these metrics capture the quality of the end-to-end system, including its robustness to distributional shift [34].

Sub-system level metrics (open-loop): In the open-loop evaluations, our model is evaluated on real data from the logs in the ATG4D dataset (i.e., the scenes visited by the

expert driver), as opposed to closed-loop evaluations where we unroll our own plans. To evaluate the object detection quality we measure the standard *mean-average precision (mAP)*, but defer the results to the supplementary because all the models share the same perception backbone architecture, and it is not the focus of this paper. To measure the reconstruction capability and the diversity of the scene-level motion forecasts, we use $K = 15$ scene samples, meaning that there are 15 distinct future scenarios predicted, each with 1 trajectory per actor. The *minimum scene average displacement error (minSADE)* measures how well we recall the ground-truth trajectory, while the *mean scene average displacement error (meanSADE)* measures the precision of the predicted distribution as proposed in [5]. To evaluate how the diversity of these predictions impact the subsequent contingent plans, we measure the pairwise plan average self-distance (*meanPlanASD*), i.e., the average distance between the contingent plans for 2 distinct futures. Finally, the scene average self-distance (*meanSASD*) computes the average pairwise distance among scene samples as a way to measure general diversity as proposed by [46, 47]. Details are available in the supplementary.

4.2. Comparison against state-of-the-art

Planning benchmark: The closed-loop experiment results for motion planning are shown in Table 1. LOOKOUT outperforms the baselines in almost all metrics. In particular, we see a 21% increase in progress per collision to the next best baseline for this metric, CVAE + PLT. This is a combination of having 8% fewer collisions in addition to 12% greater progress, showing our model is able to avoid dangerous scenarios on the road without slowing down (i.e., it provides additional safety while being less conservative). For completeness, the results of the baselines paired with our contingency planner are available in the supplementary.

ID	\mathcal{M}_η	E_p	s_ψ	Planner	CR(%)	$\frac{\text{Progress}}{\text{collision}}(m)$	Progress(m)	Jerk($\frac{m}{s^3}$)	Lat.Acc.($\frac{m}{s^2}$)	Acc($\frac{m}{s^2}$)	Decel($\frac{m}{s^2}$)
LOOKOUT	✓	✓	✓	Conting.	7.93	790.37	62.65	4.69	0.37	0.79	0.53
M_1	✗	N/A	✗	Conting.	9.15	709.60	64.90	4.40	0.38	0.77	0.52
M_2	✓	✗	✓	Conting.	10.98	626.79	68.79	8.77	0.20	1.09	0.73
M_3	✓	✓	✗	Conting.	9.15	658.58	60.24	4.96	0.35	0.79	0.53
M_4	✓	✓	✓	PLT	12.80	436.29	55.87	6.26	0.16	0.90	0.59

Table 2. **Ablation study** on the effect of the diverse sampler \mathcal{M}_η , planning diversity energy E_p , scenario scorer s_ψ and motion planner towards the end-to-end driving capability (evaluated in closed-loop simulations).

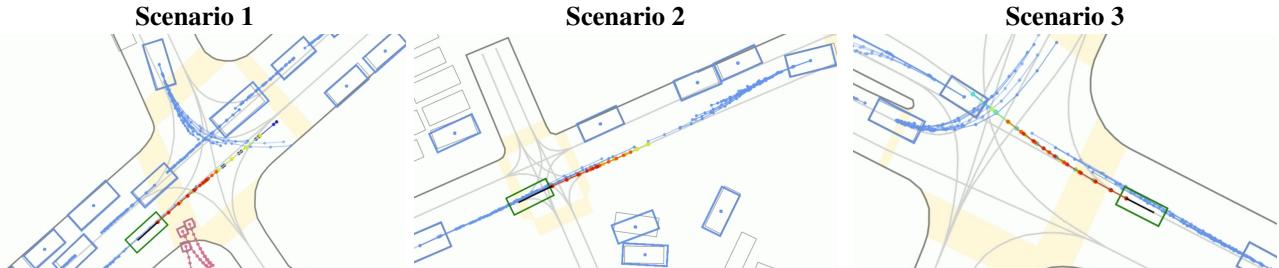


Figure 6. **Diverse multi-future predictions and plans** in closed-loop, zoomed in. Object detections and motion forecasts are blue for vehicles and pink for pedestrians. The green bounding box is the SDV, its immediate action (1s) is shown in black (starting from its rear axle), and its contingent trajectories planned for each possible future scenario are shown in distinct colors. LiDAR points are not visualized.

Diversity tradeoffs: LOOKOUT achieves the safest driving (Fig. 5a) and makes the most progress (Fig. 5b), while attaining a very high diversity among its contingent plans. In contrast, we see how more diversity often makes driving more unsafe for the baselines, while diminishing their progress. As seen in Fig. 5c and Fig. 5d, more diversity typically results in less precise and covering motion forecasts when evaluated at a constant budget of samples. Our method escapes this *diversity curse* by focusing on forecasting the future scenarios that matter for the downstream task of driving. This allows our model to make cautious, safe plans without the shortcomings of too much irrelevant or excessively variant predictions. We include the results of the baselines paired with the contingency planner in the supplementary materials.

4.3. Ablation Study

Table 2 shows the impact of our main contributions, providing insights into what makes LOOKOUT’s driving safer and less conservative.

Diverse sampler vs. Monte Carlo sampling: M_1 samples independently from the prior $p(Z)$. When comparing this to LOOKOUT, we see that the full diverse sampler (including the scenario scoring and planning diversity energy) is crucial to anticipate and avoid far more collisions, while attaining almost the same comfort and progress.

Planning diversity energy: M_2 shows that removing the planning diversity energy from the diverse sampler objective increases the collision rate by 36%. We also observe contrary fluctuations in jerk and lateral acceleration. We hypothesize that this energy term favors early and preventive lateral displacements instead of late hard brakes from the planner. Further investigation is left for future work.

Scenario scoring vs. uniform probabilities: M_3 removes the scenario scoring, assigning each diverse scenario an equal probability as input to the planner. We can see that scenario scoring improves safety and progress, showing us that it prevents the SDV from unnecessary premature braking to avoid low-probability risks.

Contingency planner vs. PLT: The ablation M_4 demonstrates the importance of the contingency planner as it improves almost every metric when compared to the PLT planner, notably reducing collisions by 38%.

4.4. Qualitative results

Figure 6 shows three challenging scenarios the SDV encountered while driving in closed-loop simulation. We can see in each scenario that the SDV plans multiple contingent trajectories that each respond safely to one of the predicted futures. Thus, the SDV can take a non-conservative immediate action and still find a safe future trajectory if any of the on-coming or turning cars block its path.

5. Conclusion

We have proposed a prediction and planning model that generates more diverse motion forecasts and safer trajectories for the SDV. Our prediction model learns to generate multimodal trajectory samples from a joint distribution over actor trajectories. Unlike previous diverse forecasting approaches, we directly optimize for predicting rare behavior that could impact the SDV, and estimate the probability distribution over these samples for more accurate risk assessment. Our contingency planner improves the decision making over these diverse samples. All in all, LOOKOUT achieves safer and less conservative driving than previous state-of-the-art models.

References

- [1] Zlatan Ajanovic, Bakir Lucevic, Barys Shyrokau, Michael Stoltz, and Martin Horn. Search-based optimal motion planning for automated driving. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4523–4530. IEEE, 2018. [1](#), [2](#), [5](#)
- [2] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE CVPR*, 2016. [1](#)
- [3] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016. [2](#)
- [4] S. Casas, C. Gulino, R. Liao, and R. Urtasun. Spagnn: Spatially-aware graph neural networks for relational behavior forecasting from sensor data. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020. [1](#), [2](#), [3](#), [7](#)
- [5] Sergio Casas, Cole Gulino, Simon Suo, Katie Luo, Renjie Liao, and Raquel Urtasun. Implicit latent variable model for scene-consistent motion forecasting. *ECCV 2020*, 2020. [1](#), [2](#), [3](#), [4](#), [7](#)
- [6] Sergio Casas, Wenjie Luo, and Raquel Urtasun. Intentnet: Learning to predict intention from raw sensor data. In *Conference on Robot Learning*, 2018. [2](#), [3](#)
- [7] Sergio Casas, Abbas Sadat, and Raquel Urtasun. Mp3: A unified model to map, perceive, predict and plan. In *CVPR*, 2021. [2](#)
- [8] Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. *arXiv preprint arXiv:1910.05449*, 2019. [1](#), [2](#), [7](#)
- [9] Felipe Codevilla, Matthias Müller, Antonio López, Vladlen Koltun, and Alexey Dosovitskiy. End-to-end driving via conditional imitation learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4693–4700. IEEE, 2018. [2](#)
- [10] Henggang Cui, Vladan Radosavljevic, Fang-Chieh Chou, Tsung-Han Lin, Thi Nguyen, Tzu-Kuo Huang, Jeff Schneider, and Nemanja Djuric. Multimodal trajectory predictions for autonomous driving using deep convolutional networks. *arXiv preprint arXiv:1809.10732*, 2018. [1](#)
- [11] Nemanja Djuric, Vladan Radosavljevic, Henggang Cui, Thi Nguyen, Fang-Chieh Chou, Tsung-Han Lin, and Jeff Schneider. Motion prediction of traffic actors for autonomous driving using deep convolutional networks. *arXiv preprint arXiv:1808.05819*, 2018. [1](#), [2](#)
- [12] Martin Engelcke, Dushyant Rao, Dominic Zeng Wang, Chi Hay Tong, and Ingmar Posner. Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks. In *2017 ICRA*, 2017. [2](#)
- [13] Haoyang Fan, Zhongpu Xia, Changchun Liu, Yaqin Chen, and Qi Kong. An auto-tuning framework for autonomous vehicles. *arXiv preprint arXiv:1808.04913*, 2018. [1](#)
- [14] Tianyu Gu and John M Dolan. On-road motion planning for autonomous vehicles. In *International Conference on Intelligent Robotics and Applications*, pages 588–597. Springer, 2012. [5](#)
- [15] Jason Hardy and Mark Campbell. Contingency planning over probabilistic obstacle predictions for autonomous road vehicles. *IEEE Transactions on Robotics*, 29(4):913–929, 2013. [3](#)
- [16] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. [4](#)
- [17] Joey Hong, Benjamin Sapp, and James Philbin. Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. [2](#)
- [18] Xin Huang, Stephen G. McGill, Jonathan A. DeCastro, Luke Fletcher, John J. Leonard, Brian C. Williams, and Guy Rosman. Diversitygan: Diversity-aware vehicle motion prediction via latent semantic sampling, 2020. [2](#)
- [19] Ajay Jain, Sergio Casas, Renjie Liao, Yuwen Xiong, Song Feng, Sean Segal, and Raquel Urtasun. Discrete residual flow for probabilistic pedestrian behavior prediction. *arXiv preprint arXiv:1910.08041*, 2019. [2](#)
- [20] Alex Kendall, Jeffrey Hawke, David Janz, Przemyslaw Mazur, Daniele Reda, John-Mark Allen, Vinh-Dieu Lam, Alex Bewley, and Amar Shah. Learning to drive in a day. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8248–8254. IEEE, 2019. [2](#)
- [21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2013. [4](#)
- [22] Bo Li. 3d fully convolutional network for vehicle detection in point cloud. In *2017 IEEE/RSJ IROS*, 2017. [2](#)
- [23] Wenjie Luo, Bin Yang, and Raquel Urtasun. Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In *Proceedings of the IEEE CVPR*, 2018. [2](#), [3](#)
- [24] Jianqi Ma, Weiyuan Shao, Hao Ye, Li Wang, Hong Wang, Yingbin Zheng, and Xiangyang Xue. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Transactions on Multimedia*, 2018. [3](#)
- [25] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11167–11176, 2020. [6](#)
- [26] Matthias Müller, Alexey Dosovitskiy, Bernard Ghanem, and Vladlen Koltun. Driving policy transfer via modularity and abstraction. *arXiv preprint arXiv:1804.09364*, 2018. [2](#)
- [27] Brian Paden, Michal Čáp, Sze Zheng Yong, Dmitry Yershov, and Emilio Frazzoli. A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on intelligent vehicles*, 1(1):33–55, 2016. [2](#), [5](#)
- [28] Seong Hyeon Park, Gyubok Lee, Manoj Bhat, Jimin Seo, Minseok Kang, Jonathan Francis, Ashwin R Jadhav, Paul Pu

- Liang, and Louis-Philippe Morency. Diverse and admissible trajectory forecasting through multimodal context understanding. *arXiv preprint arXiv:2003.03212*, 2020. 2
- [29] Tung Phan-Minh, Elena Corina Grigore, Freddy A Boulton, Oscar Beijbom, and Eric M Wolff. Covernet: Multimodal behavior prediction using trajectory sets. *arXiv preprint arXiv:1911.10298*, 2019. 1, 2
- [30] Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA ARTIFICIAL INTELLIGENCE AND PSYCHOLOGY ..., 1989. 2
- [31] Nicholas Rhinehart, Jeff He, Charles Packer, Matthew A. Wright, Rowan McAllister, Joseph E. Gonzalez, and Sergey Levine. Contingencies from observations: Tractable contingency planning with learned behavior models. *CoRR*, abs/2104.10558, 2021. 3
- [32] 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. 2
- [33] Nicholas Rhinehart, Rowan McAllister, Kris Kitani, and Sergey Levine. PRECOG: PREdiction Conditioned On Goals in Visual Multi-Agent Settings. *arXiv e-prints*, page arXiv:1905.01296, May 2019. 1, 2, 7
- [34] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635, 2011. 7
- [35] Abbas Sadat, Sergio Casas, Mengye Ren, Xinyu Wu, Pranab Dhawan, and Raquel Urtasun. Perceive, predict, and plan: Safe motion planning through interpretable semantic representations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2
- [36] Abbas Sadat, Mengye Ren, Andrei Pokrovsky, Yen-Chen Lin, Ersin Yumer, and Raquel Urtasun. Jointly learnable behavior and trajectory planning for self-driving vehicles. *IROS 2019*, 2019. 1, 2, 5, 6, 7
- [37] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*, pages 3483–3491, 2015. 2, 3, 4, 7
- [38] Charlie Tang and Russ R Salakhutdinov. Multiple futures prediction. In *Advances in Neural Information Processing Systems*, pages 15398–15408, 2019. 1, 2
- [39] Ömer Sahin Tas, Felix Hauser, and Christoph Stiller. Decision-time postponing motion planning for combinatorial uncertain maneuvering. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2419–2425. IEEE, 2018. 1, 3, 5
- [40] Ameni Trabelsi, J. Ross Beveridge, and Nathaniel Blanchard. Drowned out by the noise: Evidence for tracking-free motion prediction. *CoRR*, abs/2104.08368, 2021. 2
- [41] Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical Review E*, 62(2):1805–1824, Aug 2000. 6
- [42] Moritz Werling, Julius Ziegler, Sören Kammler, and Sebastian Thrun. Optimal trajectory generation for dynamic street scenarios in a frenet frame. In *2010 IEEE International Conference on Robotics and Automation*, pages 987–993. IEEE, 2010. 6
- [43] Bin Yang, Ming Liang, and Raquel Urtasun. Hdnet: Exploiting hd maps for 3d object detection. In *Conference on Robot Learning*, pages 146–155, 2018. 2
- [44] Bin Yang, Wenjie Luo, and Raquel Urtasun. Pixor: Real-time 3d object detection from point clouds. In *Proceedings of the IEEE CVPR*, 2018. 2, 3, 4
- [45] Sung Wook Yoon, Alan Fern, Robert Givan, and Subbarao Kambhamampati. Probabilistic planning via determinization in hindsight. In *AAAI*, pages 1010–1016, 2008. 3
- [46] Ye Yuan and Kris Kitani. Diverse trajectory forecasting with determinantal point processes. *arXiv preprint arXiv:1907.04967*, 2019. 1, 2, 7
- [47] Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. *arXiv preprint arXiv:2003.08386*, 2020. 1, 2, 4, 7
- [48] Wenyuan Zeng, Wenjie Luo, Simon Suo, Abbas Sadat, Bin Yang, Sergio Casas, and Raquel Urtasun. End-to-end interpretable neural motion planner. In *Proceedings of the IEEE CVPR*, 2019. 2
- [49] Wenyuan Zeng, Shenlong Wang, Renjie Liao, Yun Chen, Bin Yang, and Raquel Urtasun. Dsdnet: Deep structured self-driving network. In *ECCV*, 2020. 2
- [50] Wei Zhan, Changliu Liu, Ching-Yao Chan, and Masayoshi Tomizuka. A non-conservatively defensive strategy for urban autonomous driving. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 459–464. IEEE, 2016. 1, 3, 5
- [51] Hang Zhao, Jiyang Gao, Tian Lan, Chen Sun, Benjamin Sapp, Balakrishnan Varadarajan, Yue Shen, Yi Shen, Yuning Chai, Cordelia Schmid, et al. Tnt: Target-driven trajectory prediction. *arXiv preprint arXiv:2008.08294*, 2020. 2
- [52] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3D object detection. In *CVPR*, 2018. 2
- [53] Julius Ziegler, Philipp Bender, Thao Dang, and Christoph Stiller. Trajectory planning for bertha—a local, continuous method. In *2014 IEEE intelligent vehicles symposium proceedings*, pages 450–457. IEEE, 2014. 1, 2