This article has been accepted for publication in IEEE Transactions on Intelligent Transportation Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: 10.1109/TITS.2023.3337104

Goal-Oriented Pedestrian Motion Prediction

Jingyuan Wu, Johannes Ruenz, Hendrik Berkemeyer, Liza Dixon, and Matthias Althoff

***Abstract*—Forecasting the motion of others in shared spaces is a key for intelligent agents to operate safely and smoothly. We present an approach for probabilistic prediction of pedestrian motion incorporating various context cues. Our approach is based on goal-oriented prediction, yielding interpretable results for the predicted pedestrian intention, even without the prior knowledge of goal positions. By using Markov chains, the resulting probability distribution is deterministic—a beneficial property for motion planning or risk assessment in automated and assisted driving. Our approach outperforms a physics-based approach and improves over state-of-the-art approaches by reducing standard deviations of prediction errors and improving robustness against realistic, noisy measurements.**

***Index Terms*—Markov processes, motion planning, probabilistic model, road safety, pedestrian motion prediction.**

I. Introduction

PREDICTING the motion of traffic participants is a core issue for intelligent transportation systems designed to enhance road safety. Safety cannot be assured unless the safety of all road users, including vulnerable road users, such as pedestrians, is considered. In addition, intelligent vehicles should operate with a comfortable, natural, and smooth driving style. To this end, a long-term prediction horizon is required to enable driving with foresight. However, predicting the motion of pedestrians is challenging, as they have less inertia than larger agents, such as road vehicles. Moreover, their behavior can be influenced by various factors, such as their own intended destination, the road layout (including static obstacles), and their interaction with other road users. All these factors can result in a high degree of motion uncertainty. Some components of pedestrian movement behavior, such as group behavior and interactions among pedestrians in close proximity to each other [1]–[6], are not considered. They have arguably secondary effects from the perspective of autonomous vehicles and are used, e.g., for predictions of crowd dynamics in evacuation situations [7], which are not a focus of this work.

One possibility for computing the multimodal distribution of future trajectories is first identifying goals (a set of target positions on the map) and then predicting the motion conditioned on each goal. Many previous works [8]–[12] assume every position cell in a grid world to be a potential goal, whereas a small set of goals is used in [13]–[15]. To generate representative goals in unknown environments, one can use rules, such as allocating goals on sidewalks [13], or heuristic approaches [14]. However, those procedures of generating goals may lead to ambiguity, e.g., "Why is this location regarded as a target position instead of another location nearby?" and "Should the amount of generated goals on each topological space, such as sidewalks, be kept the same?" (we address this in Sec. III-C2).

The probability distribution over goals to an agent given its past trajectory can be estimated directly through the forward pass in artificial neural networks [10]–[12] or using Bayes' rule with dynamic models [8], [9], [13]–[15]. The inference of a large number of goals can still be efficient by means of the utility values of the states in a Markov decision process as in [8], [9], where the time complexity of computing those utility values is made independent of the number of goals by using deterministic state transition dynamics. However, this time complexity significantly increases when time is included in the state space [16, Chap. 4.7]. This makes it difficult to realize real-time applications that consider the cues from dynamic scenes both when predicting motion conditioned on goals and inferencing goals through utility values. Alternatively, one can augment a time-invariant planning-based predictor with spatiotemporal variations [13]–[15].

A. Related Work

Pedestrian motion prediction is still an active research area and its application domains include, e.g., automated driving, service robots, and surveillance [17], [18]. Following [7], [17], the prediction approaches can be categorized into physics-based, pattern-based, planning-based, and hybrid approaches.

a) Physics-based approaches: Approaches under this category use explicitly defined dynamic equations [19], [20]. Many microscopic models [2]–[4] are based on the prominent social force model introduced by Helbing and Molnar [1]. They originally aim at simulating realistic trajectories, e.g., for crowd simulations and evacuation dynamics in different types of infrastructures [7]. Recently, microscopic models have been applied to urban scenarios like signalized intersections [21], and for the prediction of jaywalking behavior [22]. Furthermore, Kooij *et al.* [23] incorporated various context cues in a dynamic Bayesian network to govern the switching between motion types. Batkovic *et al.* [24] used a closed-loop regulator to predict pedestrian trajectories and state covariances along a graph of connected edges representing pedestrianized areas. Koschi *et al.* [25] present a set-based prediction approach which includes all possible future states of pedestrians while considering formalized traffic rules. In general, physics-based approaches possess good explainability. However, their model

This work was supported by the project "interACT" within the European Union's Horizon 2020 programme under Grant 723395.

Jingyuan Wu, Johannes Ruenz, Hendrik Berkemeyer, and Liza Dixon are with Robert Bosch GmbH, Germany (e-mail: jingyuan.wu@de.bosch.com; johannes.ruenz@de.bosch.com; hendrik.berkemeyer@de.bosch.com; liza.dixon@de.bosch.com).

Matthias Althoff is with the Department of Computer Engineering, Technical University of Munich, Germany (e-mail: althoff@in.tum.de).

capacity is often limited and they can not benefit from large amounts of data [7].

b) Pattern-based approaches: Learning typical behavior patterns in known environments [26], [27], such as recognizing pedestrian crossing intention [28], [29], showed promising results. With the rapid advancement in deep learning, diverse approaches are proposed to learn social interactions between pedestrians [5], [6], [30], [31] and more complex spatiotemporal behavior patterns [32]–[34]. In contrast to predicting pedestrian trajectories from the first-person perspective [35], [36], many approaches use the rasterized bird-eye images of the agent's environment and encode the road context with convolutional neural networks [37]–[39]. Recent development adopts a vectorized representation of high-definition maps and agent trajectories, combined with graph neural networks, where the computational cost for context encoding can be reduced by a large margin compared to using convolutional neural networks [40], [41]. Often, recurrent neural networks, such as long short-term memory networks [42], are adopted to encode sequential observations and decode future trajectories [43]–[45]. Yao *et al.* [46] developed a bidirectional decoder to reduce accumulated long-term prediction errors by incorporating the endpoint of trajectories. More recent work [47]–[49] uses transformer networks [50] through a self-attention mechanism to better explore spatiotemporal dependency of features. Li *et al.* [51] used multi-scale graph-based spatial transformers taking the scene segmentation map and observed trajectories as inputs, with a memory graph to improve the smoothness of predictions. Bae and Jeon [52] constructed weighted multi-relational pedestrian graphs to account for social interactions.

Capturing the multimodal nature of motion prediction in the decoder part of neural networks can be achieved through, e.g., (i) learning a mixture of Gaussians [52], [53], (ii) sampling latent variables based on conditional variational autoencoders [12], [43], [45] or generative adversarial networks [6], [34] with multiple generators [54], (iii) employing a maneuver-based multi-task learning framework [55], (iv) adapting the particle filtering on top of recurrent models [56], and (v) using non-parametric approaches [37], [38]. Among them, deep generative models often suffer from the mode collapse [57] and the use of implicit latent variables prevents the interpretation of intermediate prediction results. To eliminate the mode collapse problem and make multimodal trajectory prediction more robust and explainable [58], many approaches condition the prediction on goals (or anchors, endpoints, etc., having similar concepts) [41], [48], [59], [60]. Still, sampling at runtime leads to non-deterministic prediction results, which negatively affects the evaluation of collision risk in assisted and automated driving. In [49], Gu *et al.* present a new framework for sampling pedestrian trajectories by progressive denoising from a noise distribution, where the reverse diffusion process [61], [62] is learned via a parameterized Markov chain. However, the computational effort of this approach is expensive due to the required multiple steps in the reverse diffusion process.

In general, pattern-based approaches with high performance can predict more accurate trajectories than physics-based approaches when a sufficient amount of suitable training data are available, but at the cost of low interpretability [7], [17]. The lack of interpretability makes it difficult to validate these approaches in safety-critical applications [63]. Moreover, their ability to generate physically consistent results as well as their generalizability and robustness against unseen situations and noisy inputs are questionable [7], [17].

c) Planning-based approaches: According to [16, Chap. 8.1], the term *planning* can refer to "any computational process that takes a model as input and produces or improves a policy for interacting with the modeled environment". Planning-based approaches emphasize the reasoning about the intention of rational agents and possible paths to goals [17]. In contrast to those forward planning approaches by using a predefined reward function [13], [14], inverse planning approaches aim to recover the reward function from demonstrated behavior by using feature matching techniques. Early works [8], [9] focus on the interaction between agents and static environments and learn the weights of designed local scene features. Furthermore, Kretzschmar *et al.* [64] jointly predicted the trajectories of interacting agents by introducing discrete navigation decision features to represent, e.g., cultural preferences of passing on a specific side. Instead of handcrafting features, Wulfmeier *et al.* [65] exploited deep neural networks as the reward function approximator to learn complex features. Moreover, it is not uncommon for planning-based methods to be combined with other methods. For instance, Ma *et al.* [66] forecasted multiple pedestrian trajectories by utilizing motion planning and fictitious play [67] from game theory. In [10], the destination prediction and the planning-based motion prediction are cascaded within a monolithic neural network. Deo and Trivedi [11] utilized inverse reinforcement learning to predict the policy conditioned on the jointly inferred goals and applied an attention-based trajectory generator. More recently, the vehicle-pedestrian interaction is modeled as a multi-agent deep reinforcement learning problem in [68], where the pedestrian agent can learn an intelligent crossing behavior and a low collision rate is achieved under a high measurement noise level in a simulation. Essentially, planning-based approaches perform well in structured environments given well-defined goals, and they tend to yield more accurate long-term predictions compared to physics-based approaches and generalize better in unknown environments compared to pattern-based approaches [17]. However, the computational effort when using classical methods (such as value iterations) increases exponentially with the number of continuous state variables [17, Sec. 8.2], [16, Chap. 4.7].

d) Hybrid approaches: Hybrid approaches combining physics-based approaches and deep learning algorithms have gained momentum in recent years [7]. Hossain *et al.* [69] modeled each module of an extended social force model by neural networks for predicting a single trajectory. Yue *et al.* [70] proposed a neural differentiable equation model consisting of goal attraction, inter-agent repulsion, and environment repulsion parts. The uncertainty in the motion dynamics and observations is considered via a variational autoencoder and the goal positions are sampled from a goal sampling network. Our work is related to this category with its core component

using planning.

B. Contributions

This work significantly differs from our previous work in [14], [71]. First, we improved the interaction between pedestrians and their environment by replacing heuristics with policies systematically obtained from machine learning. In particular, we model pedestrian interaction with the road layout using inverse reinforcement learning. Furthermore, we can now directly utilize empirical data regarding the temporal aspect of pedestrian behavior through a novel input transition model, which also further modularizes our approach. Second, we developed a novel generation of goal regions to completely eliminate the ambiguity of the goal generation in [14]. Third, the collision probability computation can now consider multiple other traffic participants appropriately; before, only the most important other traffic participant was considered. Fourth, we evaluated our approach for the first time on large-scale datasets and compared it with state-of-the-art methods. Fifth, the effectiveness of each module of our approach is examined in our ablation study. In summary, the core contributions of this paper are as follows:

- We incorporate various techniques in Markov chains (as the basis for our approach) for multimodal motion prediction. This integration of various context cues is novel, and it shows promising results compared to state-of-the-art approaches while offering superior interpretability to many end-to-end approaches. Our approach does not rely on sampling, and the prediction results are deterministic.
- An approach for constructing goal regions is proposed to address the challenge of generating goals in unknown environments. The impact of the number of goal regions on prediction performance is investigated. By means of these goal regions, our approach does not suffer from mode collapse, which can affect deep generative models.
- Unlike end-to-end approaches, we demonstrate that each module of our approach can be efficiently trained in a separate manner, which allows us to first validate each module before adding others and helps system inspection.
- We investigate the prediction performance under realistic, noisy position measurements. Our work shows that our approach is more robust than baseline physics-based and pattern-based approaches.

C. Organization

The remainder of this paper is structured as follows. Sec. II presents a holistic view of our approach and explains the motivation behind our framework. In Sec. III, we introduce our goal-oriented pedestrian motion model. Additionally, Sec. IV models the interaction between pedestrians and road vehicles. Our evaluation on large-scale datasets is found in Sec. V. Finally, Sec. VI concludes the paper.

II. Overview

We use controlled Markov chains (as a special case of Markov decision processes) for probabilistic motion prediction. Although there are systematic errors from the required discretization of the state and input space, no probabilistic errors are introduced because no random sampling is applied. Hence, the resulting probability distribution is deterministic [72]—a beneficial property for motion planning or risk assessment in automated and assisted driving. Our approach involves both physics-based and learning-based models. Since the human body movement (change of position) based on decision making (move in a certain direction with a certain velocity) can be well modeled, we propagate the position following a kinematic model. In contrast, the underlying cognitive processes of pedestrians are complex. Hence, we learn the input transition probabilities from data.

We consider for pedestrian decision making the typical context cues in urban traffic scenarios—the road layout (static environments) and road vehicles (dynamic obstacles), as shown in the left part of Fig. 1. Unlike end-to-end pipelines, we separately encode the road layout and road vehicles into our controlled Markov chains for the following reasons:

- This avoids solving planning problems with dynamic obstacles (which are in general computationally expensive) and makes real-time applications more tractable. Yet, we augment our time-invariant planning-based predictor with spatiotemporal variations (cf. Sec. III-D2).
- The separate modeling of interaction with road vehicles is conditioned on their future trajectories (cf. Sec. IV-A), which facilitates trajectory planning for intelligent vehicles, such as, “How would the pedestrian behave if the system maintains the intended speed and stays on the desired path?”
- We are able to first train the pedestrian interaction model with the road layout (cf. Sec. III-D2) and then optimize the interaction model with road vehicles (cf. Sec. IV-C). This method not only copes with the intrinsic bias of most datasets towards uncritical situations by emphasizing relevant interactions, but also facilitates divide-and-conquer strategies for validation.

Specifically, we apply goal-oriented prediction and learn the reward function (cf. Sec. III-C3) due to the following critical advantages compared to pattern-based methods:

- Planning-based methods tend to generalize better in unknown environments, as the reward function encapsulates human activities in terms of local scene features instead of physical locations [9].
- Instead of learning the policy for a specific task, the reward-function-based approaches can be easily applied to different tasks (goals) by deriving the corresponding policy [65].
- The reasoning about goals provides a better explainability of the agent’s intent, such as towards a target region across the road (cf. Sec. III-C2 and Sec. III-E).

III. Modeling of Pedestrian Motion

We model pedestrian motion in urban traffic environments as Markov decision processes. For the mentioned reasons in Sec. II, we divide our prediction model into submodels: the state transitions are abstracted from a kinematic model and the learned input transitions represent the underlying cognitive

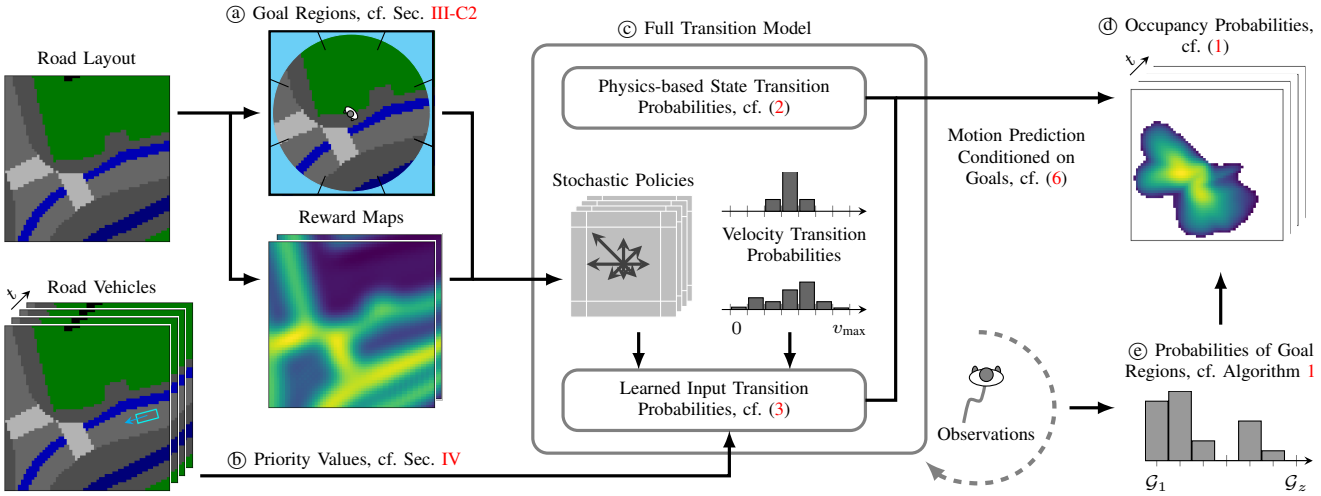


Fig. 1. Flowchart of our motion prediction.

processes of pedestrians. First, we introduce the notation and formulate the prediction objective (Fig. 1ⓓ) by means of the intended destination of pedestrians as latent variables. Then, we formulate our Markov decision processes, where we discuss the difficulty of assigning target positions on the map and propose a method for automatically generating goal regions (Fig. 1ⓐ). Later, we present our full transition model (Fig. 1ⓒ) considering additional context cues (Fig. 1ⓑ). Last, we present an algorithm to infer the intended destination of pedestrians as intermediate prediction results based on online observations (Fig. 1ⓔ).

A. Notation

We use grid cells to discretize the state space $\mathcal{S} \subset \mathbb{R}^2$ and input space $\mathcal{U} \subset \mathbb{R}^2$, cf. Fig. 2. The state $s = [s_x, s_y]^T$ is defined as the position in x and y direction. The input $u = [\psi, v]^T$ consists of the orientation $\psi \in \mathbb{R}$ and velocity $v \in \mathbb{R}_{\geq 0}$. We denote the state cells by $\mathcal{S}_i$ with Latin indices and input cells by $\mathcal{U}_\alpha$ with Greek indices. In addition, the indices $\overline{\alpha}$ and $\underline{\alpha}$ are used to distinguish between an orientation interval $\mathcal{A}_{\overline{\alpha}} \subset \mathbb{R}$ and a velocity interval $\mathcal{V}_{\underline{\alpha}} \subset \mathbb{R}_{\geq 0}$. Each combination of $\overline{\alpha}$ and $\underline{\alpha}$ determines a unique index of input cell α, and vice versa.

In addition, the latent variables $g \in \mathbb{R}^2$ are introduced to represent the goal position which the pedestrian aims to reach. Each goal region is denoted by $\mathcal{G}_z \subset \mathbb{R}^2$ and their union is denoted by $\mathcal{G}$.

For notational simplicity, we use a shorthand notation for events, such as $s_i \coloneqq (s \in \mathcal{S}_i)$, $u_\alpha \coloneqq (u \in \mathcal{U}_\alpha)$, $\psi_{\overline{\alpha}} \coloneqq (\psi \in \mathcal{A}_{\overline{\alpha}})$, $v_{\underline{\alpha}} \coloneqq (v \in \mathcal{V}_{\underline{\alpha}})$, and $g_z \coloneqq (g \in \mathcal{G}_z)$.

B. Prediction Objective

We aim to predict pedestrian motion based on the observations from time t_m to t_0, where $t_m < t_0$. The observed position, orientation, and velocity of a pedestrian is denoted by $y \coloneqq [s_x, s_y, \psi, v]^T$. We abbreviate the according event by $y_i^\alpha \coloneqq (y \in \mathcal{S}_i \times \mathcal{U}_\alpha)$. Let the sequence of observed events be $Y(t_{m:0}) \coloneqq (y_i^\alpha(t_m), \dots, y_{i'}^{\alpha'}(t_0))$. Given observations $Y(t_{m:0})$, one can predict the occupancy probabilities $P(s_i(t_k)|Y(t_{m:0}))$ at time $t_k > t_0$ by marginalizing out inputs and goals [73, (1.10)]:

$$P(s_i(t_k)|Y(t_{m:0})) = \sum_{z=1}^{|\mathcal{G}|} P(g_z|Y(t_{m:0})) \sum_{\alpha=1}^{|\mathcal{U}|} P(s_i(t_k), u_\alpha(t_k)|Y(t_{m:0}), g_z). \quad (1)$$

The first term on the right side of the equation above refers to the posterior probabilities of goals. We do not change the goal probabilities over prediction steps, which is a common assumption as used, e.g., in [74]. The second term refers to the predictions conditioned on goals. We detail their computation in subsequent subsections.

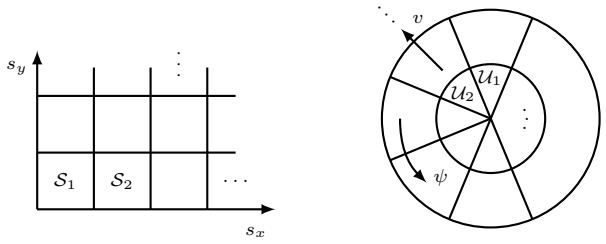


Fig. 2. State (left) and input cells (right).

C. Markov Decision Processes

The decision making in humans can be understood as a learning process that "is supervised through the use of reward signals in response to the observed outcomes of actions" [75]. Hence, we formulate that decision making as Markov decision processes (MDPs) aiming to find the policy mapping from the state space (where the agent arrives) to the action space (in which direction the agent moves). In order to reduce the computational effort for solving MDPs, we not only derive stationary policies by considering static environments, but also exclude the velocity from the action space of MDPs. In

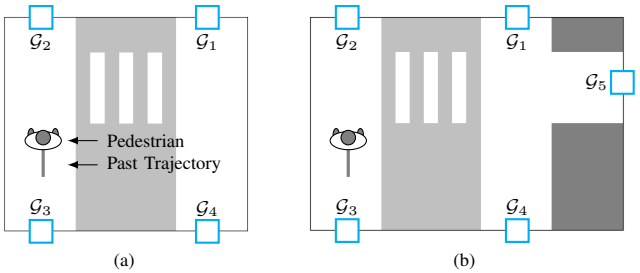


Fig. 3. A pedestrian in two different maps with different sizes: *Where should the possible goal regions $\mathcal{G}_i$ be placed?*

other words, we consider only the spatial aspect of possible trajectories in this subproblem.

1) Formulation of Markov decision processes: The used MDP in this work is described by the tuple $(\mathcal{S}, \mathcal{A}, P(s_i|s_j, \psi_{\overline{\alpha}}), R(s_i, \psi_{\overline{\alpha}}))$, where the state space is $\mathcal{S}$, the action space is $\mathcal{A}$, the state transition dynamics $P(s_i|s_j, \psi_{\overline{\alpha}})$ specifies the probability that an agent arrives in the cell $\mathcal{S}_i$ after moving one *step* in direction $\psi \in \mathcal{A}_{\overline{\alpha}}$ from an adjacent cell $\mathcal{S}_j$, and the reward function $R(s_i, \psi_{\overline{\alpha}}) \rightarrow \mathbb{R}_{\leq 0}$ maps the pair of events to a real value.

2) Goal regions: We solve multiple MDPs, each with a different set of terminal states. The set of terminal states is used in the MDP not only because it makes the total amount of expected reward finite, but also because it can be interpreted as the intended destination of pedestrians, i.e., a goal region $\mathcal{G}_z$.

In some environments, meaningful locations can be regarded as goals, such as the entrances and exits of buildings. In urban environments, however, it might be challenging to predefine goals (i.e., their location and prior probability). Previous works [13], [14] generate goals, e.g., on the sidewalk or on the boundary of a considered region. A drawback of these attempts is the difficulty of assigning the prior probability of generated goals appropriately. An example is illustrated in Fig. 3: the possibilities of the goal regions $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_5$ are assumed to have similar values to a pedestrian based on their past trajectory. When the prior probability distribution over goals is uniform, the estimated crossing probability (the sum of probabilities of those goals across the street) to the pedestrian in Fig. 3(a) can differ much from that in Fig. 3(b). Then, a question may arise: "Should the amount of those generated goals in each sidewalk area be considered in their prior probability distribution?" To put it simply, the amount of generated goals as well as their location and prior probability distribution have a strong influence on the prediction results not only in terms of the final occupancy probabilities, but also in terms of intermediate results, such as how likely it is that the pedestrian will cross the road.

In order to avoid the above-mentioned problems, for each pedestrian we first consider an individual square region centered around their current position, cf. Fig. 4(e). Under the assumption that their real destination is outside the circle that fits in this individual region, we refer to the outside part as the union of goal regions $\mathcal{G}$. Further, we split $\mathcal{G}$ into several regions $\mathcal{G}_i$ by beams from the center with equalized central

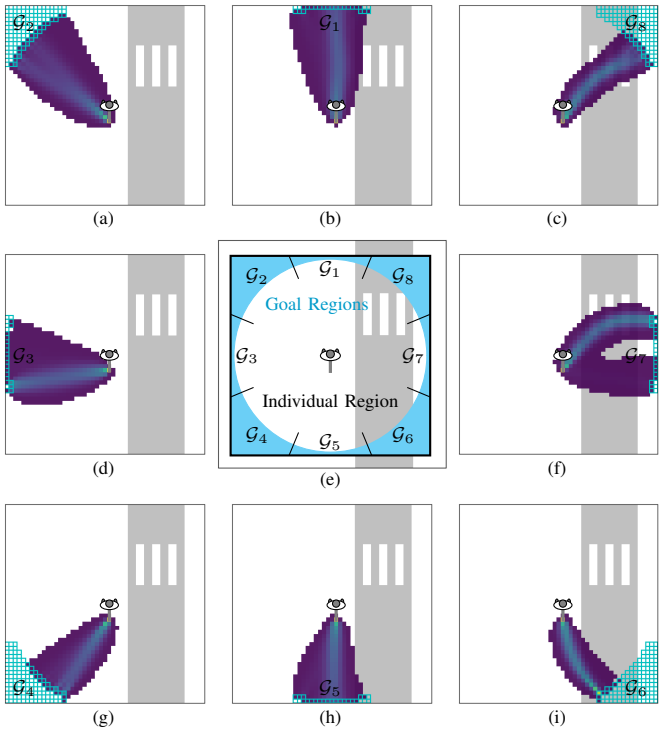


Fig. 4. Split the goal regions (e) for a pedestrian with $|\mathcal{G}| = 8$ as an example. Expected state visitation frequencies from the initial state towards each goal region under stochastic policies (a-d, f-i), where the terminal state cells of each MDP are shown in unfilled cyan squares.

angles. By doing this, we construct $|\mathcal{G}|$ possible goal regions.

3) Learning the reward, maximum entropy policies: Since there is a sense that the agent is "programmable" through the provided reward function [75], we learn the reward function $R(s_i, \psi_{\overline{\alpha}})$ modeled as a convolutional neural network [76], [77] from demonstrated trajectories. To this end, we adopt the popular maximum entropy inverse reinforcement learning approach in [78] aiming to find stochastic policies for prediction, which are the least biased estimation on the given information while being as uncertain as possible [79].

The scheme of maximum entropy inverse reinforcement learning is illustrated in Fig. 5. Here, since we focus on the spatial aspect of trajectories, each demonstrated trajectory is converted to a sequence of event pairs $\zeta := ((s_{i'}, \psi_{\overline{\alpha}'}), \ldots, (s_{i''}, \psi_{\overline{\alpha}''}))$. Our convolutional neural network (cf. Appendix A for details) takes as input the (static) map information, which can be presented in channels representing surface types in case of semantic maps (as shown in Fig. 5) or color components in case of RGB satellite images. As output, different kinds of reward maps $R(s_i, \cdot)$ are generated (e.g., one for the orientations towards directly adjacent state cells, and the other one for the orientations towards diagonally adjacent state cells). For each demonstrated trajectory, we solve a MDP with the rewards produced by the neural network on the current training iteration, where the terminal state set is defined as the ending position cell of the demonstrated trajectory (as common in [9], [65]). After finding the stochastic policies $\pi(\psi_{\overline{\alpha}}|s_i)$ by solving the MDP, cf. [81, Theorem 6.8], one can compute the

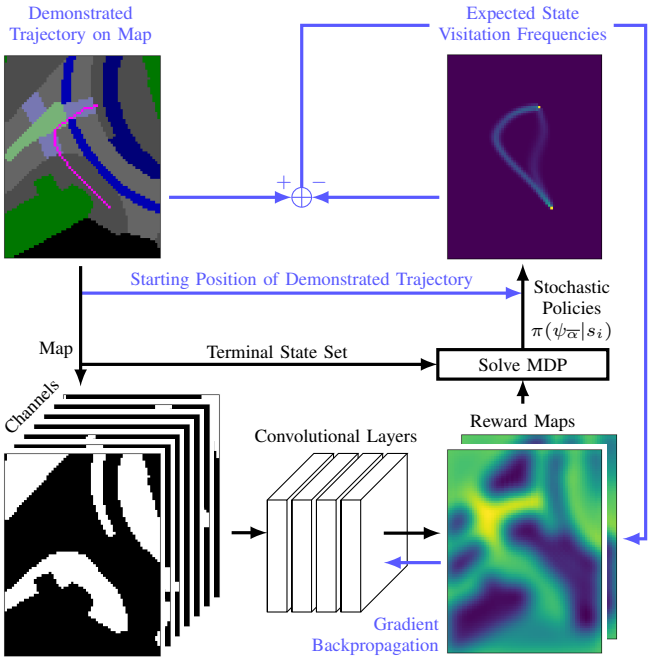


Fig. 5. Scheme of maximum entropy inverse reinforcement learning for training a convolutional neural network as the reward function (scheme inspired by [65], [80]). Reward maps and state visitation frequencies with lighter colors mean larger values. A demonstrated trajectory "—" is plotted on the semantic map with "■" sidewalk, "■" road, "■" zebra island, "■" traffic island, "■" restricted area, "■" building, "■" bicycle path, and "■" undefined area.

expected action visitation frequencies (the expected number of times that an action in a state is executed, cf. [81, Algorithm 9.3]) from the starting position cell of each demonstrated trajectory and following stochastic policies. Then, the expected action visitation frequencies are compared with the empirical ones from demonstrated trajectories. Their differences are used to compute the gradient for updating weights, cf. [65, (6)].

During model inference (cf. Fig. 5, the steps marked black), each goal region $\mathcal{G}_z$ is related to a MDP and regarded as the terminal state set, cf. Fig. 4(a-d, f-i). The obtained stochastic policies are embedded in our full transition model for motion prediction, which is detailed in the next subsection.

D. Decomposition of Transition Model

In this subsection, we present our controlled Markov chains based on [82] as a special case of Markov decision processes. We decompose our full transition model into a physics-based state transition model and a learned input transition model. Specifically, we augment the stochastic policies (obtained by solving MDPs, cf. Sec. III-C) with spatiotemporal variations. This full transition model is then used to recursively compute the motion conditioned on goals in (1).

1) Abstraction of kinematic model: We abstract the kinematic model $\dot{s}_x = v\cos\psi$, $\dot{s}_y = v\sin\psi$ to a Markov chain. Given an input event $u_\alpha(t_k)$ at time t_k, the state transition probabilities

$$\Phi_{ij}^{\alpha} \coloneqq P(s_i(t_{k+1})|s_j(t_k), u_\alpha(t_k)) \tag{2}$$

for a time step increment can be computed offline by Monte Carlo simulations [82, (4.16)]. We store the values Φ_{ij}^{α} in the state transition matrices $\Phi^{\alpha} \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$, where each matrix Φ^{α} is subject to a certain input cell $\mathcal{U}_\alpha$ and Φ_{ij}^{α} denotes the element in the ith row and jth column of Φ^{α}.

2) Input transition probability: We learn the transitions for inputs from data. Let $\Gamma_{i,z}(t_k) \in \mathbb{R}^{|\mathcal{U}|\times|\mathcal{U}|}$ be the time-varying input transition matrix subject to state cell $\mathcal{S}_i$ and goal region $\mathcal{G}_z$. The element in the αth row and βth column of $\Gamma_{i,z}(t_k)$ represents the input transition probability

$$\Gamma_{i,z}^{\alpha\beta}(t_k) \coloneqq P\left(u_\alpha(t_k)'|u_\beta(t_k), s_i(t_k), g_z\right), \tag{3}$$

where the instantly changed input probability distribution at points in time is indicated by a prime [82, (4.22)]. We further decompose each matrix $\Gamma_{i,z}(t_k)$ into a time-invariant goal-dependent matrix $\Psi_{i,z} \in \mathbb{R}^{|\mathcal{U}|\times|\mathcal{U}|}$ and a time-varying goal-independent priority vector $\lambda_i(t_k) \in \mathbb{R}^{|\mathcal{U}|}$ (both are conditioned on state cell $\mathcal{S}_i$). Accordingly, $\Psi_{i,z}^{\alpha\beta}$ and $\lambda_i^{\alpha}(t_k)$ represent elements in $\Psi_{i,z}$ and $\lambda_i(t_k)$, respectively. While the transition probabilities $\Psi_{i,z}^{\alpha\beta}$ describe the evolution of the pedestrian's orientation and velocity during interaction with static environments, the priority values $\lambda_i^{\alpha}(t_k) \geq 0$ capture the cues from dynamic environments. In this way, the input transition probabilities $\Gamma_{i,z}^{\alpha\beta}(t_k)$ are computed as [82, (5.4)]

$$\begin{aligned}\Gamma_{i,z}^{\alpha\beta}(t_k) &= \mathtt{norm}\left(\hat{\Gamma}_{i,z}^{\alpha\beta}(t_k)\right) \coloneqq \frac{\hat{\Gamma}_{i,z}^{\alpha\beta}(t_k)}{\sum_\alpha \hat{\Gamma}_{i,z}^{\alpha\beta}(t_k)},\\ \hat{\Gamma}_{i,z}^{\alpha\beta}(t_k) &= \lambda_i^{\alpha}(t_k)\Psi_{i,z}^{\alpha\beta}\end{aligned} \tag{4}$$

with the normalization operator $\mathtt{norm}()$. The motivation of this decomposition is as follows: First, we assume that pedestrians rarely change their destination intention regardless whether a vehicle is approaching or not. Second, pedestrians tend to avoid colliding with other moving traffic participants [18]. A small priority value can be assigned to the actions that lead to a dangerous situation, producing higher probabilities of other actions, such as slowing down. Hence, the multiplication of $\lambda_i^{\alpha}(t_k)$ and $\Psi_{i,z}^{\alpha\beta}$ ensures a collision-aware prediction of pedestrian behavior while taking into account their route preference.

Under the assumption that the orientation and velocity are stochastically independent because pedestrian walking speeds are unlikely affected by their goal intention [13], the time-invariant input transition probabilities $\Psi_{i,z}^{\alpha\beta}$ subject to $\mathcal{G}_z$ in (4) can be computed as

$$\Psi_{i,z}^{\alpha\beta} \propto \underbrace{\pi_z\left(\psi_{\overline{\alpha}}\middle|s_i\right)}_{\text{stochastic policies}} \underbrace{P\left(v_{\underline{\alpha}}\middle|v_{\underline{\beta}}\right)}_{\text{statistics from data}}. \tag{5}$$

The stochastic policies $\pi_z(\psi_{\overline{\alpha}}|s_i) \coloneqq P(\psi_{\overline{\alpha}}|s_i, g_z)$ are the solution to a MDP with a goal region $\mathcal{G}_z$ as the terminal state set, cf. Sec. III-C3. The velocity transition probabilities $P(v_{\underline{\alpha}}|v_{\underline{\beta}})$ can be extracted from empirical data. This is achieved through counting the frequencies of the pedestrian's velocity at the successive points in time t_k and t_{k+1} falling into the velocity interval corresponding to the indices $\underline{\beta}$ and $\underline{\alpha}$, respectively. The computation of priority values $\lambda_i^{\alpha}(t_k)$ in (4) will be detailed in Sec. IV.

Algorithm 1: Estimate posterior probabilities of states and inputs, and that of goals (adapted from [13, Algorithm 1]).

Given observation $\exists j, \beta\colon y_j^\beta(t_m)$, initialize prior probabilities $\forall i, \alpha\colon P(s_i(t_m), u_\alpha(t_m)|y_j^\beta(t_m), g_z)$, $\forall z\colon P(g_z|y_j^\beta(t_m))$

for $k = m+1, \dots, 0$ **do**

  for $z = 1, \dots, |\mathcal{G}|$ **do**

    Predict states and inputs conditioned on goals, cf. (6):

$$\forall i, \alpha\colon P(s_i(t_k), u_\alpha(t_k)|Y(t_{m:k-1}), g_z) = \sum_{j=1}^{|\mathcal{S}|} \Phi_{ij}^\alpha \sum_{\beta=1}^{|\mathcal{U}|} \Gamma_{j,z}^{\alpha\beta}(t_{k-1}) P(s_j(t_{k-1}), u_\beta(t_{k-1})|Y(t_{m:k-1}), g_z) \tag{A1.1}$$

    Given observation $\exists j, \beta\colon y_j^\beta(t_k)$, update probabilities of states and inputs using Bayes' rule [73, (12.7)]:

$$\forall i, \alpha\colon P(s_i(t_k), u_\alpha(t_k)|Y(t_{m:k}), g_z) \propto \underbrace{P(y_j^\beta(t_k)|s_i(t_k), u_\alpha(t_k), g_z)}_{\text{observation likelihood}} P(s_i(t_k), u_\alpha(t_k)|Y(t_{m:k-1}), g_z) \tag{A1.2}$$

    Compute marginal likelihood of goals [73, (12.6)]:

$$P(y_j^\beta(t_k)|Y(t_{m:k-1}), g_z) = \sum_{i=1}^{|\mathcal{S}|} \sum_{\alpha=1}^{|\mathcal{U}|} \underbrace{P(y_j^\beta(t_k)|s_i(t_k), u_\alpha(t_k), g_z)}_{\text{observation likelihood}} P(s_i(t_k), u_\alpha(t_k)|Y(t_{m:k-1}), g_z) \tag{A1.3}$$

    Update probability distribution over goals [73, (12.4, 12.5)]:

$$P(g_z|Y(t_{m:k})) \propto P(y_j^\beta(t_k)|Y(t_{m:k-1}), g_z) P(g_z|Y(t_{m:k-1})) \tag{A1.4}$$

3) Propagation of states and inputs conditioned on goals: The state and input cells can be jointly propagated, where the input probability distribution changes instantly at points in time, while the state probability distribution is updated for a time step increment. By virtue of the Markov property, one can recursively compute the joint probabilities of states and inputs in (1) with the state and input transition probabilities in (2) and (3), respectively [82, (4.25)]:

$$P(s_i(t_{k+1}), u_\alpha(t_{k+1})|Y(t_{m:0}), g_z) = \sum_{j=1}^{|\mathcal{S}|} \Phi_{ij}^\alpha \sum_{\beta=1}^{|\mathcal{U}|} \Gamma_{j,z}^{\alpha\beta}(t_k) P(s_j(t_k), u_\beta(t_k)|Y(t_{m:0}), g_z). \tag{6}$$

E. Bayesian Filtering

We use a Bayesian filter to estimate the posterior probability distribution over goals $P(g_z|Y(t_{m:0}))$ in (1), i.e., how likely each of the goal regions $\{\mathcal{G}_1, \mathcal{G}_2, \dots\}$ is in the light of online observations, as described in Algorithm 1. The joint probabilities of states and inputs are initialized based on the earliest observation. Furthermore, we assume a uniform prior probability distribution over goals. At each time step, the states and inputs are jointly predicted conditioning on each goal, cf. (A1.1). By using Bayes' rule, one obtains the posterior probabilities of states and inputs $P(s_i(t_k), u_\alpha(t_k)|Y(t_{m:k}), g_z)$ from the product of the observation likelihood and the predicted states and inputs, cf. (A1.2). By marginalizing out the states and inputs from that product, one obtains the marginal likelihood of goals $P(y_j^\beta(t_k)|Y(t_{m:k-1}), g_z)$, cf. (A1.3). Finally, one uses the marginal likelihood of goals to recursively update their probability $P(g_z|Y(t_{m:k}))$, cf. (A1.4).

Note that the interaction with moving agents is considered in estimating the probability distribution of goal regions (during the observation period). Although we do not change the goal probability distribution for $t_k > t_0$, cf. (1), the interaction with moving agents further affects the prediction results conditioned on each goal region.

IV. Interaction with Road Vehicles

In addition, the presence of other traffic participants influences pedestrian behaviors. In this section, we present an algorithm which constantly adapts the predicted velocity and orientation of pedestrians by evaluating their collision probabilities with respect to road vehicles. The purpose of this section is to compute the time-varying priority values $\lambda_i^\alpha(t_k)$ in (4) for dynamic scenes.

A. Risk Measure Caused by Road Vehicles

Firstly, we predict the motion of road vehicles. Then, we weight the future driving corridor of vehicles. Based on that, we introduce a risk measure.

1) Vehicle motion prediction: Due to the relative difference in the inertia of vehicles compared to that of pedestrians, we can faithfully predict vehicles' future trajectories by using a physics-based model, e.g., the constant velocity model. Note that the planned trajectory of the ego vehicle can also be employed, which enables a direct assessment of planning with the conditioned pedestrian motion prediction results.

2) Weighted driving corridor: There is a certain distribution of the minimum time gap when pedestrians traverse the driving corridor of approaching vehicles [45], which is known as pedestrian gap acceptance [83]. Hence, we weight the future driving corridor of vehicles by the estimated time of arrival, cf. Fig. 6: the shorter the time gap, the more likely it will be

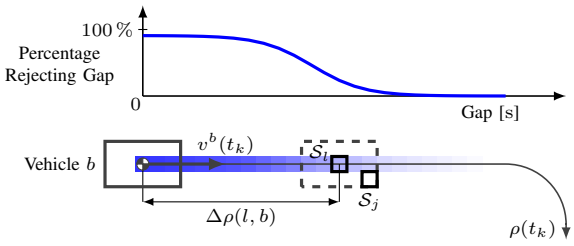


Fig. 6. Weights on the future driving corridor of a vehicle based on pedestrian gap acceptance [83]. A pedestrian with reference point in $\mathcal{S}_j$ intersects with a vehicle with reference point in $\mathcal{S}_l$ by considering its body geometry (dashed box).

rejected by pedestrians. Let $w_l^b(t_k) \in \mathbb{R}_{\geq 0}$ denote the weight of state cell $\mathcal{S}_l$ caused by a vehicle $b \in \mathcal{B}$ at time t_k, where $\mathcal{B}$ is the set of vehicles. We place a curvilinear coordinate system at the center of gravity of each vehicle b at time t_k. The axis ρ, in its simplest form, is along the vehicle moving direction at t_k (ρ can also be defined as the planned trajectory of the ego vehicle, or the predicted trajectory of other vehicles). We introduce the set $\mathcal{S}^b_{\text{corridor}}(t_k) \subset \mathbb{R}^2$ which contains the state cells lying on the axis $\rho(t_k)$ of vehicle b. The distance between a state cell $\mathcal{S}_l$ and a vehicle b is denoted by $\Delta\rho(l,b) \in \mathbb{R}_{\geq 0}$. The weights for state cells can be computed based on a logistic function in [83] with the parameters $\vartheta_1, \vartheta_2 \in \mathbb{R}_{\geq 0}$ as

$$w_l^b(t_k) = \begin{cases} \frac{1}{1+\exp\left(-\vartheta_1+\vartheta_2 \frac{\Delta\rho(l,b)}{v^b(t_k)}\right)}, & \text{if } \mathcal{S}_l \subset \mathcal{S}^b_{\text{corridor}}(t_k) \\ 0, & \text{otherwise,} \end{cases} \tag{7}$$

where $v^b(t_k)$ is the predicted velocity of vehicle b. We omit in the notation $w_l^b(t_k)$ the dependence on the parameters ϑ_1, ϑ_2 for notational simplicity.

3) Risk measure: We consider the weighted driving corridors of multiple vehicles and their body geometry. We first introduce an indicator function $\mathbf{1}^b(j,l)$ which returns 1 if a pedestrian with reference point in $\mathcal{S}_j$ intersects with a vehicle b with reference point in $\mathcal{S}_l$, and 0 otherwise (cf. Fig. 6). Then, the risk measure $p_j^{\text{risk}}(t_k)$ of being at a place $s(t_k) \in \mathcal{S}_j$ for the pedestrian can be computed as

$$p_j^{\text{risk}}(t_k) = \max_{b \in \mathcal{B}, \mathcal{S}_l \subset \mathcal{S}^b_{\text{corridor}}(t_k)} w_l^b(t_k) \mathbf{1}^b(j,l). \tag{8}$$

When multiple vehicles occur, the risk measure of being at each place is computed via the maximization over $b \in \mathcal{B}$, i.e., the vehicle with the least estimated time of arrival at that place (which causes the highest risk measure) is considered, such that $0 \leq p_j^{\text{risk}}(t_k) \leq 1$. The maximization over $\mathcal{S}_l \subset \mathcal{S}^b_{\text{corridor}}(t_k)$ implies that the driving corridor in terms of the reference point of a vehicle is extended by considering its body geometry via the indicator function $\mathbf{1}^b(j,l)$.

B. Time-Varying Priority Values

We relate the state-dependent and input-dependent priority value λ_i^α with the state-dependent risk measure p_j^{risk}. To this end, the priority value for each event pair $(s_i(t_k), u_\alpha(t_k))$ is evaluated by the risk measure under the effect of the event pair at the subsequent points in time $t_{k+1}, \ldots, t_{k+N_{\text{check}}}$ with $N_{\text{check}} \in \mathbb{N}$ as user-defined collision checking steps. With a larger value of N_{check}, the pedestrian is assumed to behave more conservatively, such as they would already adapt their behavior even if they are far away from road vehicles. The priority values at time t_k can be computed as [14, (20)]

$$\lambda_i^\alpha(t_k) = 1 - \max_{\kappa \in \{1,\ldots,N_{\text{check}}\}} \sum_{j=1}^{|\mathcal{S}|} p_j^{\text{risk}}(t_{k+\kappa}) P(s_j(t_{k+\kappa}) | s_i(t_k), u_\alpha(t_k)) \tag{9}$$

with $P(s_j(t_{k+\kappa})|s_i(t_k), u_\alpha(t_k))$ as the element in the jth row and ith column of the κth power of the state transition matrix, i.e., $(\Phi^\alpha)^\kappa \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$. Here, the use of the summation operator takes into account the state transition uncertainty from a state cell $\mathcal{S}_i$ by taking an input cell $\mathcal{U}_\alpha$ for κ time steps. The use of the max operator is equivalent to picking the maximal risk measure over a period [84, (19)]. For notational simplicity, we omit in the notation $\lambda_i^\alpha(t_k)$ the dependence on the parameter N_{check} as well as on the parameters ϑ_1, ϑ_2 implicitly via p_j^{risk} in (8).

The working principles of the interaction-aware capacities of our approach are illustrated in Fig. 7. The priority values are visualized for all input cells and the state cell $\mathcal{S}_i$ in which the pedestrian is at time t_k. At the collision checking step $\kappa = 1$, the risk measure for moving at high speed towards the road is higher than moving at normal walking speed or slowing down. Although moving at high speed towards the road can probably avoid colliding with the car coming from the left side at the collision checking step $\kappa = 2$, it can lead to a dangerous situation at $\kappa = 3$ regarding another car coming from the right side. Note that multiple road vehicles are dealt with in (8) and the risk measures can be applied to other predicted pedestrians, which makes the computation of priority values easy to scale up. The resulting priority values (cf. Fig. 7 left, with $N_{\text{check}} = 3$) control the input transition probabilities for the state cell $\mathcal{S}_i$ as in (4). Such a process repeats for each state cell and each prediction time step.

C. Optimization

Incorporating collision probabilities in the interaction model with road vehicles does not assure that pedestrians will never enter the driving corridor of approaching vehicles. Apart from the risk measure, other factors can affect the crossing decision of pedestrians. For instance, in the same situation where a vehicle is approaching, it is more likely that pedestrians will traverse the road in front of that vehicle if they are currently walking at a relatively higher speed [85]. Hence, the interaction model with road vehicles, i.e., the parameters ϑ_1, ϑ_2 in (7) and N_{check} in (9), need to be calibrated using empirical trajectories by taking into account the learned transition probabilities $\Psi_{i,z}^{\alpha\beta}$ in (4).

We optimize the interaction model with road vehicles using the selected scenarios where, e.g., a pedestrian is near the curb and a vehicle is approaching. This accelerates the optimization process while emphasizing relevant interactions. The optimization is performed using genetic algorithms [86],

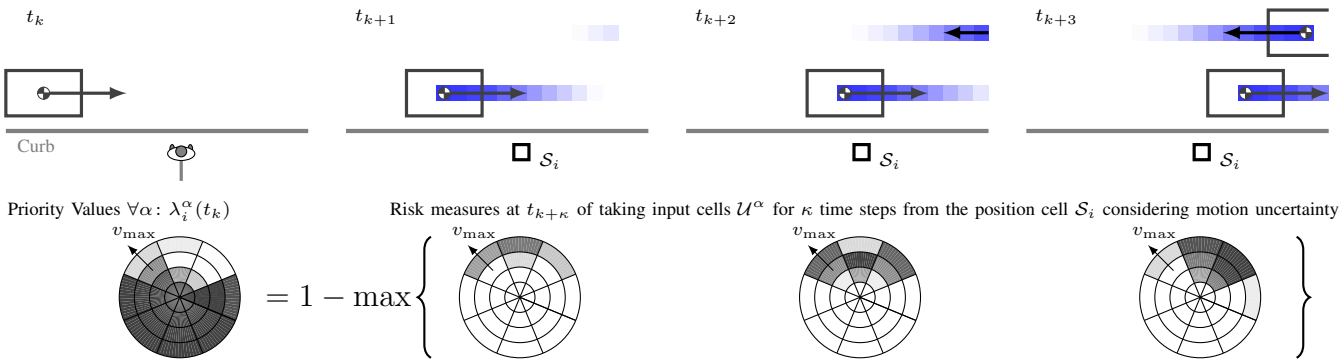


Fig. 7. Illustration of computing priority values in (9), where a pedestrian approaches two road vehicles. The input space (cf. Fig. 2) relates to the state cell $\mathcal{S}_i$ in which the pedestrian is at time t_k. The darker gray colors in the most left drawing of the input space mean higher priority values, whereas they represent higher risk measures in other drawings of the input space.

[87] which are especially well suited for the problems where derivative information is unavailable and certain variables are integer-valued. We minimize the negative log likelihood of occupancy given the ground truth positions

$$\begin{aligned} &\forall n, k, \exists i\colon s_{\mathrm{gt}}^{(n)}(t_k) \in \mathcal{S}_i, \\ &\overline{\mathrm{NLL}} = -\frac{1}{N_{\mathrm{opt}} N_{\mathrm{pred}}} \sum_{n=1}^{N_{\mathrm{opt}}} \sum_{k=1}^{N_{\mathrm{pred}}} \log P(s^{(n)}(t_k) \in \mathcal{S}_i) \end{aligned} \tag{10}$$

as the objective function, where N_{opt} is the number of used scenarios for optimization, N_{pred} is the number of prediction steps, and $P(s^{(n)}(t_k) \in \mathcal{S}_i)$ is the occupancy probability of the state cell $\mathcal{S}_i$ to which the ground truth position $s_{\mathrm{gt}}^{(n)}(t_k)$ from the nth scenario at time t_k belongs. More details on our optimization settings can be found in Appendix B.

V. Experiments

A. Datasets

We evaluate our approach on the Stanford drone dataset [88] (SDD), the inD dataset [89], the Daimler dataset [23], and our in-house dataset [45]. On the Stanford drone dataset, we follow the common train-test split as in [59]. From the inD dataset, we randomly select 6 out of 33 recordings for testing. Our in-house dataset for urban traffic scenarios was recorded by a test vehicle on three different routes varying between 2 and 4 kilometers in southern Germany. We use 5000 pedestrian trajectories for training, while 500 diverse scenes are kept for testing. The average trajectory length in our in-house dataset is about 7.5 seconds. In addition, we use the Daimler dataset to test model performance focusing on the pedestrian-vehicle interaction scenarios. While we use the benchmarked observation horizon of 3.2 seconds and prediction horizon of 4.8 seconds on the Stanford drone dataset, the observation horizon and prediction horizon on the other used datasets are set to 1 and 5 seconds, respectively.

B. Evaluation Metrics

1) Negative log likelihood of occupancy: To enable a comparison between prediction results with and without closed-form expressions for the probability density function, we compute the negative log likelihood of occupancy by returning the predicted occupancy probability of the state cell to which the ground truth position belongs. Given the ground truth position $s_{\mathrm{gt}}^{(n)}(t_k)$ from the nth evaluation scenario, the negative log likelihood of occupancy is evaluated by

$$\begin{aligned} &\forall n, k, \exists i\colon s_{\mathrm{gt}}^{(n)}(t_k) \in \mathcal{S}_i, \\ &\qquad \mathrm{NLL}^{(n)}(t_k) = -\log P(s^{(n)}(t_k) \in \mathcal{S}_i). \end{aligned} \tag{11}$$

2) Minimum final displacement error from samples: Compared to the negative log likelihood of occupancy, geometric metrics, such as the average displacement error, have a straightforward physical interpretation. However, using geometric metrics can be misleading when they are applied to probabilistic prediction by averaging over the (multimodal) distribution [17]. To encourage diversity in motion prediction, geometric metrics can be adapted to measure the minimum error between the ground truth position and K samples drawing from occupancy probabilities [43]. For the nth evaluation scenario at time t_k, we denote the predicted position of the ith sample by $s^{(n,i)}(t_k)$. Given the ground truth position $s_{\mathrm{gt}}^{(n)}(t_k)$, the minimum displacement error from K samples is computed as

$$\mathrm{minFDE}_K^{(n)}(t_k) = \min_{i \in \{1,\dots,K\}} \left\| s^{(n,i)}(t_k) - s_{\mathrm{gt}}^{(n)}(t_k) \right\|_2. \tag{12}$$

3) Offroad rate: To evaluate how well a model interprets the underlying semantic map, Deo and Trivedi [11] adopt the metric offroad rate which is defined as the fraction of the predicted position points that fall outside the path. We report the offroad rate of our approach on the Stanford drone dataset.

4) Precision-recall curve: In pedestrian-vehicle interaction scenarios, the crossing intention of a pedestrian can be inferred from their probabilistic occupancy as in [71]. Thus, we compute the pedestrian's probability of staying on the sidewalk at points in time and use the precision-recall curve [90] as an evaluation metric.

5) Negative log likelihood of demonstrated trajectory under predicted policy: To assess the quality of the learned reward function, we compute the negative log likelihood of a demonstrated trajectory ζ (spatial aspect, cf. Sec. III-C3) under the

predicted orientation distribution $\pi(\psi_{\overline{\alpha}}|s_i)$ in dependence on reward maps:

$$\mathrm{NLL}(\zeta) = -\frac{1}{|\zeta|}\log \prod_{(s_i,\psi_{\overline{\alpha}})\in\zeta} \pi(\psi_{\overline{\alpha}}|s_i), \tag{13}$$

which measures the probability of drawing a demonstrated trajectory from the predicted policy distribution and is normalized by the trajectory length [9], [15], [65], [78]. The policy is computed by utilizing the ending position cell of the demonstrated trajectory as the terminal state.

C. Baseline Models

We compare our approach using controlled Markov chains (CMC) against a physics-based approach—Kalman filter (KF) using the constant velocity model [91]. We implement two end-to-end baselines based on the mixture density network (MDN) [92] and the conditional variational autoencoder (CVAE) [93], respectively. For both end-to-end baselines, the pedestrian's past trajectory is concatenated with the extracted features of the surrounding map including road vehicles by a pretrained MobileNetV2 network [94], which are then fed into a long short-term memory [42]. For the mixture density network baseline, we use 12 Gaussian mixtures and apply the Winner-Takes-All loss [95] to overcome mode collapse. As the conditional variational autoencoder baseline yields a non-parametric position distribution using Gaussian latent variables, we adopt the Best-of-Many [96] L2 loss during training as in [46]. In addition, an off-the-shelf kernel density function is used during evaluation to approximate the probability distribution of position from the sampled results first (as in [30]), based on which we then compute the probabilistic occupancy (cf. Sec. V-B1) for a fair comparison. Moreover, we compare our approach with the reported results of the state-of-the-art approaches [11], [12], [48], [59], [70] on the Stanford drone dataset.

D. Quantitative Evaluation

Table I reports the negative log likelihood of occupancy and minimum final displacement error[1] at different prediction times in testing datasets. The standard deviations are reported behind each mean value. On our in-house dataset, our approach outperforms the baselines, yielding not only the lowest mean values, but also the lowest standard deviations. On the inD dataset, the mixture density network baseline shows the lowest mean values of both metrics. Although the mean performance of our approach is slightly worse than the end-to-end baselines, our approach yields relatively lower standard deviations of both metrics and tends to provide more accurate long-term motion predictions.

However, by analyzing the autocorrelation of trajectories [72] based on the "oriented step-length" [97] in various datasets (additionally the ETH dataset [2]), it turns out that the inD dataset exhibits smoother tracking results than others, cf. Fig. 8. This can be disproportionately advantageous for the Kalman filter especially for a short prediction horizon (cf. Table I). In order to investigate how models perform with realistic, noisy measurements, we sample the positions in x and y direction from a Gaussian distribution with the original positions as the mean and with different noise levels (with different standard deviations σ_{noise}). The autocorrelation values for the trajectories from the inD dataset after the above modification are plotted in the dashed lines in Fig. 8, where the dashed line with noise level $\sigma_{\mathrm{noise}} = 0.15\,\mathrm{m}$ achieves the same level of autocorrelation as in other datasets. Then, we test the model performance by using those noisy measurements as model input during prediction. The models are kept the same as those trained by using original measurements. The purpose of this experimental setting is to evaluate the generalization ability. As presented in Table II, for a moderate noise level with $\sigma_{\mathrm{noise}} = 0.1\,\mathrm{m}$ our approach outperforms the baselines in terms of negative log likelihood of occupancy, while the performance gap in terms of minimum final displacement error between both end-to-end baselines and our approach becomes smaller. Under the realistic, noisy position measurements with $\sigma_{\mathrm{noise}} = 0.15\,\mathrm{m}$, our approach outperforms the baselines in terms of all metrics. Conspicuously, the mixture density network baseline degenerates significantly in this case and its performance in terms of negative log likelihood of occupancy even drops below that of the Kalman filter. In contrast, the mean performance of our approach decreases only slightly and its standard deviations remain low, which shows the robustness of our approach against noisy measurements.

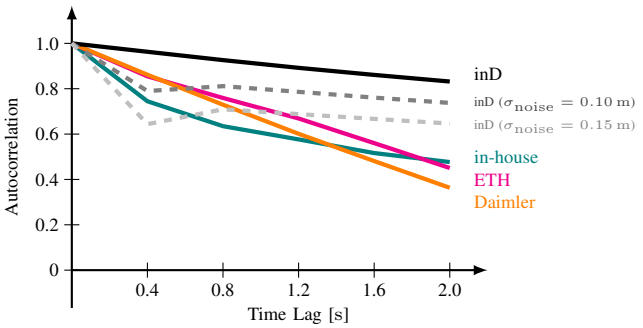


Fig. 8. Autocorrelation of trajectories in various datasets.

The minimum final displacement errors of the recent approaches [11], [12], [48], [59], [70] on the Stanford drone dataset are reported in Table III. Although the approaches in [48], [59], [70] achieve the lower displacement errors in pixels, our approach shows better performance than the end-to-end approach in [12] and the approach using planning in [11]. Furthermore, a comparatively low offroad rate of our approach can be observed as the model P2T [11], which suggests that our approach interprets the road layout well.

Beyond the overall system evaluation, we assessed the quality of our learned reward function by using the metric in (13). According to the mean and standard deviation of this metric evaluated on our in-house testing dataset (where the

[1] We do not consider the metric minimum average displacement error to measure the prediction quality throughout the pedestrian's journey, as it is unusual to sample trajectories from probabilistic occupancies which propagate at time steps, such as in [10]. By trying to do this, the sampled trajectories can be jittery (we argue that this is at least due to the discretization of state and input space).

TABLE I
MODEL PERFORMANCE ON OUR IN-HOUSE DATASET [45] AND IND DATASET [89].

Model	NLL ↓					minFDE_{20} [m] ↓
	1 s	2 s	3 s	4 s	5 s	5 s
(in-house dataset)						
KF [91]	$2.20_{\pm 2.06}$	$3.48_{\pm 2.36}$	$4.38_{\pm 2.37}$	$5.10_{\pm 2.41}$	$5.68_{\pm 2.41}$	$1.00_{\pm 1.07}$
MDN [92]	$2.15_{\pm 1.63}$	$3.38_{\pm 1.67}$	$4.27_{\pm 1.82}$	$4.89_{\pm 1.86}$	$5.37_{\pm 1.77}$	$0.91_{\pm 0.83}$
CVAE [93]	$2.97_{\pm 9.72}$	$3.60_{\pm 5.31}$	$4.30_{\pm 3.74}$	$4.85_{\pm 2.84}$	$5.36_{\pm 2.54}$	$0.89_{\pm 0.92}$
CMC (ours)	$\mathbf{2.09_{\pm 1.34}}$	$\mathbf{3.23_{\pm 1.40}}$	$\mathbf{4.02_{\pm 1.43}}$	$\mathbf{4.66_{\pm 1.51}}$	$\mathbf{5.21_{\pm 1.62}}$	$\mathbf{0.81_{\pm 0.81}}$
(inD dataset, smoothed position measurements)						
KF [91]	$1.65_{\pm \mathbf{0.52}}$	$2.88_{\pm \mathbf{0.87}}$	$3.77_{\pm 1.11}$	$4.46_{\pm 1.28}$	$5.02_{\pm 1.39}$	$0.74_{\pm 0.69}$
MDN [92]	$\mathbf{1.10}_{\pm 2.29}$	$\mathbf{2.20}_{\pm 2.19}$	$\mathbf{3.09}_{\pm 2.38}$	$\mathbf{3.75}_{\pm 2.41}$	$\mathbf{4.24}_{\pm 2.28}$	$\mathbf{0.51}_{\pm 0.64}$
CVAE [93]	$2.47_{\pm 11.6}$	$2.74_{\pm 9.01}$	$3.32_{\pm 7.28}$	$3.91_{\pm 7.13}$	$4.41_{\pm 7.13}$	$\mathbf{0.51}_{\pm 0.70}$
CMC (ours)	$1.76_{\pm 0.63}$	$2.79_{\pm 0.91}$	$3.48_{\pm \mathbf{1.10}}$	$4.02_{\pm \mathbf{1.23}}$	$4.46_{\pm \mathbf{1.32}}$	$0.59_{\pm \mathbf{0.60}}$

The implementation details for comparison models can be found in Sec. V-C.

TABLE II
MODEL PERFORMANCE ON IND DATASET [89] UNDER REALISTIC, NOISY POSITION MEASUREMENTS.

Model	NLL ↓					minFDE_{20} [m] ↓
	1 s	2 s	3 s	4 s	5 s	5 s
(inD dataset, noisy position measurements, $\sigma_{\text{noise}} = 0.10\,\text{m}$)						
KF [91]	$\mathbf{1.98_{\pm 0.76}}$	$3.15_{\pm 1.02}$	$4.01_{\pm 1.25}$	$4.68_{\pm 1.40}$	$5.23_{\pm 1.51}$	$0.81_{\pm 0.75}$
MDN [92]	$2.04_{\pm 3.28}$	$3.01_{\pm 2.67}$	$3.86_{\pm 2.75}$	$4.48_{\pm 2.71}$	$4.91_{\pm 2.50}$	$0.64_{\pm 0.69}$
CVAE [93]	$4.26_{\pm 14.7}$	$3.58_{\pm 8.86}$	$3.95_{\pm 7.48}$	$4.42_{\pm 6.86}$	$4.86_{\pm 6.83}$	$\mathbf{0.60}_{\pm 0.76}$
CMC (ours)	$2.02_{\pm 0.81}$	$\mathbf{3.00_{\pm 1.00}}$	$\mathbf{3.66_{\pm 1.17}}$	$\mathbf{4.17_{\pm 1.28}}$	$\mathbf{4.60_{\pm 1.37}}$	$0.63_{\pm \mathbf{0.65}}$
(inD dataset, noisy position measurements, $\sigma_{\text{noise}} = 0.15\,\text{m}$)						
KF [91]	$2.38_{\pm 1.13}$	$3.50_{\pm 1.29}$	$4.33_{\pm 1.47}$	$4.97_{\pm 1.60}$	$5.50_{\pm 1.68}$	$0.91_{\pm 0.83}$
MDN [92]	$3.07_{\pm 4.42}$	$3.79_{\pm 3.33}$	$4.58_{\pm 3.38}$	$5.15_{\pm 3.30}$	$5.51_{\pm 3.01}$	$0.78_{\pm 0.76}$
CVAE [93]	$6.65_{\pm 19.0}$	$4.81_{\pm 11.6}$	$4.75_{\pm 8.51}$	$5.03_{\pm 7.66}$	$5.37_{\pm 7.32}$	$0.72_{\pm 0.87}$
CMC (ours)	$\mathbf{2.35_{\pm 1.09}}$	$\mathbf{3.26_{\pm 1.17}}$	$\mathbf{3.89_{\pm 1.30}}$	$\mathbf{4.38_{\pm 1.40}}$	$\mathbf{4.79_{\pm 1.47}}$	$\mathbf{0.69_{\pm 0.71}}$

The implementation details for comparison models can be found in Sec. V-C.

TABLE III
MODEL PERFORMANCE ON STANFORD DRONE DATASET [88].

Model	minFDE_5 [px] ↓	minFDE_{20} [px] ↓	Offroad rate ↓
	4.8 s	4.8 s	
PECNet [12]	25.98	15.88	-
P2T [11]	23.95	14.08	**0.06**
Y-Net [59]	**20.23**	11.85	-
Goal-SAR [48]	-	11.83	-
NSP-SFM [70]	-	**10.61**	-
CMC (ours)	23.87	13.35	0.07

semantic maps are available), the performance of our proposed module (with the learned reward maps) is improved from $1.48_{\pm 1.36}$ to $1.09_{\pm 0.83}$ compared to using the handcrafted reward maps [14].

E. Ablation Study

An ablation study is performed to understand the contribution of each component of our approach (with $|\mathcal{G}| = 12$) to the overall performance, cf. Fig. 9. We remove the stochastic policies ("w/o road layout") in (5), the velocity transition probabilities ("w/o velocity transition") in (5), and the priority values ("w/o road vehicles") in (4), respectively. As one would expect, the model without considering the road layout exhibits a significant decrease of performance. Furthermore, a proper velocity transition model contributes to an improvement of performance. Since critical scenarios with interaction are rare in most (large-scale) datasets just like in reality, the effect of our interaction model with road vehicles appears low and is hard to observe in the upper part of Fig. 9. Therefore, we additionally adopt the Daimler dataset [23] which contains critical pedestrian-vehicle interaction scenarios. In light of its limited amount of recordings, we directly test the model (trained on our in-house dataset) on this dataset. As described in Sec. V-B, the evaluation can be formulated as a classification problem—whether the pedestrian will be on the sidewalk or on the road. The precision-recall curves in the lower part of Fig. 9 emphasize the benefits of considering road vehicles via priority values in our approach.

In addition, we investigated the impact of the number of goal regions in our approach on the mean and standard deviation of the negative log likelihood of occupancy. As shown in Fig. 10, using a larger number of goal regions improves the model performance. While the mean values of evaluation metrics almost stagnate for $|\mathcal{G}| \geq 8$, their standard deviations are further reduced for $|\mathcal{G}| = 12$ compared to $|\mathcal{G}| = 8$. Hence, we set $|\mathcal{G}| = 12$ as a compromise between computational effort and the diversity of goal intention.

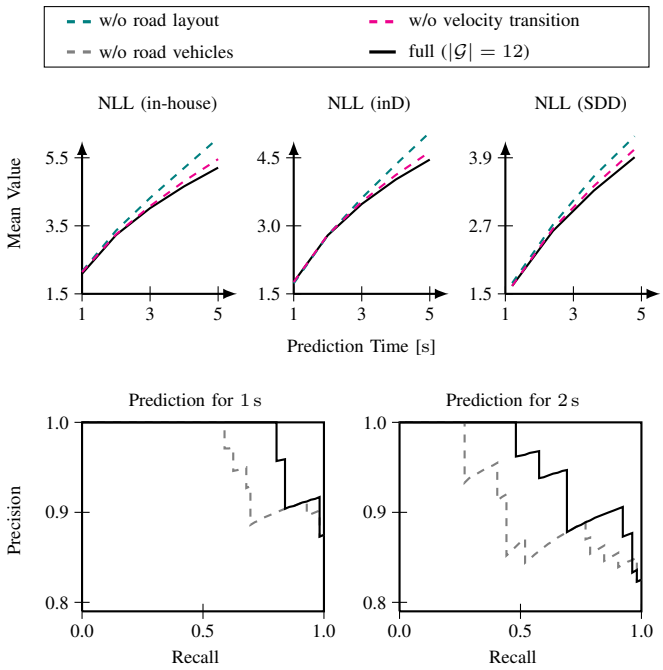


Fig. 9. Ablation study for our controlled Markov chains. The top row shows the mean values of the negative log likelihood of occupancy at different prediction times. The bottom row shows the precision-recall curves for classifying whether the pedestrian will be on the sidewalk.

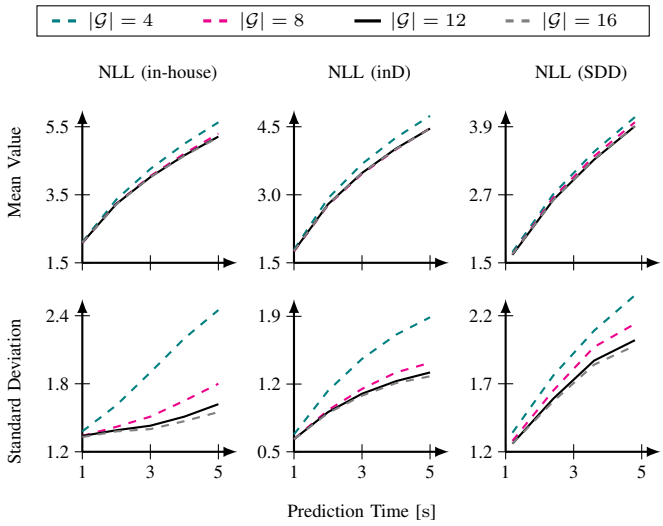


Fig. 10. The impact of the number of goal regions on mean and standard deviation of negative log likelihood of occupancy.

F. Qualitative Evaluation

The top row of Fig. 11 illustrates a scenario in the Daimler dataset, where a pedestrian is walking on the sidewalk towards an approaching vehicle. The predicted occupancies in 3 seconds (last column) depict that the pedestrian will most likely stop, whereas the ground truth position (red cross) indicates a crossing behavior in front of the vehicle. The bottom row of Fig. 11 contains the same trajectory recording with a later starting frame, i.e., the prediction beginning t_0 is shifted by

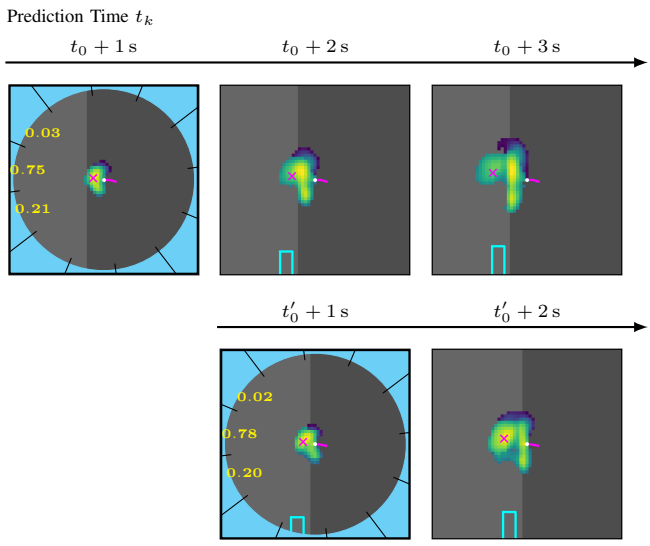


Fig. 11. Prediction examples of our controlled Markov chains in the Daimler dataset [23]. The semantic maps consist of the sidewalk (dark gray) and the road (light gray). The pedestrian's past trajectory and current position are denoted by the magenta line and the white dot, respectively. The probabilities of goal regions as intermediate results are shown in the first plot of each row (values less than 0.01 are discarded to improve visualization). While the magenta cross represents the ground truth positions of pedestrians at prediction times, the ground truth positions of road vehicles are shown in cyan rectangles. The predicted occupancies with lighter colors mean higher probabilities (we discard those occupancies with lower probabilities for a better visualization).

1 second compared to that in the top row. As shown by those predicted occupancies in 2 seconds (bottom row, last column), the probability of pedestrian crossing in front of the vehicle becomes higher as the pedestrian gets closer to the curb without slowing down. In such a dangerous situation, the vehicle had to brake to avoid a potential collision.

The prediction examples of our approach in our in-house dataset and the inD dataset are shown in Fig. 12. In the scene (a) where the pedestrian is walking near the curb, our approach predicts that the pedestrian will most likely keep walking along this side of the street. The example (b) shows a scene consisting of two zebra crossings. The predicted occupancies indicate that there is a certain probability that the pedestrian will choose the farther crosswalk later. In the scene (c), it is predicted that the pedestrian will most likely cross the road in front of an oncoming vehicle. Moreover, the ability of our approach to incorporate the cues from dynamic environments and the road layout[2] in motion prediction is also depicted in the examples (d) and (e). In the example (f), as a low walking speed of the pedestrian is observed, the predicted occupancies are located near the initial position of that pedestrian due to the learned velocity transition probabilities.

In the bottom row of Fig. 13, we visualize the risk measures on the road caused by multiple road vehicles considering their body geometry (cf. Sec. IV-A3). They are used for computing the time-varying priority values (cf. Sec. IV-B). Accordingly,

[2]Note that instead of using semantic maps, we directly use the provided RGB images from the inD dataset and Stanford drone dataset (as input to our convolutional neural networks) to compute reward values at each pixel.

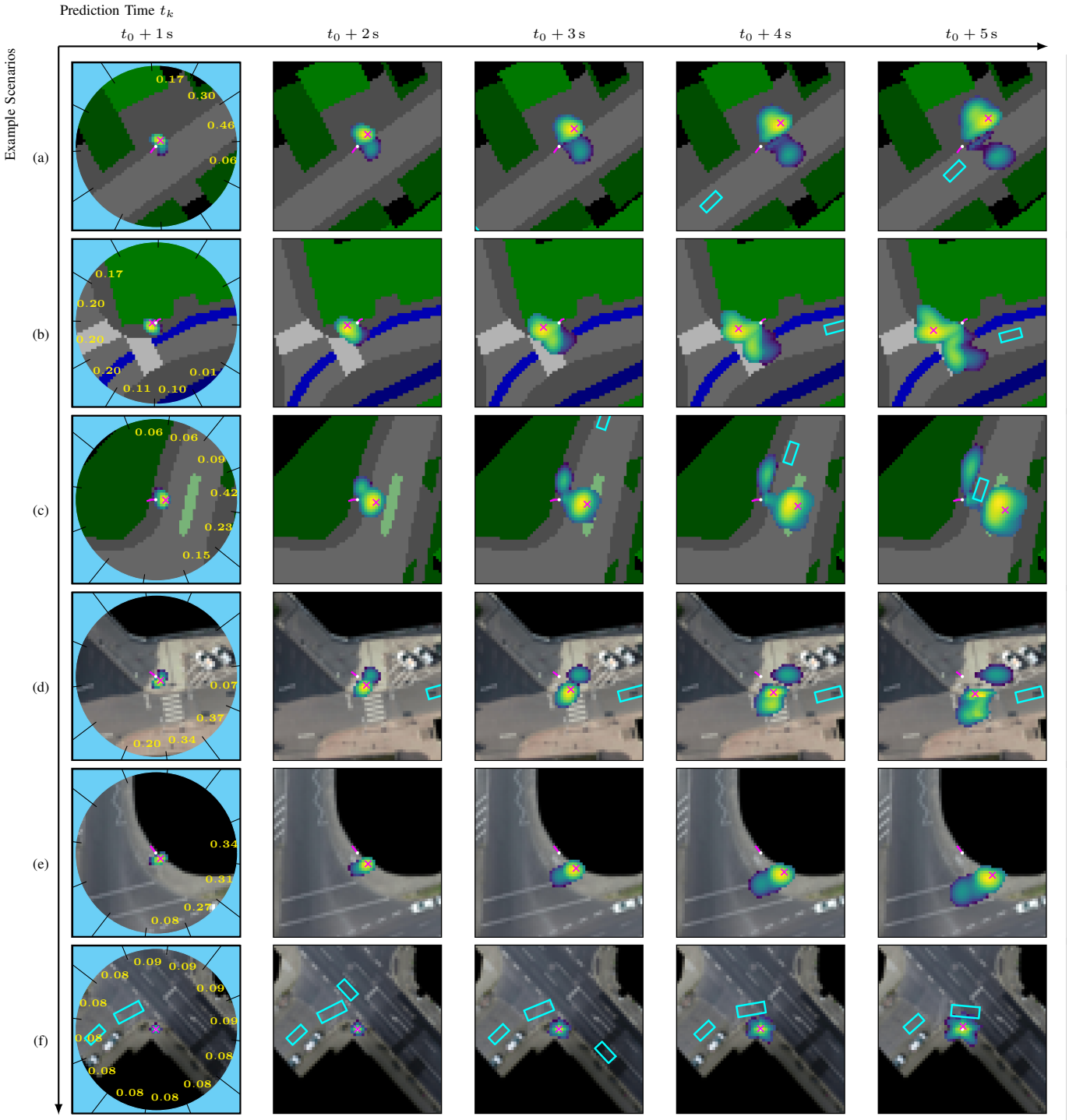


Fig. 12. Prediction examples of controlled Markov chains in our in-house dataset [45] (a)-(c) (the annotations of semantic maps see Fig. 5) and inD dataset [89] (d)-(f). The pedestrian's past trajectory and current position are denoted by the magenta line and the white dot, respectively. The probabilities of goal regions as intermediate results are shown in the first plot of each row (values less than 0.01 are discarded to improve visualization). While the magenta cross represents the ground truth positions of pedestrians at prediction times, the ground truth positions of road vehicles are shown in cyan rectangles. The predicted occupancies with lighter colors mean higher probabilities (we discard those occupancies with lower probabilities for a better visualization).

our approach predicts that the pedestrian will most likely slow down when approaching the curb.

More prediction examples of our approach on the Stanford drone dataset are shown in Fig. 14. The examples (a) and (b) illustrate that our approach interprets the semantic scenes well and predicts that pedestrians avoid obstacles. In the scene (c), a pedestrian has stepped into the green space. Our predictions indicate that the pedestrian continues walking on that terrain.

As opposed to sampling-based approaches, the predicted probabilistic occupancies from our approach are deterministic.

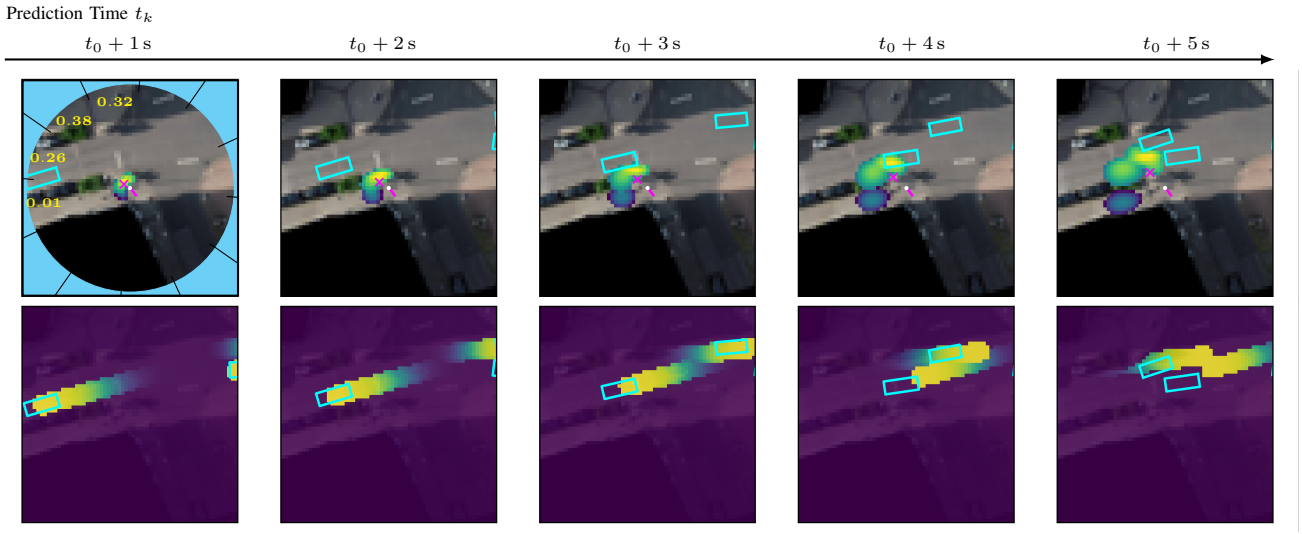


Fig. 13. Prediction example of controlled Markov chains in the inD dataset [89]. The annotations of the observations, our prediction results, and the ground truth information see Fig. 11 or Fig. 12. The second row visualizes the risk measures on the road caused by multiple road vehicles considering their body geometry (lighter colors mean higher risk measures).

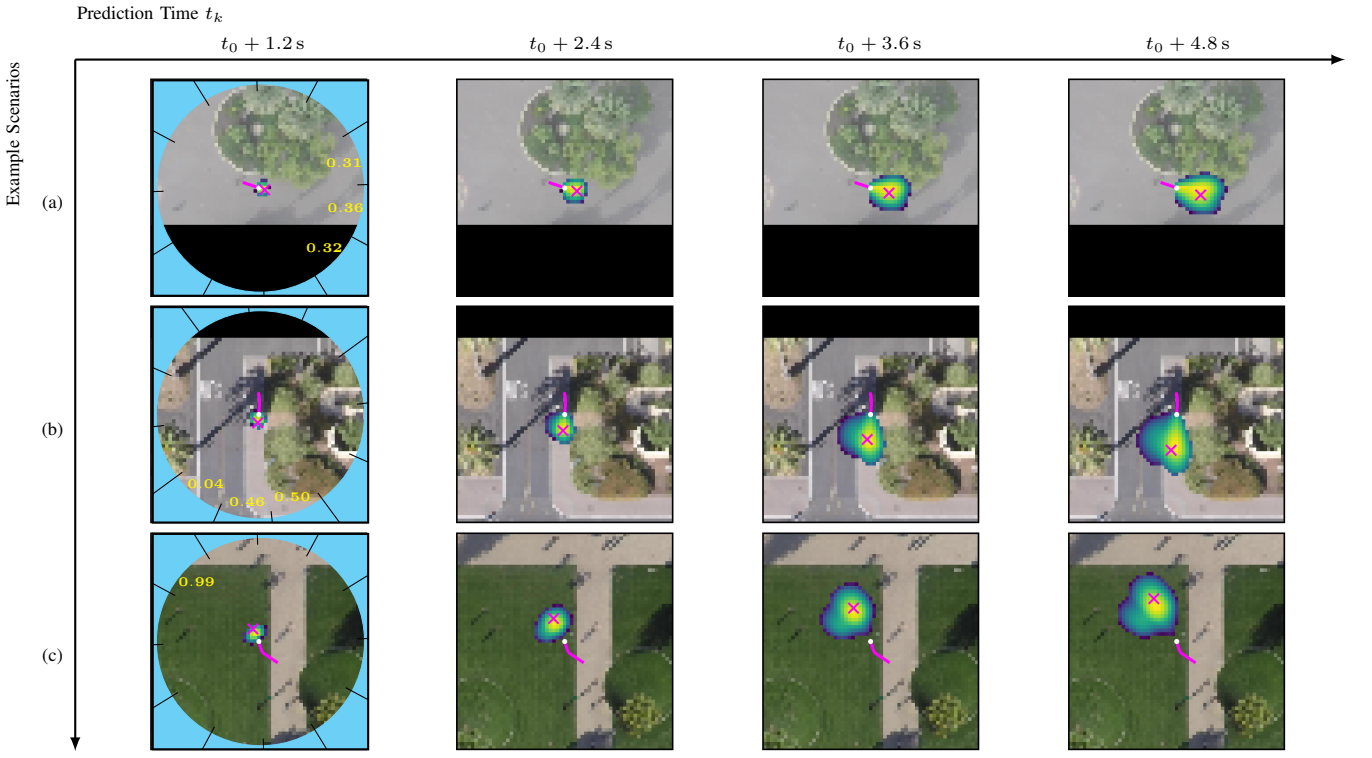


Fig. 14. Prediction examples of controlled Markov chains in the Stanford drone dataset [88]. The annotations of the observations, our prediction results, and the ground truth information see Fig. 11 or Fig. 12.

This benefits online risk assessment and motion planning for ego vehicles without confronting the need to define, for example, the necessary number of sampled trajectories from the prediction model to ensure a reliable assessment.

VI. Conclusions

Incorporating context cues in Markov chains enables us to forecast multimodal behavior of pedestrians without relying on random sampling. This makes it possible to compute the probabilities deterministically, which is important for certification and reliability of the method. As opposed to black-box deep learning, we present a more structured approach,

TABLE IV
HYPERPARAMETERS.

Option	Value	Option	Value
populationSize	30	*numberGeneration*	50
select	"selTournament"	*tournamentSize*	5
mate	"cxOnePoint"	*mateProb*	0.7
mutationProb	0.3	*mutationIndProb*	0.3

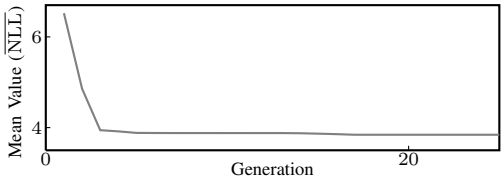


Fig. 15. Convergence of objective value of candidate solutions.

which provides interpretable prediction as well as intermediate results. This benefits not only further online assessment, but also system inspection. Specifically, we address the difficulty of obtaining prior knowledge of goal positions in unknown environments for most planning-based approaches and propose a technique without strong assumptions.

Evaluated on large-scale datasets, our approach yields more accurate motion prediction results than a physics-based model and achieves comparable mean performance as state-of-the-art approaches, but with lower standard deviations of prediction errors. In particular, we show the robustness of our approach against realistic measurement noises, which is a common issue in practical applications in the field of motion prediction.

Our approach has some limitations. Regarding the pedestrian interaction with other traffic participants, we considered only road vehicles. We aim to extend the interaction module and the application scenarios in future work by considering, e.g., interactions among pedestrians and pedestrian group behaviors. Although we focused on integrating various techniques in the proposed prediction pipeline, different backbone models (e.g., U-net [98]) can be investigated and used as our reward function approximator for achieving better prediction performance. Furthermore, the computational effort of updating input and state probabilities based on our current CPU-oriented implementation is about 150 milliseconds for 10 time steps for each goal region (measured on a machine with an Intel i7-11850H 2.50 GHz processor). Implementing these operations on a specific hardware can be a way to make our approach deployable for real-time applications.

APPENDIX A
CONVOLUTIONAL NEURAL NETWORK IMPLEMENTATION

The architecture of our convolutional neural networks (one takes the semantic maps as input, and the other one takes the RGB satellite images as input) is as follows:

$$C^{32}_{3\times3,1} \rightarrow C^{16}_{5\times5,2} \rightarrow C^{16}_{5\times5,2} \rightarrow C^{16}_{3\times3,2} \rightarrow C^{16}_{3\times3,2} \rightarrow C^{2}_{3\times3,1}$$

All convolutional layers are denoted as $C^{\text{number of filters}}_{\text{kernel size, dilation rate}}$ with the rectified linear unit activation function. The negation of output values are used as rewards. The architecture is chosen based on a hyperparameter search involving the number of layers, kernel size, and the number of filters by evaluating the negative log likelihood of demonstrated trajectories.

During backpropagation (cf. Sec. III-C3), L1 regularization is used in computing the gradient of the loss function, cf. [65, (2)]. In addition, we augment the training data by rotating and flipping the maps and demonstrated trajectories to reduce overfitting.

APPENDIX B
GENETIC ALGORITHMS

For the optimization process described in Sec. IV-C, we use the evolutionary computation framework DEAP [86] with the hyperparameters listed in Table IV.

We initialize the candidate solutions by uniformly sampling the parameters $\vartheta_1 \in [0, 15]$, $\vartheta_2 \in [0, 15]$, and $N_{\text{check}} \in \{1, \ldots, 8\}$. The Gaussian addition mutation of the mean 0 and standard deviation of 1.0 is applied to both ϑ_1 and ϑ_2, while N_{check} is drawn from the discrete uniform distribution during mutation. The convergence of the objective value in (10) of candidate solutions on our in-house dataset is shown in Fig. 15.

ACKNOWLEDGMENT

The authors appreciate the insightful comments and valuable suggestions from the anonymous reviewers.

REFERENCES

[1] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Physical Review E*, vol. 51, no. 5, pp. 4282–4286, 1995.
[2] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *Proc. of the IEEE International Conference on Computer Vision*, 2009, pp. 261–268.
[3] F. Zanlungo, T. Ikeda, and T. Kanda, "Social force model with explicit collision prediction," *Europhysics Letters*, vol. 93, no. 6, 2011, article no. 68005.
[4] M. Hussein and T. Sayed, "A bi-directional agent-based pedestrian microscopic model," *Transportmetrica A: Transport Science*, vol. 13, no. 4, pp. 326–355, 2017.
[5] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 961–971.
[6] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GAN: Socially acceptable trajectories with generative adversarial networks," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.
[7] R. Korbmacher and A. Tordeux, "Review of pedestrian trajectory prediction methods: Comparing deep learning and knowledge-based approaches," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24 126–24 144, 2022.
[8] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 3931–3936.
[9] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, "Activity forecasting," in *Proc. of the European Conference on Computer Vision*, 2012, pp. 201–214.
[10] E. Rehder, F. Wirth, M. Lauer, and C. Stiller, "Pedestrian prediction by planning using deep neural networks," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2018, pp. 5903–5908.
[11] N. Deo and M. M. Trivedi, "Trajectory forecasts in unknown environments conditioned on grid-based plans," *arXiv preprint arXiv:2001.00735v2*, 2021.

[12] K. Mangalam, H. Girase, S. Agarwal, K.-H. Lee, E. Adeli, J. Malik, and A. Gaidon, "It is not the journey but the destination: Endpoint conditioned trajectory prediction," in *Proc. of the European Conference on Computer Vision*, 2020, pp. 759–776.

[13] V. Karasev, A. Ayvaci, B. Heisele, and S. Soatto, "Intent-aware long-term prediction of pedestrian motion," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2016, pp. 2543–2549.

[14] J. Wu, J. Ruenz, and M. Althoff, "Probabilistic map-based pedestrian motion prediction taking traffic participants into consideration," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2018, pp. 1285–1292.

[15] A. Rudenko, L. Palmieri, and K. O. Arras, "Joint long-term prediction of human motion using a planning-based social force approach," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2018, pp. 4571–4577.

[16] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. The MIT Press, 2018.

[17] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," *The International Journal of Robotics Research*, vol. 39, no. 8, pp. 895–935, 2020.

[18] F. Camara, N. Bellotto, S. Cosar, F. Weber, D. Nathanael, M. Althoff, J. Wu, J. Ruenz, A. Dietrich, G. Markkula, A. Schieben, F. Tango, N. Merat, and C. Fox, "Pedestrian models for autonomous driving part II: High-level models of human behavior," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 5453–5472, 2021.

[19] N. Schneider and D. M. Gavrila, "Pedestrian path prediction with recursive Bayesian filters: A comparative study," in *Proc. of the German Conference on Pattern Recognition*, 2013, pp. 174–183.

[20] C. Schöller, V. Aravantinos, F. Lay, and A. Knoll, "What the constant velocity model can teach us about pedestrian motion prediction," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1696–1703, 2020.

[21] W. Zeng, P. Chen, G. Yu, and Y. Wang, "Specification and calibration of a microscopic model for pedestrian dynamic simulation at signalized intersections: A hybrid approach," *Transportation Research Part C: Emerging Technologies*, vol. 80, pp. 37–70, 2017.

[22] A. Rasouli and I. Kotseruba, "Intend-wait-cross: Towards modeling realistic pedestrian crossing behavior," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2022, pp. 83–90.

[23] J. F. P. Kooij, N. Schneider, F. Flohr, and D. M. Gavrila, "Context-based pedestrian path prediction," in *Proc. of the European Conference on Computer Vision*, 2014, pp. 618–633.

[24] I. Batkovic, M. Zanon, N. Lubbe, and P. Falcone, "A computationally efficient model for pedestrian motion prediction," in *Proc. of the European Control Conference*, 2018, pp. 374–379.

[25] M. Koschi, C. Pek, M. Beikirch, and M. Althoff, "Set-based prediction of pedestrians in urban environments considering formalized traffic rules," in *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, 2018, pp. 2704–2711.

[26] D. Ellis, E. Sommerlade, and I. Reid, "Modelling pedestrian trajectory patterns with Gaussian processes," in *Proc. of the IEEE International Conference on Computer Vision Workshops*, 2009, pp. 1229–1234.

[27] T. P. Kucner, M. Magnusson, E. Schaffernicht, V. H. Bennetts, and A. J. Lilienthal, "Enabling flow awareness for mobile robots in partially observable environments," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1093–1100, 2017.

[28] B. Völz, K. Behrendt, H. Mielenz, I. Gilitschenski, R. Siegwart, and J. Nieto, "A data-driven approach for pedestrian intention estimation," in *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, 2016, pp. 2607–2612.

[29] Z. Fang and A. M. López, "Is the pedestrian going to cross? Answering by 2D pose estimation," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2018, pp. 1271–1276.

[30] B. Ivanovic and M. Pavone, "The Trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2375–2384.

[31] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, "Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 412–14 420.

[32] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5308–5317.

[33] J. Liang, L. Jiang, J. C. Niebles, A. G. Hauptmann, and L. Fei-Fei, "Peeking into the future: Predicting future person activities and locations in videos," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2960–2963.

[34] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, and S. Savarese, "Social-BiGAT: Multimodal trajectory forecasting using bicycle-GAN and graph attention networks," in *Proc. of the International Conference on Neural Information Processing Systems*, 2019, pp. 137–146.

[35] T. Yagi, K. Mangalam, R. Yonetani, and Y. Sato, "Future person localization in first-person videos," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7593–7602.

[36] Y. Cai, L. Dai, H. Wang, L. Chen, Y. Li, M. A. Sotelo, and Z. Li, "Pedestrian motion trajectory prediction in intelligent driving from far shot first-person perspective video," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 5298–5313, 2022.

[37] A. Jain, S. Casas, R. Liao, Y. Xiong, S. Feng, S. Segal, and R. Urtasun, "Discrete residual flow for probabilistic pedestrian behavior prediction," in *Proc. of the Conference on Robot Learning*, 2020, pp. 407–419.

[38] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "HOME: Heatmap output for future motion estimation," in *Proc. of the IEEE Intelligent Transportation Systems Conference*, 2021, pp. 500–507.

[39] D. Ridel, N. Deo, D. Wolf, and M. Trivedi, "Scene compliant trajectory forecast with agent-centric spatio-temporal grids," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2816–2823, 2020.

[40] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "VectorNet: Encoding HD maps and agent dynamics from vectorized representation," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 522–11 530.

[41] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, C. Li, and D. Anguelov, "TNT: Target-driven trajectory prediction," in *Proc. of the Conference on Robot Learning*, 2020, pp. 895–904.

[42] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[43] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker, "DESIRE: Distant future prediction in dynamic scenes with interacting agents," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2165–2174.

[44] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *Proc. of the European Conference on Computer Vision*, 2020, pp. 683–700.

[45] M. Herman, J. Wagner, V. Prabhakaran, N. Möser, H. Ziesche, W. Ahmed, L. Bürkle, E. Kloppenburg, and C. Gläser, "Pedestrian behavior prediction for automated driving: Requirements, metrics, and relevant features," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 922–14 937, 2022.

[46] Y. Yao, E. Atkins, M. Johnson-Roberson, R. Vasudevan, and X. Du, "BiTraP: Bi-directional pedestrian trajectory prediction with multi-modal goal estimation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1463–1470, 2021.

[47] F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, "Transformer networks for trajectory forecasting," in *Proc. of the International Conference on Pattern Recognition*, 2021, pp. 10 335–10 342.

[48] L. F. Chiara, P. Coscia, S. Das, S. Calderara, R. Cucchiara, and L. Ballan, "Goal-driven self-attentive recurrent networks for trajectory prediction," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2022, pp. 2517–2526.

[49] T. Gu, G. Chen, J. Li, C. Lin, Y. Rao, J. Zhou, and J. Lu, "Stochastic trajectory prediction via motion indeterminacy diffusion," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 092–17 101.

[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. of the International Conference on Neural Information Processing Systems*, 2017, pp. 6000–6010.

[51] L. Li, M. Pagnucco, and Y. Song, "Graph-based spatial transformer with memory replay for multi-future pedestrian trajectory prediction," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2221–2231.

[52] I. Bae and H.-G. Jeon, "A set of control points conditioned pedestrian trajectory prediction," in *Proc. of the AAAI Conference on Artificial Intelligence*, 2023, pp. 6155–6165.

[53] O. Makansi, E. Ilg, Ö. Cicek, and T. Brox, "Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7137–7146.

[54] P. Dendorfer, S. Elflein, and L. Leal-Taixé, "MG-GAN: A multi-generator model preventing out-of-distribution samples in pedestrian tra-

jectory prediction," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 13 138–13 147.

[55] H. Berkemeyer, R. Franceschini, T. Tran, L. Che, and G. Pipa, "Feasible and adaptive multimodal trajectory prediction with semantic maneuver fusion," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2021, pp. 8530–8536.

[56] R. Hug, S. Becker, W. Hübner, and M. Arens, "Particle-based pedestrian path prediction using LSTM-MDL models," in *Proc. of the IEEE Intelligent Transportation Systems Conference*, 2018, pp. 2684–2691.

[57] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," in *Proc. of the International Conference on Learning Representations*, 2017.

[58] R. Huang, H. Xue, M. Pagnucco, F. Salim, and Y. Song, "Multimodal trajectory prediction: A survey," *arXiv preprint arXiv:2302.10463*, 2023.

[59] K. Mangalam, Y. An, H. Girase, and J. Malik, "From goals, waypoints & paths to long term human trajectory forecasting," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 213–15 222.

[60] C. Wang, Y. Wang, M. Xu, and D. J. Crandall, "Stepwise goal-driven networks for trajectory prediction," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2716–2723, 2022.

[61] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *Proc. of the International Conference on Machine Learning*, 2015, pp. 2256–2265.

[62] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. of the International Conference on Neural Information Processing Systems*, 2020, pp. 6840–6851.

[63] J. Luettin, S. Monka, C. Henson, and L. Halilaj, "A survey on knowledge graph-based methods for automated driving," in *Proc. of the Iberoamerican Knowledge Graphs and Semantic Web Conference*, 2022, pp. 16–31.

[64] H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard, "Socially compliant mobile robot navigation via inverse reinforcement learning," *The International Journal of Robotics Research*, vol. 35, no. 11, pp. 1289–1307, 2016.

[65] M. Wulfmeier, D. Rao, D. Z. Wang, P. Ondruska, and I. Posner, "Large-scale cost function learning for path planning using deep inverse reinforcement learning," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1073–1087, 2017.

[66] W.-C. Ma, D.-A. Huang, N. Lee, and K. M. Kitani, "Forecasting interactive dynamics of pedestrians with fictitious play," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4636–4644.

[67] G. W. Brown, "Iterative solution of games by fictitious play," in *Activity Analysis of Production and Allocation*, T. C. Koopmans, Ed. New York: Wiley, 1951, pp. 374–376.

[68] R. Trumpp, H. Bayerlein, and D. Gesbert, "Modeling interactions of autonomous vehicles and pedestrians with deep multi-agent reinforcement learning for collision avoidance," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2022, pp. 331–336.

[69] S. Hossain, F. T. Johora, J. P. Müller, S. Hartmann, and A. Reinhardt, "SFMGNet: A physics-based neural network to predict pedestrian trajectories," in *Proc. of the AAAI Spring Symposium on Machine Learning and Knowledge Engineering for Hybrid Intelligence*, 2022, pp. 1–16.

[70] J. Yue, D. Manocha, and H. Wang, "Human trajectory prediction via neural social physics," in *Proc. of the European Conference on Computer Vision*, 2022, pp. 376–394.

[71] J. Wu, J. Ruenz, and M. Althoff, "Calibration of controlled Markov chains for predicting pedestrian crossing behavior using multi-objective genetic algorithms," in *Proc. of the IEEE Intelligent Transportation Systems Conference*, 2019, pp. 1032–1038.

[72] M. Althoff and A. Mergel, "Comparison of Markov chain abstraction and Monte Carlo simulation for the safety assessment of autonomous cars," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1237–1247, 2011.

[73] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

[74] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov, "Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction," in *Proc. of the Conference on Robot Learning*, 2020, pp. 86–99.

[75] M. L. Littman, "Reinforcement learning improves behaviour from evaluative feedback," *Nature*, vol. 521, pp. 445–451, 2015.

[76] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, pp. 193–202, 1980.

[77] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[78] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proc. of the AAAI Conference on Artificial Intelligence*, 2008, pp. 1433–1438.

[79] E. T. Jaynes, "Information theory and statistical mechanics," *Physical Review*, vol. 106, pp. 620–630, 1957.

[80] Y. Zhang, W. Wang, R. Bonatti, D. Maturana, and S. Scherer, "Integrating kinematics and environment context into deep inverse reinforcement learning for predicting off-road vehicle trajectories," *arXiv preprint arXiv:1810.07225*, 2018.

[81] B. D. Ziebart, "Modeling purposeful adaptive behavior with the principle of maximum causal entropy," Ph.D. dissertation, Carnegie Mellon University, 2010.

[82] M. Althoff, "Reachability analysis and its application to the safety assessment of autonomous cars," Ph.D. dissertation, Technische Universität München, 2010.

[83] M. Brewer, K. Fitzpatrick, J. Whitacre, and D. Lord, "Exploration of pedestrian gap-acceptance behavior at selected locations," *Transportation Research Record*, vol. 1982, no. 1, pp. 132–140, 2006.

[84] J. Jansson and F. Gustafsson, "A framework and automotive application of collision avoidance decision making," *Automatica*, vol. 44, no. 9, pp. 2347–2351, 2008.

[85] M. M. Hamed, "Analysis of pedestrians behavior at pedestrian crossings," *Safety Science*, vol. 38, no. 1, pp. 63–82, 2001.

[86] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, pp. 2171–2175, 2012.

[87] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., 1989.

[88] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *Proc. of the European Conference on Computer Vision*, 2016, pp. 549–565.

[89] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein, "The inD dataset: A drone dataset of naturalistic road user trajectories at German intersections," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2020, pp. 1929–1934.

[90] J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proc. of the International Conference on Machine Learning*, 2006, pp. 233–240.

[91] S. Y. Chen, "Kalman filter for robot vision: A survey," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4409–4420, 2012.

[92] C. M. Bishop, "Mixture density networks," Aston University, Working-Paper, 1994.

[93] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Proc. of the International Conference on Neural Information Processing Systems*, 2015, pp. 3483–3491.

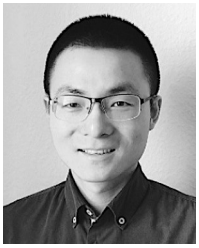
[94] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[95] A. Guzmán-Rivera, D. Batra, and P. Kohli, "Multiple choice learning: Learning to produce multiple structured outputs," in *Proc. of the International Conference on Neural Information Processing Systems*, 2012, pp. 1799–1807.

[96] A. Bhattacharyya, B. Schiele, and M. Fritz, "Accurate and diverse sampling of sequences based on a 'best of many' sample objective," in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8485–8493.

[97] T. Abaigar, J. Barbero, and W. V. Holt, "Trajectory variance and autocorrelations within single-sperm tracks as population-level descriptors of sperm track complexity, predictability, and energy-generating ability," *Journal of Andrology*, vol. 33, no. 2, pp. 216–228, 2012.

[98] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. of the Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.



Jingyuan Wu is currently working in Cross-Domain Computing Solutions at Robert Bosch GmbH. He received his Bachelor of Science degree in mechanical engineering from University of Science and Technology Beijing, China in 2013 and Master of Science degree in mechanical engineering from Technische Universität Kaiserslautern, Germany in 2017. His research interests include signal processing and in particular pedestrian motion prediction.

Johannes Ruenz is currently working in Cross-Domain Computing Solutions at Robert Bosch GmbH. He studied Electrical Engineering and Information Technologies at Karlsruher Institute of Technology (KIT) and received his Master of Science in 2014. His research topic is focused on active pedestrian protection systems.

Hendrik Berkemeyer is currently working in Cross-Domain Computing Solutions at Robert Bosch GmbH. He received his Bachelor of Science degree in Information Systems at University of Münster and his Master of Science degree in Cognitive Science from University of Osnabrück, Germany. His research interests include multi-agent behavior prediction and interaction models for automated driving.

Liza Dixon is a doctoral candidate in Cross-Domain Computing Solutions, Advanced Engineering Projects at Robert Bosch GmbH. She received her Master of Science degree in Usability Engineering at Hochschule Rhein-Waal University of Applied Sciences in 2019. Her research interests include trust in automation, human-machine interaction design in automated driving, and the effects of autonowashing.

Matthias Althoff is an associate professor in computer engineering at Technische Universität München, Germany. He received his diploma engineering degree in mechanical engineering in 2005, and his Ph.D. degree in electrical engineering in 2010, both from Technische Universität München, Germany. From 2010 to 2012 he was a postdoctoral researcher at Carnegie Mellon University, Pittsburgh, USA, and from 2012 to 2013 an assistant professor at Technische Universität Ilmenau, Germany. His research interests include formal verification of continuous and hybrid systems, reachability analysis, planning algorithms, nonlinear control, automated vehicles, and power systems.