Intent-Aware Long-Term Prediction of Pedestrian Motion

Vasiliy Karasev Alper Ayvaci Bernd Heisele Stefano Soatto

Abstract—We present a method to predict long-term motion of pedestrians, modeling their behavior as jump-Markov processes with their goal a hidden variable. Assuming approximately rational behavior, and incorporating environmental constraints and biases, including time-varying ones imposed by traffic lights, we model intent as a policy in a Markov decision process framework. We infer pedestrian state using a Rao-Blackwellized filter, and intent by planning according to a stochastic policy, reflecting individual preferences in aiming at the same goal.

I. INTRODUCTION

Safe operation of autonomous systems co-mingling with humans could benefit from some level of understanding of human behavior by the robot. One of the simplest requirements of safe operation is collision avoidance: The robot must avoid regions of space likely to be concurrently occupied by a human. This requires predicting human motion beyond a short time-horizon. While short-term prediction is sufficiently informed by recent past history, longer time-scale prediction depends on intents or goals, which are not manifest to the viewer. We posit that knowledge of a person’s intents or goals, in conjunction with contextual knowledge can help prediction beyond a few seconds into the future. Since in reality we do not know a person’s goal, we can infer it along with a prediction of human behavior or marginalize it as a nuisance variable. In this paper we explore the problem of long-term prediction of pedestrian behavior in the autonomous (or assisted) driving scenario.

A. Related work and contributions

Common approaches to prediction of time series assume they are samples from a process generated by a linear model (e.g. ARMA) driven by a white, zero-mean, Gaussian input. Traditional tools from filtering and system identification can then be used to infer the state and model parameters [1]. When the order of the model is restricted, it yields predictions that are accurate at short time-scales, but insufficient beyond a few steps, as the processes it represents are stationary. It is also difficult to embed context, such as partial knowledge of the environment, into these models while preserving their structure. Extension to switching models (e.g. PWARX) have been used in constrained scenarios [2], [3], [4] but again their predictive ability has only been demonstrated on short horizons. A nonparametric approach based on Gaussian processes proposed in [5] is accurate on longer time-scales, but requires significant computational efforts. A number of models of “social” or “socially compliant” behavior have been proposed [6], [7], [8], [9], [10], [11]. These models permit prediction of actors’ joint behavior, yet have been verified only on relatively short-term horizons. In this paper we focus on each pedestrian individually, neglecting their interactions with other traffic participants.

A common approach to improve accuracy of long-term prediction has been to postulate “goals” or “destinations”, and to assume that agents navigate toward them by approximately following stochastic shortest paths [12], [13], [14], [15], [16], [17]. This framework allows one to represent rich dynamics, and to naturally incorporate environment constraints – that the agent cannot navigate through “obstacles” [12], [14], [17], and that regions may be more or less preferred depending on their semantic category [16]. During training, this framework requires one to identify the shortest path policy used by the agents – i.e. to solve an inverse reinforcement learning problem [18], [19], [20]. At test time, one uses the agent’s recent measured motion to estimate the goal, whose
knowledge then completely reveals future trajectories. We show that these models of agents’ behavior can be interpreted as switching nonlinear dynamical systems, with the latent goal variable governing the switches, and the policy describing the nonlinear motion dynamics.

In this paper, we provide an interpretation of models in [13], [14], [16] as jump-Markov processes. The large body of work on this topic provides guidance on how estimation/prediction can be approached, while systematically accounting for the uncertainty in dynamics or measurements. We apply the framework to the problem of long-term pedestrian motion prediction. Discrete-space models are applied to a problem in continuous space by adding speed as an additional latent variable. Further, we show how the model can be naturally extended to handle dynamic environments (traffic lights) and additional measurements (pedestrian orientation).

Sec. II describes the framework: Sec. II-A is dedicated to estimation and prediction. Sec. II-B focuses on the underlying optimal control problems. Sec. II-C explains the extension to dynamic environments. Sample results of our framework are shown in Fig. 1 and we further verify the performance of our method in Sec. III.

II. FORMALIZATION

The state of the pedestrian is described by her position, orientation, and speed in the global coordinate frame \((x, \theta, s) \in SE(2) \times \mathbb{R}_+\), and the (unknown) goal \(g \subset \mathbb{R}^2\) (possibly just a point) from a finite set \(G\) which is known a-priori.

We define intent to be a function \(\pi\) mapping goals to actions: given \(g\) and a current state \(x = (x, \theta, s)\), an intent \(\pi\) yields future physical state trajectories from the current time \(t\) to a future time when the goal is achieved. Because different individuals having the same goal will act differently, the map \(\pi\) is stochastic. In the language of Markov decision processes (MDPs), an intent \(\pi\) is called a policy, a sample from it, integrated over time, is called a plan, which at each instant of time corresponds to an action. For simplicity, we assume that the stochastic component of the intent is represented by a process \(w_\pi\), given which the policy \(\pi\) has a known functional form. So, we write \(\pi(x, g, w_\pi)\) to indicate that, given a sample \(w_\pi\) and a goal \(g\), the policy uniquely determines an action.

We assume that the goal \(g\) is slowly time-varying: specifically that at each time instant it switches to another, uniformly chosen goal with a small probability. This can be summarized by a distribution \(p(g_{t+1}|g_t)\), or by the relation \(g_{t+1} = g_t \oplus w_{g,t}\), where \(g\) is interpreted as an integer-valued random process, and the operation is addition modulo \(|G|\). This allows us to write the model as a discrete-time jump-Markov process:

\[
\begin{align*}
    x_{t+1} &= f(x_t, \pi(x_t, g_t, w_{\pi,t})) \\
    g_{t+1} &= g_t \oplus w_{g,t}
\end{align*}
\]

(1)

where the low-level pedestrian dynamics are abstracted into \(f\), the state transition map. This is a model of a stochastic hybrid system with continuous states \(x\), discrete state \(g\), and feedback \(\pi\). As a further simplification, we assume that \(\pi\) is only affected by the positional component \(x\) of \(\mathbf{x}\). This is equivalent to saying that the goal only affects the direction of heading, whereas the speed at which the pedestrian walks is unknown, but otherwise unaffected by the goal. This is clearly a simplification (one may walk at different speeds depending on where s/he is heading), but sufficient for our purpose. We therefore summarize the model as:

\[
\begin{align*}
    x_{t+1} &= x_t + s_t \begin{bmatrix} \cos(\theta_t) \\ \sin(\theta_t) \end{bmatrix} \\
    \theta_{t+1} &= \pi(x_t, g_t, w_{\pi,t}) \\
    s_{t+1} &= s_t + w_{s,t} \\
    g_{t+1} &= g_t \oplus w_{g,t}
\end{align*}
\]

(2)

where \((w_{\pi,t}, w_{s,t}, w_{g,t}) \sim P_{w}\) are the state transition processes, with distribution \(P_{w}\). The pose of our vehicle in the global coordinate frame is described by \((x_{\text{ego},t}, \theta_{\text{ego},t})\) and is assumed to be measured. The observation \(y_t\) consists of the pedestrian position measured relative to the vehicle:

\[
y_t = R(\theta_{\text{ego},t})^T (x_t - x_{\text{ego},t}) + \eta_t
\]

(3)

where \(R\) maps orientation to a rotation matrix and \(\eta_t \sim P_{\eta}\) is the measurement noise.

If we knew the goal state \(g_t\), assuming rational behavior of the agent, we could infer her intent by computing the (future) state trajectory that minimizes expected time-to-goal or, equivalently in our case, path cost. Unfortunately, we do not know the goal state, which we must instead infer using the past state trajectory, which is only known through the history of measurements up to time \(t\), i.e. \(y_1, \ldots, y_t\) or \(y^t\). The goal state could be inferred along with the physical state by estimating (filtering) the posterior \(p(x_t, g_t|y^t)\). A filter maintains an estimate of the filtering density by computing the prediction \(p(x_{t+1}, g_{t+1}|y^t)\) using Chapman-Kolmogorov’s equation and knowledge of \(P_{w,G}\), and updating the prediction once measurements \(y_{t+1}\) become available: \(p(x_{t+1}, g_{t+1}|y^{t+1})\) using Bayes’ rule and knowledge of \(P_{Q}\).

The filtering density can be approximated by a generic particle filter, which maintains a sample-based representation of the posterior. However, in our case we exploit the structure of the model and apply a lower-complexity Rao-Blackwellized filter, as described in Sec. II-A. Once we have the filtering density, we can infer intent by solving the same stochastic optimal control problem that, presumably, a rational agent solves. To do that, we must know the reward function, which we obtain by leveraging the contextual information in the form of a semantic map, as described in Sec. II-B In Sec. II-C we extend the model in (2) to account for pedestrians changing their behavior with respect to changes in the environment. In an assisted driving scenario, an important example of this involves pedestrian behavior being influenced by traffic rules, including time-varying ones enforced by traffic lights. In Sec. II-D we show that the inference is improved via additional

---

1 Throughout the paper we will exploit finiteness of \(G\) to interpret \(g\) either as a region in \(\mathbb{R}^2\), or as an index in \([1, \ldots, |G|]\). The use should be clear from context.
Algorithm 1 Rao-Blackwell particle filter

for $i = 1 \ldots |G|$ do
    $p(x_i|y^{i-1}, g_{t-1} = i) \leftarrow $ predict $\{p(x_{i-1}|y^{i-1}, g_{t-1} = i)\}$
end for

Compute marginal likelihood:

$$p(y_t|y^{t-1}, g_t = i) = \int p(y_t|x_t, g_t = i)p(x_t|y^{t-1}, g_{t-1} = i)dx$$

Predict $g$:

$$p(g_t|y^{t-1}) = \sum_{g_{t-1}} p(g_t|g_{t-1})p(g_{t-1}|y^{t-1})$$

Update posterior over $g$:

$$p(g_t|y^t) \propto p(y_t|y^{t-1}, g_t)p(g_t|y^{t-1})$$

for $i = 1 \ldots |G|$ do
    $p(x_t|y^t, g_t = i) \leftarrow $ update $\{y_t, p(x_t|y^{t-1}, g_t = i)\}$
end for

Algorithm 2 Sampling state trajectory $\hat{x}_t^{t+\tau}, \hat{g}_t^{t+\tau}$

Generate $x_t, g_t \sim p(x_t|y^t, g_t)p(g_t|y^t)$

for $k = 1 \ldots \tau$ do
    $g_{t+k} \sim p(g_{t+k}|g_{t+k-1})$
    $\hat{x}_{t+k} \sim p(x_{t+k}|x_{t+k-1}, g_{t+k-1})$
end for

measurements of the pedestrian’s orientation. Performance of our proposed model is compared to multiple baselines in Sec. III.

A. Filtering and prediction

To infer the state posterior we use a Rao-Blackwellized particle filter (RBPF) [21]. In our case, the posterior is represented by a discrete distribution over $G$, $p(g_t|y^t)$, and by a set of $|G|$ filters (Kalman or particle filters) that approximate each distribution $p(x_t|g_t = i, y^t)$. In Alg. [1] we list the RBPF filtering steps. The “predict” and “update” steps are the standard steps performed by either Kalman or particle filters.

Once we have the posterior $p(g_t, x_t|y^t)$, we can perform prediction. We are interested in the state $\tau$ steps into the future, i.e. $p(x_{t+\tau}, g_{t+\tau}|y^t)$ (or just the position, i.e. $p(x_{t+\tau}|y^t)$). This is a multi-modal distribution and the integration needed to compute it is intractable. However, state trajectories $\hat{x}^{t+\tau}_t, \hat{g}^{t+\tau}_t$ can readily be sampled as described in Alg. [2] and can be used to approximate $p(x_{t+\tau}|y^t)$. As a byproduct it allows us to compute an “occupancy map”, which gives a probability that a region $r \subset \mathbb{R}^2$ is occupied between $t$ and $t + \tau$:

$$P_{occ}(r) = \mathbb{P}(\bigcup_{k=1}^{\tau} \{x_{t+k} \in r\})$$

This quantity enables visualizing future trajectories by discounting time.

B. Goals and environment context

In this section we describe how to model the intent of an agent as a solution to a planning problem. As noted above, the “rational” navigation is abstracted into $\pi$, which decomposes into independent functions $\{\pi_g\}_{g \in G}$. Each is defined by $\pi_g : \mathbb{R}^2 \to S$, with $S$ being the action space. This function specifies the optimal “move direction” for reaching the goal $g$ quickly, while satisfying environment constraints and pedestrian’s contextual preferences. Specifically, it is a solution to the optimal control problem, which can be formulated as a Markov decision process (MDP) [22], as follows:

$$\pi_g^* = \arg \max_{\pi} \sum_{t=0}^{\infty} \gamma^t R_g(x_t, \pi(x_t))$$

subject to $x_{t+1} = x_t + \pi(x_t)$

where the objective is the sum of $\gamma$-discounted rewards $R_g$ for being at location $x$ and applying action $\pi(x)$. In our case, the transition dynamics given by $G$ are deterministic; in the general MDPs they can be stochastic, as we will also explore in Sec. II-C. This sum in the objective is also denoted by $V^\pi_g(x_0)$ — the value of starting at $x_0$ and applying policy $\pi_g$ thereafter. Using the value function, it becomes possible to rewrite the optimization problem recursively as

$$\begin{cases}
V^\pi_g(x) = \max_u Q^\pi_g(x, u) \\
Q^\pi_g(x, u) = R_g(x, u) + \gamma V^\pi_g(x + \pi_g(x)).
\end{cases}$$

Within this formalism, the optimal policy attains the maximum of $Q$ at every $x$, that is, $\pi_g^* = \arg \max_u Q^\pi_g(x, u)$. To fully describe the MDP, we need to specify $R_g$, and to do that we leverage availability of the semantic map. Each point in $\mathbb{R}^2$ is classified in one of several categories (namely, “building”, “sidewalk”, “crosswalk”, “road”, “grass”), and the reward $R_g$ is taken to be a function of the environment category. Specifically, $R_g(x, u) = \theta^T \phi(x)$ where $\theta$ is a vector of preferences and $\phi(x)$ is a vector specifying probabilities of different semantic categories at $x$. The reward is low in regions corresponding to buildings (since those are “obstacles”), high on sidewalks and crosswalks, and is lower on roads. Moreover, for $x \in g$, $R_g(x, u) = 0$ and elsewhere $R_g(x, u) < 0$, which implies that the optimal policy $\pi^*_g$, the solution to (5) describes a shortest path to $g$. We show a region of the world with overlaid semantic categories and associated reward preferences in Fig. 2.

Commonly, one solves an MDP to obtain a deterministic optimal policy. However, for prediction, it is advantageous to use a stochastic, possibly suboptimal policy. This is because whenever shortest paths to a certain $g$ are not unique, nonzero probability should be assigned to each. Moreover, an optimal policy assumes that pedestrians are completely rational and assigns zero probability to any deviations from the optimal trajectory. To allow small deviations from optimality, we use a stochastic Boltzmann policy, also used in [13], [14], [16]:

$$u \sim \pi_g(x) \ \text{w.p.} \ \alpha \exp (\alpha (Q_g(x, u) - V_g(x)))$$

As $\alpha \to \infty$, the policy assigns nonzero probability only to the optimal actions: $u \sim \pi_g(x) \ \text{w.p.} \ \alpha \{u \in \arg \max_u Q_g(x, u)\}$. As $\alpha \to 0$, the policy becomes uniformly random: $u \sim \pi_g(x) \ \text{w.p.} \ \frac{1}{|G|}$. Fig. 2 shows several stochastic shortest paths for different goals, note that the shortest path is not necessarily unique.
The goals are marked with red “∗”. Stochastic shortest paths toward sample goals are shown in the same two areas, overlaid on the satellite image. Pedestrian is marked with “◦” and the destination – with “∗”.

C. Environment dynamics

Accurately predicting pedestrian behavior is particularly important near intersections. The framework outlined above inadequately describes this behavior, as it does not account for traffic signals, the simplest instance of environment dynamics. In this section, we describe a simple extension to the framework that more accurately models pedestrians waiting for the light signal near intersections. To achieve this, we include the state of the traffic signal into the policy \( \pi \).

A traffic signal can itself be represented by a dynamical system. The output is one of four values \( \{0, 1, 2, 3\} \). The four states correspond to the most typical signal configuration with red/green in either direction (2 states) and red/yellow in either direction (2 additional states). Extensions to multi-way signals are possible but not addressed here. If \( T_0, T_1, T_2, T_3 \) are the durations that each output is observed, and \( \sum_{i=0}^{3} T_i = T \), then the state dynamics is a timer: \( c_{t+1} = \text{mod}(c_t + 1, T) \). This model is deterministic and neglects feedback, due for instance to loop sensors detecting proximity of vehicles, including the own-vehicle. Nevertheless, even for this simplified model, directly adding \( c_t \) into \( \pi \) would yield a dramatic increase in state-space dimension, since \( T \) is typically very large, and since the pedestrian must operate on \( c_t \) for this simplified model, directly adding \( c_t \) to the framework that more accurately models pedestrians dynamics. In this section, we describe a simple extension for traffic signals, the simplest instance of environment.

D. Pedestrian orientation

The model described above uses pedestrian motion to estimate the latent goal, and when the pedestrian is initially detected, the distribution over \( g \) is uniform. However, since a pedestrian does not typically walk backwards, we can leverage its orientation to infer which goals are more likely even at the first detection time. We do this by incorporating an additional measurement of the pedestrian orientation. At each time we observe \( y_t = (y_{1,t}, y_{2,t}) \) with \( y_{1,t} = R(\theta_{ego,t})^T (x_{1,t} - x_{ego,t}) + n_t \) (as before) and \( y_{2,t} = \phi(\theta_t - \theta_{ego,t} + n_{2,t}) \), where \( \phi(x) = \text{floor}(\frac{\pi}{180} x) \) models the output of a pose detector quantized to \( N \) orientation values. The uncertainty of the pose detector response is accounted for by \( n_{2,t} \). The additional measurement makes it possible to quickly reduce uncertainty over goals, yielding a more accurate prediction earlier. We show an example of this in Sec. III-D. This improvement is especially important in situations when the pedestrian is tracked only for a small number of frames and an early prediction is required.

III. EXPERIMENTS

We compare against the following baselines:

\[
\text{(RW)} \quad x_{t+1} = x_t + n_t, \quad n_t \sim \mathcal{N}(0, \sigma^2)
\]

\[
\text{(LM)} \quad \begin{cases} x_{t+1} = x_t + v_t \\ v_t = v_t + n_t, \quad n_t \sim \mathcal{N}(0, \sigma^2) \end{cases}
\]

\[
\text{(MM)} \quad x_{t+1} = x_t + u_t, \quad u_t \sim p(u|x_t)
\]

\[
\text{(DM)} \quad x_{t+1} = x_t + \pi(x_t, g, n_t)
\]

“RW” is a simple random-walk. “LM” is a linear “constant-velocity” model commonly used for tracking, and used
in comparisons in [2]. “MM” (“Markov model”) learns a location-dependent probability distribution over actions, but does not reason about “goals”, used in [16]. “DM” (“discrete model”) is described in [13], [14], [16], and is similar to ours, except that it assumes a fixed goal and constant-length steps – it does not adapt to the observations of the pedestrian speed. In our experiments, step lengths for MM and DM were taken to be the average pedestrian speed in the training set.

In sections III-B and III-C we evaluate our “basic” model that does not use the orientation measurement. For this reason, $\theta$ can be omitted from the state in (2), and the model can be simplified to:

$$
\begin{align*}
{x}_{t+1} & = x_t + s_t \frac{\cos(\theta_t)}{\sin(\theta_t)}, \\
{s}_{t+1} & = s_t + w_{s,t}, \\
{g}_{t+1} & = {g}_t + w_{g,t}
\end{align*}
$$

(14)

To perform inference with a RBPF, in this case, we approximated $p({x}_t|y^t, {g}_t = i)_{i=1,\ldots,|G|}$ using a set of Kalman filters. This is possible if one assumes that $\nabla x \pi(x_t, g_t, w_{s,t}) = 0$, and that the constraint $s_t \in \mathbb{R}_+$ can be dealt with by projecting the variable onto the feasible set after each update.

The orientation-aware model is evaluated in Sec. III-E. There, the system dynamics are nonlinear, and to approximate $p({x}_t|y^t, {g}_t = i)_{i=1,\ldots,|G|}$ we used a set of particle filters.

**Error metrics:** The difficulty of prediction depends both on prediction horizon and on the period of observation (since some time is required to infer the pedestrian location and the goal). A natural measure that evaluates the model’s ability is the expected $L_2$ error between the model’s predictions and the actual pedestrian locations (denoted by $\{x_t^{obs}\}_{t=1}^T$), for a given prediction horizon $\tau$ and observed sequence length $t$:

$$
\mathcal{E}(\tau, t) = \mathbb{E}_{p(x_t, y^t)} [\|x_{t+\tau} - x_{t+\tau}^{obs}\|].
$$

This quantity depends on two variables and makes it inconvenient to perform comparisons. A summary error for a given prediction horizon $\tau$ can be computed as the average error over all observation periods, as $\mathcal{E}(\tau) = \frac{1}{T-\tau} \sum_{t=1}^{T-\tau} \mathcal{E}(\tau, t)$.

A. Dataset and implementation details

To verify the effectiveness of our model, we captured a dataset on a vehicle that is equipped with two cameras, LiDAR, and DGPS, each acquiring data at 15Hz. The streams are synchronized with best-effort which produces satisfactory data batches for the long-term prediction task at low vehicle speed. Data from LiDAR and DGPS makes it possible to accurately localize the pedestrians in world coordinates.

Sample data sequences acquired by the vehicle are shown in Fig. 4. For our experiments, we manually annotated 17 video sequences, ranging from 30 to 900 frames, and containing 67 pedestrian trajectories, used in the “fully-observable” experiment. In the “partially-observable” experiment, pedestrians in those videos were automatically detected and tracked; we used a total of 50 of the resulting trajectories (some were lost due to poor detections). In addition to this dataset, we evaluated our approach on a subset of videos from the KITTI benchmark [24].

For learning model parameters, we used 80/20% train/test splits and five-fold cross-validation. Pedestrian detection was performed using a cascade classifier [25] using HoG [26] features. The location of the detections with respect to the vehicle is determined by clustering the LiDAR measurements that are projected into the detection window. Once the relative position of the target object is known, its world-coordinates can be found by addition of the vehicle location given by the DGPS. In the experiment reported in Sec. III-D the traffic light behavior was manually annotated.

**Goals, MDPs, and rewards:** Destination points were placed near the map boundaries and near building entrances (due to lack of a large enough dataset to learn them); each map region contained less than 15 goals. We used standard value iteration to solve MDPs and produce policies $\pi$. Note that MDPs are typically formulated and solved in discrete spaces, but throughout the paper we treated the MDP state-space (and action-space) as being continuous (i.e. $\mathbb{R}^2$ and $S$). Since the state space is fairly low-dimensional, we can solve the discretized MDP and perform interpolation thereafter: at-test time $\pi$ is bilinearly interpolated from the state’s nearest neighbors. We discretize $S$ to 16-25 directions. To specify the reward in MDPs, formally, one should learn the preference vector $\theta$ in $R^g$ using a large enough dataset. We found that the order relations between the cost of each category inferred via IRL [19] matches the obvious ones appointed manually.

**Error computation:** At very short observation periods, the state estimate (pedestrian location, velocity, goal) and the predictions are unreliable, and do not accurately reflect the model performance. For this reason, in computing the prediction errors for all models, we ignored all errors with observation period $t < 10$. The error during this “initialization stage” is shown in the error surfaces $\mathcal{E}(t, \tau)$. In all experiments we used 5000 samples to approximate the predictive $p(x_{t+\tau}|y^t)$ and to calculate prediction errors, for $\tau = 1, \ldots, 350$ (about 23 seconds).
B. Prediction in the perfectly observed case

We first evaluated our framework on manually annotated “ground-truth” trajectories. In this case, the measurement noise $n_t = 0$, and observations fully disclose pedestrian locations. Quantitative performance results are summarized in Fig. 5, where we show prediction errors for up to 350 steps. Each time step corresponds to 67 ms, i.e. the camera frame rate is 15 Hz. As is well known, MM degenerates to random walk and is often unable to generate meaningful predictions, similarly to RW. LM accurately predicts short-term behavior but cannot predict inevitable direction changes. These baselines do not utilize environment semantics, which strongly affect behavior of traffic participants. DM generates meaningful predictions, but ones that are not temporally accurate, since the model does not estimate pedestrian speed. In particular, the predictions are erroneous when the pedestrian is stationary (e.g. waiting for the signal at the intersection) and the model predicts motion. Our model utilizes environment semantics, estimates the pedestrian speed, and performs best.

C. Prediction in the imperfectly observed case

We also tested the framework in the imperfectly observed setting (with $n_t \sim P_Q$), where observations are obtained from noisy detector responses. These results are qualitatively similar and are shown in Fig. 5 (bottom). Qualitative examples of predictions generated by our model and by the baseline are shown in Fig. 6. In that figure DM is not shown, as it generates an occupancy map nearly identical to ours. Unfortunately, its error is high due to predicting motion at the incorrect speed.

D. Intent near intersections

We compared the basic framework with the traffic-aware extension described in Sec. II-C. The extended framework is expected to reduce the probability that a pedestrian violates traffic rules (e.g. the probability that she walks on the crosswalk on “red”). An appropriate measure that illustrates the difference between the two models is the probability that the pedestrian is on the road at $t + \tau$, given the observations $y_t$, i.e. $P_{\text{road}}(\tau, t) = \mathbb{E}_{p(x_{t+\tau}|y_t)}[1\{x_{t+\tau} \in \text{road}\}]$. In Fig. 7 we show that the extended model correctly predicts that pedestrian does not walk out on the road (doing so would violate traffic rules), and improves the overall prediction error.

E. Pedestrian orientation

To leverage orientation, we trained a linear classifier with HoG features that discretized the orientation $\theta$ to four values (“left”, “front”, “right”, “back”), and learned a response likelihood $p(y_2|\theta)$. The improvement due to the additional measurement occurs primarily during the first few detection
traffic lights, and changes in the environment, e.g. parked cars. These could be incorporated at a significant increase in computational cost that would prevent real-time operation. We also assume rational agent behavior, necessary to even define a predictive strategy, which means that we cannot predict rare events such as a pedestrian suddenly jumping onto the road. This limitation is shared with the majority of existing methods.

**REFERENCES**


