CASNSC: A context-based approach for accurate pedestrian motion prediction at intersections

Anonymous Author(s)
Affiliation
Address
email

Abstract

Intention recognition of pedestrians is crucial to safe and reliable working of autonomous vehicles, when serving as, for instance, indoor service robots or self-driving cars in busy urban scenes. Previously, Chen et al. [2016] combined Markovian-based and clustering-based approaches to learn motion primitives and subsequently predict pedestrian trajectories by modeling the transition between learned primitives as a Gaussian Process (GP). This work further develops their approach by incorporating semantic features from the environment (relative distance to curbside and status of pedestrian traffic lights) for more confident prediction of pedestrian trajectories at intersections. Adding the environmental context, when available, not only makes prediction more robust but can also provide increased flexibility of prediction in new environments. We test our algorithm on real data. The results show 26% improvement in prediction *accuracy* as compared to previous work, on incorporation of new features.

1 Introduction

Recent advances in sensor technologies and computing power have led to a surge in research on autonomous driving to improve road safety (Fagnant and Kockelman [2015], Bagloee et al. [2016]), reduce traffic congestion and improve vehicle utilization. For safe and efficient autonomous driving in complex urban environments, a self-driving vehicle must be able to interact with other moving objects, including pedestrians, cyclists, and, of course, cars. Pedestrian trajectory prediction is challenging as compared to that of other cars and cyclists because of the absence of a regular flow, such as driving within lanes and staying within road boundaries, that result from a fairly uniform set of predefined "rules of the road" for cars (and to some extent cyclists). The complexity is increased further when the urban environment includes pedestrian traffic lights or tightly packed sidewalks with numerous pedestrian interactions.

Several papers have been written on short-term prediction of human motion (Kooij et al. [2014], Bissacco and Soatto [2009]), but understanding goals or intent is needed to plan for longer timescales (Karasev et al. [2016], Alahi et al. [2016]). Previous work has focused on two main approaches (Lefèvre et al. [2014]) to modeling maneuvers of dynamic agents, including pedestrians: 1) prototype trajectories-based and 2) maneuver intention estimation-based. In general, prototype trajectories-based/clustering-based approaches are more robust to measurement noise when compared to maneuver intention estimation-based approaches, which are mostly Markovian (Makris and Ellis [2002], Vasquez et al. [2009]) and rely on the current state only for prediction. However, the prototype trajectories-based approaches can be computationally quite expensive (Rasmussen and Ghahramani [2002], Ferguson et al. [2015]) and hence slow in detecting changes in pedestrian intent. They are also susceptible to issues like partial trajectories in the training dataset being grouped into a cluster and learned as a trajectory prototype.

Chen et al. [2016] use a combination of prototype trajectory-based and Markovian-based methods to inherit the benefits of both techniques in developing a dictionary learning algorithm, called augmented semi nonnegative sparse coding (ASNSC). Learning motion primitives instead of complete prototype trajectories addresses partial observability of trajectories caused by occlusions or a limited field of view of on-board perception sensors. ASNSC creates a set of feasible trajectories as its prediction that are learned based on solely the spatial features of the training dataset (absolute x and y position and orientation of pedestrians), independent of the environment context that may influence a pedestrian's intent.

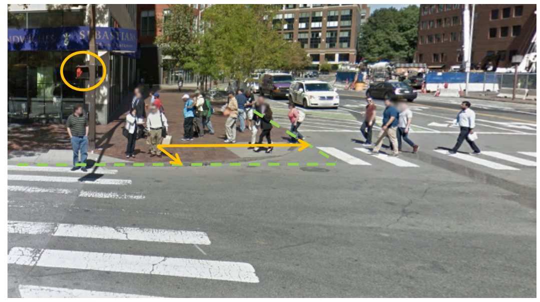


Figure 1: Example intersection scenario. Dotted green line denotes a rectangular approximation to the curbside in view. Orange arrows denote relative distance of a pedestrian from the two curbsides, which can indicate pedestrian intention. Pedestrian traffic light status is highlighted in orange, which influences pedestrian movement.

The accuracy of these predictions could be improved by adding semantic features from the environment in the learning process. Incorporating the environmental context can also provide the flexibility of application of the learned model to prediction in new, but similar environments, unexplored earlier, which is in general difficult to achieve with clustering-based approaches (Lefèvre et al. [2014]). Fig. 1 shows an intersection scenario in which, when faced by a choice between two crosswalks, pedestrian traffic light status for each of those crosswalks influences pedestrian choice. Similarly, a comparison of the relative distance to each curbside could be indicative of future direction of motion. Most of the previous work on context-based pedestrian trajectory prediction is limited to a classification problem (Schulz and Stiefelhagen [2015]). This work, in contrast, provides a continuous trajectory as its prediction output. Karasev et al. [2016] used jump-Markov process for long term prediction of pedestrian motion by incorporating traffic light and crosswalks as semantic features. The output of their prediction model is an *occupancy map* of feasible trajectory predictions. Our goal is to make prediction confident and output the *most likely trajectory* with increased accuracy.

Our approach extends ASNSC by incorporating semantic features from the environment. In order to meet these objectives, a dictionary of motion primitives is learned as in Chen et al. [2016]. However, the transition between these motion primitives is learned with respect to both spatial as well as additional environmental contexts. As illustrated in Fig. 2(c), the influence of pedestrian traffic light status on the probability of transition between two motion primitives is not captured in ASNSC. Two main features are used to incorporate the environmental context in this work: pedestrian traffic light status and relative distance to curbside. Similar to the approach followed by Chen et al. [2016], GP models are used to learn the transition between motion primitives and subsequently predict pedestrian velocity. A squared exponential (SE) kernel function with automatic relevance determination (ARD) (Rasmussen and Williams [2006]) is used to determine the relevance of each of the individual features. The results show a 26% increase in the *accuracy* of pedestrian trajectory prediction.

2 Augmented semi nonnegative sparse coding

Given a training dataset of n samples, $\mathbf{Z} = [\mathbf{x_1}, \ldots, \mathbf{x}_n]$, where $\mathbf{x_i}$ is a column vector of length p, the objective is to learn a set of K dictionary atoms, $\mathbf{D} = [\mathbf{d_1}, \ldots, \mathbf{d}_K]$, and the corresponding nonnegative sparse coefficients, $\mathbf{S} = [\mathbf{s_1}, \ldots, \mathbf{s}_n]$. Mathematically, this can be formulated as a constrained optimization problem of the form (Chen et al. [2016])

$$\underset{\mathbf{D},\mathbf{S}}{\operatorname{argmin}} ||\mathbf{Z} - \mathbf{DS}||_F^2 + \lambda \sum_{i=1}^{n} ||\mathbf{s_i}||_1 \tag{1}$$

$$\text{s.t. } \mathbf{d_k} \in \mathbf{Q},\ s_{ki} \geq 0 \ \forall \ \text{k,i} \tag{2}$$

where λ is a regularization parameter and $\mathbf{Q}$ is the feasible set in which $\mathbf{d_k}$ resides. Fig. 2(b) shows an example of dictionary atoms learned using ASNSC. $\mathbf{D}$ is used to segment the original training trajectories $\mathbf{x_i}$ into clusters, where each cluster is best explained by one of the learned dictionary atoms.

2.1 Trajectory prediction using learned dictionary

A transition matrix, $\mathbf{T} \in \mathbb{Z}^{K \times K}$ is thus created, where $T(i,j)$ denotes the number of trajectories exhibiting a transition from *i*-th dictionary atom to *j*-th dictionary atom. A transition will, therefore, be mathematically represented as a concatenation of two dictionary atoms $\{\mathbf{d_i}, \mathbf{d_j} | T(i,j) > 0\}$. Each transition is modeled as a two-dimensional GP flow field (Joseph et al. [2011], Aoude et al. [2013]). In particular, two independent GPs, (GP_x, GP_y), called GP motion patterns are used to learn a mapping from chosen features $\mathbf{X} \in \mathbb{R}^N$ to the x-y velocities. ASNSC uses $\mathbf{X} = \mathbf{X_p} = (x,y)^\top \in \mathbb{R}^2$ as the feature vector.

$$GP_x : \mathbf{X} \rightarrow v_x, \quad GP_y : \mathbf{X} \rightarrow v_y \tag{3}$$

The learned GP motion patterns, (GP_x, GP_y), are used for generating a predicted path using (3) as well as for computing the likelihood of an observed trajectory, $\mathbf{t}' = \{(\mathbf{X_1}', \mathbf{v_1}'), \ldots, (\mathbf{X_l}', \mathbf{v_l}')\}$ using

$$P(\mathbf{t}'|GP_x, GP_y) = \prod_{\mathbf{X}' \in \mathbf{t}'} \mathcal{N}(v_x; \mu_{GP_x}(\mathbf{X}'), \sigma^2_{GP_x}(\mathbf{X}')) \mathcal{N}(v_y; \mu_{GP_y}(\mathbf{X}'), \sigma^2_{GP_y}(\mathbf{X}')) \tag{4}$$

Trajectory prediction has two main steps. 1) Unitary GP motion patterns (GP_x^{uni}, GP_y^{uni}) are learned from training trajectories corresponding to $\mathbf{T}(i,j)\ \forall\ i = j$. The unitary GP motion pattern that most likely generated the observed trajectory $\mathbf{t}'$ is determined using (4), which is equivalent to selecting the most likely initial dictionary atom $\mathbf{d}_{\hat{k}}$ (Algorithm 1, line 12). 2) The set of possible future dictionary atoms can be found as $\mathscr{D} = \{j | \mathbf{T}_{\hat{k}j} > 0\}$ (Algorithm 1, line 13). Transitional GP motion patterns, $(GP_{x_{\hat{k}j}}^{tran}, GP_{y_{\hat{k}j}}^{tran})\ \forall j \in \mathscr{D}$ are used for generating a set of predicted trajectories $\{\mathbf{s}_j\}$.

3 Context-based augmented semi nonnegative sparse coding

This work develops ASNSC by incorporating semantic features from the environment in the prediction phase (Algorithm 1, lines 5-14) and is motivated by situations in which the environmental context influences transition between learned dictionary atoms (see Fig. 2(c)). The proposed approach uses two sets of features: 1) *learning features*, $\mathbf{X_p}$, which are used for learning $\mathbf{D}$ (Algorithm 1, lines 1-4); and 2) *prediction features*, $\mathbf{X}$, which are used for prediction using GP models (lines 5-14). ASNSC uses the same set of features, $\mathbf{X_p}$, for both learning and prediction. In contrast, CASNSC uses the same set of *learning features*, $\mathbf{X_p}$, as used in ASNSC, but an augmented set of features, $\mathbf{X} = (\mathbf{X_p} : \mathbf{X_c})$, as *prediction features*. Here, $\mathbf{X_c}$ denotes the set of additional *context features*.

$$GP_x : \mathbf{X} \rightarrow v_x, \quad GP_y : \mathbf{X} \rightarrow v_y \tag{5}$$

$$\text{where } \mathbf{X} \in \mathbb{R}^N \text{ s.t. } \mathbf{X} = (\mathbf{X_p} : \mathbf{X_c}),\ \mathbf{X_p} \in \mathbb{R}^2\ \mathbf{X_c} \in \mathbb{R}^{N-2} \tag{6}$$

3.1 Context features

The relative distance of a pedestrian from the left/right curbside and pedestrian traffic lights' status have been used as additional *context features* in this work. However, the described framework is generalizable to any number and type of feature selection.

3.1.1 Distance to curbside

The relative distance of a pedestrian (treated as a point mass) to curbside can be computed using a map of the environment. When approaching an intersection, pedestrians are in the vicinity of two different curbsides, assumed to intersect at a point (see Fig. 2(a)). A two-dimensional vector, $(c_l, c_r)^\top$, is therefore used as the curbside feature. This particular feature influences pedestrian intention only when the observed trajectory starts on the sidewalk. As explained in Fig. 2(a), this aspect is captured by assigning a positive or negative sign to the distance computed when constructing the feature vector.

3.1.2 Pedestrian traffic light

A pedestrian's decision to go left or right is influenced by the status of two pedestrian traffic lights (T1, T2) in a four-way intersection scenario. However, in contrast to the curbside feature, a single-dimensional feature vector, (tr), is sufficient to capture the environment context with respect to both the traffic lights as the change in status of (T1, T2) captures redundant information.

Algorithm 1: CASNSC
(Context-based augmented semi nonnegative sparse coding)

/* Learning Phase */

$\mathbf{D} \leftarrow \mathbf{0}, \mathbf{S} \leftarrow \mathbf{0}$;

while *not converged* **do**

  $\{\mathbf{D},\mathbf{S}\} = ASNSC(\mathbf{Z},\mathbf{X_p},\lambda)$

$\mathbf{T} \leftarrow$ Transition_Matrix$(\mathbf{D},\mathbf{Z},\mathbf{S})$;

/* Prediction Phase */

$\mathbf{X} = (\mathbf{X_p} : \mathbf{X_c})$;

$GP^{\,uni} \leftarrow \emptyset, GP^{\,tran} \leftarrow \emptyset$;

for $\forall\ (i,j)$ *s.t.* $\{\mathbf{T}(i,j) > 0\}$ **do**

  if $i == j$ **then**

    $GP^{\,uni}.insert((GP_x(\mathbf{X}), GP_y(\mathbf{X})))$

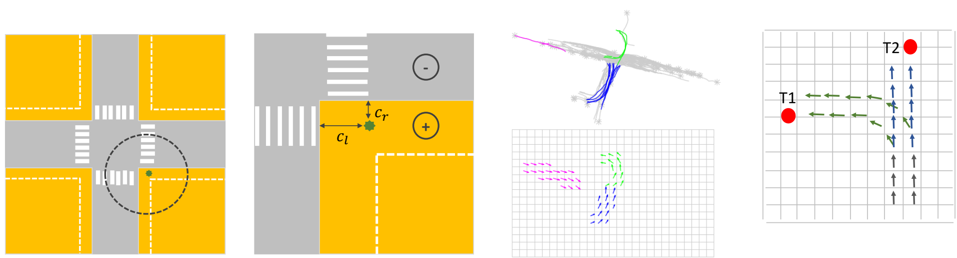
  else

    $GP^{\,tran}.insert((GP_x(\mathbf{X}), GP_y(\mathbf{X})))$

$\hat{k} = \text{argmax}_k\ P(\mathbf{t}'|GP_{x_k}^{\,uni}, GP_{y_k}^{\,uni})$

for $\forall\ j\ \in \mathscr{D} = \{j|\mathbf{T}_{\hat{k}j} > 0\}$ **do**

  $\mathbf{s}_j \leftarrow$ Predict$(\mathbf{t}', (GP_{x_{\hat{k}j}}^{\,tran}, GP_{y_{\hat{k}j}}^{\,tran}))$



(a) (b) (c)

Figure 2: (a) A typical four-way intersection used to explain the curbside feature. The zoomed portion (right) shows a pedestrian location as a green star. c_l, c_r denote the distance to the two curbsides of interest. If the green star is in the orange region (curbside), a positive value is assigned to the curbside feature. Otherwise, a negative value is assigned. (b) From Chen et al. [2016]: examples of dictionary atoms learned using ASNSC; each color represents one dictionary atom (below). Training trajectory segments that agree with each of the dictionary atoms are also shown (top). (c) Pedestrian traffic light status influences transition between dictionary atoms. T1 and T2 denote two different traffic lights. Transition between dictionary atoms represented by black and blue has a higher probability than that between black and green for T1 = 0, T2 = 1.

3.2 Kernel function

A SE kernel function with ARD is used as it allows for the combination of features with different characteristics and scales each feature in accordance with its relevance. Mathematically, it is given by the following form (Rasmussen and Williams [2006]):

$$k(\mathbf{X},\mathbf{X}') = \sigma_f^2 \exp(-\frac{1}{2}(\mathbf{X}-\mathbf{X}')\Lambda(\mathbf{X}-\mathbf{X}')) \tag{7}$$

$$\text{where } \theta = (\{\Lambda\}, \sigma_f^2)^\top \text{ and } \Lambda = \text{ diag } (\mathbf{l}^{-2}) \tag{8}$$

Here, θ is a vector containing all hyperparameters and $\mathbf{l}$ is a vector of positive values. Thus, for an N-dimensional feature vector, the number of hyperparameters needed to define the SE kernel function with ARD is $(\mathrm{N}+1)$. The $l_1,\ldots,l_\mathrm{N}$ hyper parameters represent characteristic lengths of the individual features and aid in determining the relevance of each feature in the N-dimensional feature space. In this work, the *GPML Toolbox* has been used for learning hyperparameters.

In particular, for predictions using CASNSC with pedestrian traffic light status as an additional contextual feature, $\mathbf{X} = (x, y, tr)^\top$ in (7) and $\theta = (l_x, l_y, l_{tr}, \sigma_f)^\top$ in (8). For distance to curbside as an additional *context feature*, in order to account for its dependency on pedestrian position, a linear combination of two SE with ARD kernel functions is used as follows:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_{f_1}^2 \exp(-\frac{1}{2}(\mathbf{X_p} - \mathbf{X_p}')\Lambda_p(\mathbf{X_p} - \mathbf{X_p}')) + \sigma_{f_2}^2 \exp(-\frac{1}{2}(\mathbf{X_c} - \mathbf{X_c}')\Lambda_c(\mathbf{X_c} - \mathbf{X_c}')) \tag{9}$$

$$\text{where } \theta = (l_x, l_y, l_{c_l}, l_{c_r}, \sigma_{f_1}, \sigma_{f_2})^\top \text{ and } \mathbf{X_c} = (c_l, c_r)^\top \tag{10}$$

4 Results

Our approach is tested on real pedestrian data collected by a Polaris GEM vehicle equipped with three Logitech C920 camera and a SICK LMS151 LIDAR (Miller and How [2017], Miller et al. [2016]). The dataset consists of 186 training trajectories and 32 test trajectories. An observation history of 2.5 seconds prior to the pedestrian entering the intersection is used to predict 5 seconds ahead in time. Fig. 3 provides a qualitative comparison of our approach with Chen et al. [2016] on the inclusion of distance to curbside as an additional feature. While ASNSC provides all feasible pedestrian trajectories, given the intersection geometry, CASNSC only picks those that are closest to the actual trajectory. Similarly, Fig. 5 compares prediction results in three different scenarios, on the inclusion of pedestrian traffic light status as an additional feature.

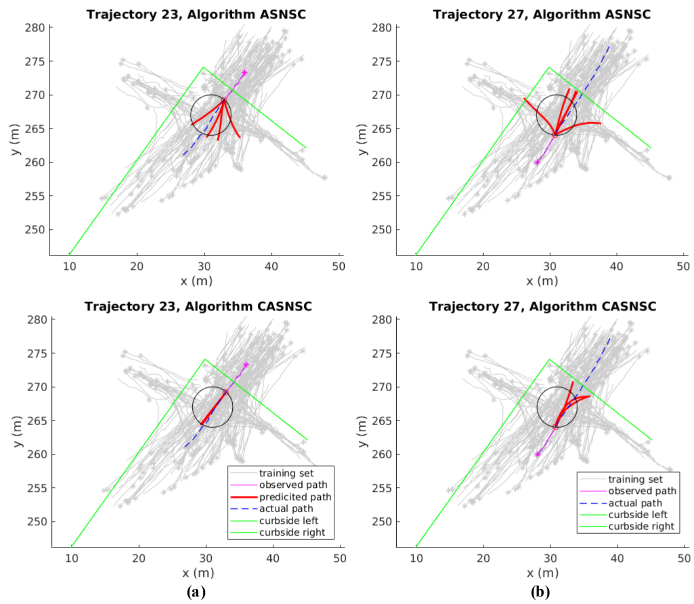


Figure 3: ASNSC vs. CASNSC on incorporation of relative distance to curbside as a *context feature*. Training trajectories are shown in gray and prediction is performed for test trajectories in the black circle.

In the first scenario (trajectory 18), pedestrian traffic lights' status is given by T1 = 0, T2 = 1 and the pedestrian has already crossed the intersection and is entering the sidewalk. While ASNSC predicts a set of two trajectories, CASNSC provides a more confident prediction. In the second scenario (trajectory 25), the traffic lights' status is the same but the pedestrian is now entering the intersection and is faced with a choice between two crosswalks. CASNSC picks the correct direction out of the set of feasible directions (as predicted by ASNSC) taking the pedestrian traffic light status into account. In the last scenario (trajectory 11), the traffic lights' status is given by T1 = 1, T2 = 0. While both ASNSC and CASNSC pick the correct direction, predictions using the latter follow the actual trajectory more closely since it can utilize the information that T2 = 0 to predict that the pedestrian will continue moving straight, with little or no probability of turning left.

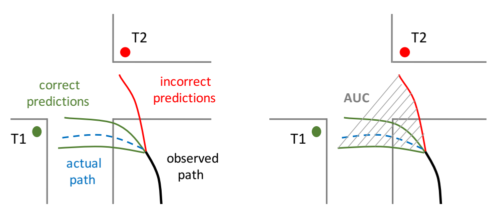


Figure 4: (Left) *Incorrect* and *correct* predictions at an intersection scenario. (Right) Use of AUC as a metric for variance in prediction.

Fig. 4 illustrates the metrics used for performance evaluation and Table 1 provides a quantitative comparison of ASNSC with CASNSC on the inclusion of pedestrian traffic light status as an additional *context feature*. As illustrated in Fig. 4, the *Area Under the Curve (AUC)* (Hand [2009]) is used as a metric for comparing the confidence level of predictions, such that a larger *AUC* corresponds to a lower confidence.

Table 1 indicates that *AUC* for predictions using CASNSC is lower than when using ASNSC, confirming that CASNSC provides a more confident prediction. Classification *accuracy* is also

Table 1: Performance evaluation comparison of CASNSC with ASNSC

Algorithm	Classification accuracy(%)	MHD(m)	AUC(m^2)	Computation time(s)
ASNSC	73.95	2.12	75.48	**0.14**
CASNSC	**100**	**1.79**	**30.51**	0.3

measured, which represents the fraction of *correct* predictions. For an correct representation of this metric, the likelihood of prediction of each trajectory is taken into account when computing the *accuracy*. For instance, if a set of n trajectories is predicted, $\{\mathbf{t}_1, \ldots, \mathbf{t}_n\}$, with their likelihood of prediction given by $\{l_1, \ldots, l_n\}$, and the *correct* predictions are identified as $\{\mathbf{t}_i\}\ \forall\ i\ \in \mathbf{C} \subset \{1, \ldots, n\}$, the classification *accuracy* is given by:

$$\text{Classification accuracy } \% = \frac{\sum_{i \in \mathbf{C}} l_i}{\sum_{k=1}^{n} l_k} \times 100\%. \tag{11}$$

In addition to the illustrated metrics, the *modified Hausdorff distance (MHD)* (Dubuisson and Jain [1994]) is used to compare predicted pedestrian trajectories with the ground truth. Note that MHD is used to compare the *correct* predictions only. Table 1 shows an improvement in all the chosen metrics, with only a slight increase in computation time.

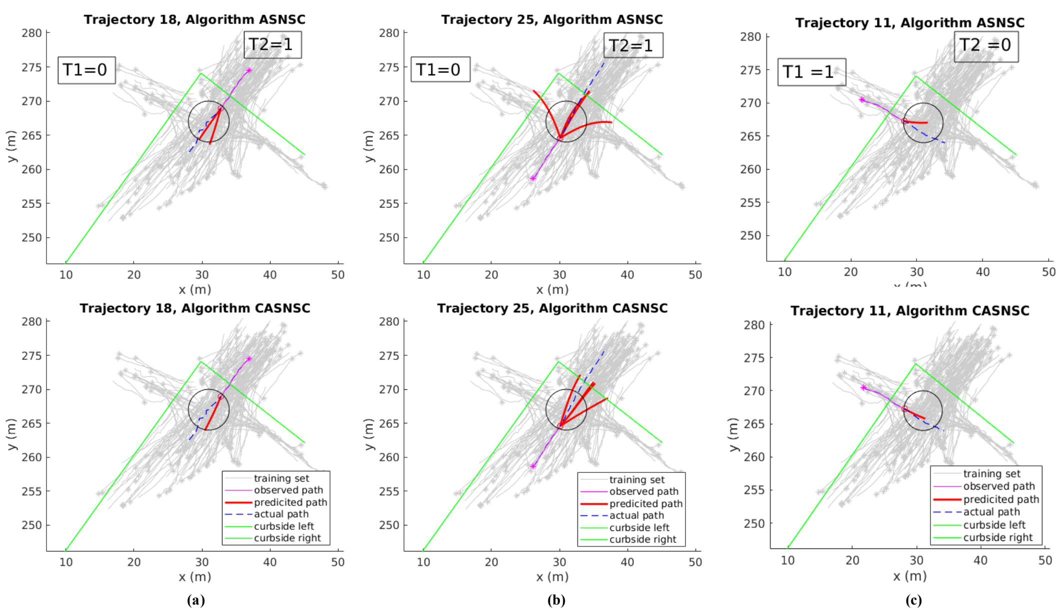


Figure 5: Comparison of prediction performance of ASNSC with CASNSC on addition of pedestrian traffic light status as a *context feature*.

5 Conclusion

We extend ASNSC by incorporating relative distance to curbside and pedestrian traffic light status as additional *context features* for more confident and accurate prediction. Our approach, CASNSC, shows a 26% improvement in *accuracy*, 15.5% improvement in *MHD* of *correct* predictions and reduces variance in prediction, as measured by *AUC*, by a factor of 2.5. There is scope for further improvement on incorporation of features (specific to intersections) that are constant in time (e.g., the existence of crosswalks and areas of interest like subway stations). Furthermore, testing the learned prediction model on new but similar, four-way intersections and incorporating interactions between pedestrians would provide good insight into the flexibility and robustness of this approach respectively.

References

Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–971, 2016.

Georges S Aoude, Brandon D Luders, Joshua M Joseph, Nicholas Roy, and Jonathan P How. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns. *Autonomous Robots*, 35(1):51–76, 2013.

Saeed Asadi Bagloee, Madjid Tavana, Mohsen Asadi, and Tracey Oliver. Autonomous vehicles: challenges, opportunities, and future implications for transportation policies. *Journal of Modern Transportation*, 24(4):284–303, Dec 2016. ISSN 2196-0577. doi: 10.1007/s40534-016-0117-3. URL `https://doi.org/10.1007/s40534-016-0117-3`.

Alessandro Bissacco and Stefano Soatto. Hybrid dynamical models of human motion for the recognition of human gaits. *International journal of computer vision*, 85(1):101–114, 2009.

Yu Fan Chen, Miao Liu, and Jonathan P How. Augmented dictionary learning for motion prediction. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 2527–2534. IEEE, 2016.

M-P Dubuisson and Anil K Jain. A modified hausdorff distance for object matching. In *Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, volume 1, pages 566–568. IEEE, 1994.

Daniel J Fagnant and Kara Kockelman. Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations. *Transportation Research Part A: Policy and Practice*, 77: 167–181, 2015.

Sarah Ferguson, Brandon Luders, Robert C Grande, and Jonathan P How. Real-time predictive modeling and robust avoidance of pedestrians with uncertain, changing intentions. In *Algorithmic Foundations of Robotics XI*, pages 161–177. Springer, 2015.

David J Hand. Measuring classifier performance: a coherent alternative to the area under the roc curve. *Machine learning*, 77(1):103–123, 2009.

Joshua Joseph, Finale Doshi-Velez, Albert S Huang, and Nicholas Roy. A bayesian nonparametric approach to modeling motion patterns. *Autonomous Robots*, 31(4):383, 2011.

Vasiliy Karasev, Alper Ayvaci, Bernd Heisele, and Stefano Soatto. Intent-aware long-term prediction of pedestrian motion. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 2543–2549. IEEE, 2016.

Julian Francisco Pieter Kooij, Nicolas Schneider, Fabian Flohr, and Dariu M Gavrila. Context-based pedestrian path prediction. In *European Conference on Computer Vision*, pages 618–633. Springer, 2014.

Stéphanie Lefèvre, Dizan Vasquez, and Christian Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *Robomech Journal*, 1(1):1, 2014.

Dimitrios Makris and Tim Ellis. Spatial and probabilistic modelling of pedestrian behaviour. In *British Machine Vision Conference 2002, vol. 2*. Citeseer, 2002.

Justin Miller and Jonathan P How. Predictive positioning and quality of service ridesharing for campus mobility on demand systems. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 1402–1408. IEEE, 2017.

Justin Miller, Andres Hasfura, Shih-Yuan Liu, and Jonathan P How. Dynamic arrival rate estimation for campus mobility on demand network graphs. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 2285–2292. IEEE, 2016.

Carl E Rasmussen and Zoubin Ghahramani. Infinite mixtures of gaussian process experts. In *Advances in neural information processing systems*, pages 881–888, 2002.

Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, 2006.

Andreas Th Schulz and Rainer Stiefelhagen. Pedestrian intention recognition using latent-dynamic conditional random fields. In *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pages 622–627. IEEE, 2015.

Dizan Vasquez, Thierry Fraichard, and Christian Laugier. Incremental learning of statistical motion patterns with growing hidden markov models. *IEEE Transactions on Intelligent Transportation Systems*, 10(3):403–416, 2009.