SIMP3: Social Interaction-Based Multi-Pedestrian Path Prediction By Self-Driving Cars

Nora Muscholl, Atanas Poibrenski, Matthias Klusch, Patrick Gebhard
German Research Center for Artificial Intelligence (DFKI)
Saarland Informatics Campus, Stuhlsatzenhausweg 3, Saarbruecken, Germany
firstname.lastname@dfki.de

***Abstract*—An accurate and fast prediction of future positions of pedestrians by a self-driving car in critical traffic scenarios remains a challenge. The intention of a pedestrian to cross the street can be influenced by social interactions with another one across the street, which may be manifested through various types of social signals such as hand waving. Current socially-aware multi-pedestrian path predictors mainly rely on geometric heuristics such as the distance between pedestrians in the field of view of the car, but do not consider their social interaction across the street. This paper presents a novel social interaction-based multi-pedestrian path predictor (SIMP3) which leverages a combination of dynamic Bayesian networks for intention detection and recurrent network for prediction of future pedestrian locations. The system has been evaluated on the benchmark OpenDS-CTS2 of critical traffic scenarios with socially interacting pedestrians across the street simulated in OpenDS. Our experiments revealed that in most scenarios SIMP3 can significantly outperform the selected competitors.**

***Index Terms*—autonomous cars, pedestrian path prediction, social interaction**

I. Introduction

One major challenge in the research area of self-driving vehicles is to accomplish a highly accurate path prediction in critical scenarios of pedestrians crossing the street [25]. Pedestrians usually navigate in a shared environment and could abruptly change their goals and paths according to observed behaviour and movement of others on the same or even on the opposite sidewalk of the street. Appropriate reasoning on observed social interactions between pedestrians across the street may help to detect their intention to step on the street, which in turn can be decisive for achieving a more reliable prediction of their future locations in time in order to avoid collisions with them.

Current approaches to multi-pedestrian path prediction (MP3) make use of various features of pedestrian dynamics such as pedestrian position, moving direction, velocity, scene context such as distance to curb, traffic light state, crosswalks, and social context such as distance to others in order to learn to estimate the future path of pedestrians. The vast majority of MP3s relies on the past trajectory of pedestrians as the main input feature, which basically renders them only effective if a pedestrian is already about to cross the street. Recent approaches to socially-aware multi-pedestrian path prediction such as Social-LSTM [2], DESIRE [15], Social-GAN [9], STGAT [10], NEXT [16], and Social-STGCNN [1] do not consider social signals of interactions between pedestrians on opposite sides of the street but mainly rely on interpersonal distances as social context feature to predict their behavior [25]. On the other hand, an experienced human driver is often able to quite early on correctly predict the intention of pedestrians in his field of view to step on the street in front of the car only based on their prior social interaction across the street. This ability to reason on social signals can be life saving in particular when the street crossing of a pedestrian to meet one or multiple others on the opposite sidewalk occurs far too abrupt such that current path predictors fail to sufficiently fast and correctly adjust their prediction.

To this end, we developed the first social interaction-based multi-pedestrian path predictor (SIMP3) that utilizes a set of dynamic Bayesian networks (DBN) for pairwise, probabilistic detection of pedestrian intention to meet each other in support of an encoder-decoder recurrent neural network that eventually predicts the future locations of all observed pedestrians. Each DBN encodes a cognitive causal model of dynamic and dyadic social interaction-based pedestrian intention based on social signal processing [3], [7], [29]. The comparative experimental performance evaluation has been conducted with our initial benchmark OpenDS-CTS2 including thousands of critical street-crossing scenarios simulated in the open-source driving simulator OpenDS6. All sources of SIMP3 and the benchmark are publicly available [link omitted:double-blind].

The remainder of the paper is structured as follows. In Section 2 we summarize related work and in Section 3 we formulate the given problem. Section 4 describes our SIMP3 solution followed by comparative experimental evaluation in Section 5 before we conclude in Section 6.

II. Related Work

A large body of literature focuses on socially unaware path predictors, considering only past motion of agents and contextual cues such as the surrounding environment. In particular, early physics-based works in this regard base on constant acceleration models, linear velocity projection and Markov Models [5], [17], [28], [35]. Some more recent pattern-based works use recurrent neural networks [11], [30] as well as convolutional neural network [19] to predict individual motions.

This work has been funded by the German Ministry for Research and Education (BMB+F) in the project REACT.

Other works use planning-based approaches [12], [22], [24]. Having no explicitly (or implicitly) defined agent interaction model can in general result into a model with less parameters which can generalize better, but will fail in real-world ad-hoc pedestrian motion changes due to social interactions with other pedestrians.

In general, socially-aware path predictors take the interactions between agents into account by learning or modeling the influence between each other. Many approaches rely on the social force model, a physics-based model that predicts a collision free agent trajectory using predefined rules [4], [6], [27], [33]. Other classical solutions in this field model the interactions between agents locally in a similar manner [20], [21], [23], [31]. Another recent strain of research proposes deep learning-based solutions for socially-aware multi-pedestrian path prediction. For example, Social-LSTM [2] uses a recurrent neural network (LSTM) together with a social pooling layer to predict future trajectories. The state-refinement LSTM (SR-LSTM) [34] improves the pooling mechanism by a weighting mechanism, and the Social-GAN [9] extends the Social-LSTM [2] into a generative recurrent neural network model. Social-BiGAT [14] uses graph attention to model the social interaction between people combined with an attention mechanism to weight the contribution of the recurrent states of each pedestrian. Social-STGCNN [1] is another very recent work, which uses social spatio-temporal graph convolutional neural network to model the relations between people. Sophie [26] uses a CNN to extract features from the environment together with a two way attention mechanism for each pedestrian. Similar to Sophie [26], DESIRE [15] uses CNN for the scene features but uses LSTM encoder-decoder architecture combined with a conditional variational autoencoder for the path prediction. Finally, the recent LSTM encoder-decoder NEXT [16] uses geometric person-person relations in order to model the social interactions.

However, all of these works basically rely either on an implicit pooling mechanism (with attention) to model the social interactions, or use proximity based features such as an explicit pedestrian map [32] or geometric inter-pedestrian distances. We believe that these models will perform poorly (see section V) when a pedestrian socially interacts with another pedestrian on the opposite side of the street and abruptly crosses the street driven by this interaction and the intention to meet the other. To the best of our knowledge, our work is the first to propose a pedestrian path predictor that uses effective social interaction signals such as gestures and head/body orientation. Moreover, none of the current approaches were evaluated on critical street-crossing synthetic situations based on real-life scenarios.

III. Problem Description

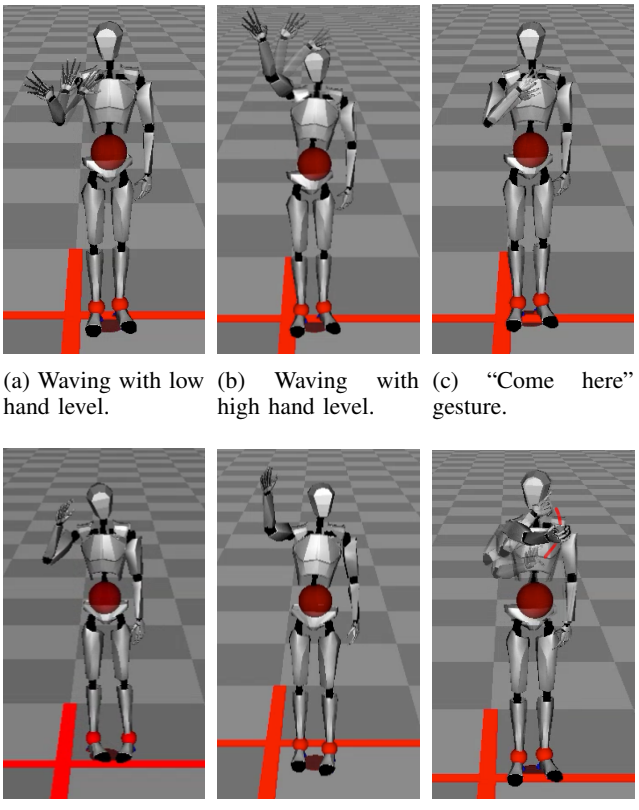
SIMP3 addresses the pedestrian path prediction problem as a sequence-to-sequence problem: It observes a pedestrian's state for a fixed amount of time steps, and then predicts the pedestrian's future locations for a predefined number of time steps. In the following, we briefly describe the pedestrian state together with the critical synthetic traffic scenarios that are considered.

A. Prerequisites

Pedestrian state: We define an instance of a pedestrian as follows: Let Pedestrians be the set of perceived pedestrians in the scene and let $i, j \in Pedestrians$ be two pedestrians on opposite sides of the street. Given some time point t, the current state of pedestrian i w.r.t j is described by the following tuple

$$\langle pos, \alpha, \beta, approaching, gesture, distance \rangle [j]_t^i$$

where $pos = (x, y) \in$ is the current 2D position of pedestrian i, $\alpha \in [0, \pi]$ is the head orientation of pedestrian i w.r.t. j, $\beta \in [0, \pi]$ is the body orientation of pedestrian i w.r.t. j, $approaching \in \{yes, maybe, no\}$ indicates whether pedestrian i is approaching j, $gesture$ (Fig. 1) gives the currently performed gesture of pedestrian i (if any), $distance \in \mathbb{R}_+$ is the distance between pedestrian i and j.



(a) Waving with low hand level. (b) Waving with high hand level. (c) "Come here" gesture.

(d) Arm raise with low hand level. (e) Arm raise with high hand level. (f) "I'm coming" gesture.

Fig. 1: Examples of pedestrian gestures in simulated scenes

Simulated critical pedestrian scenarios: For comparative evaluation of pedestrian path prediction methods by self-driving cars in critical scenarios, we used the open-source 3D driving simulator OpenDS6[1] and recorded real scenes to

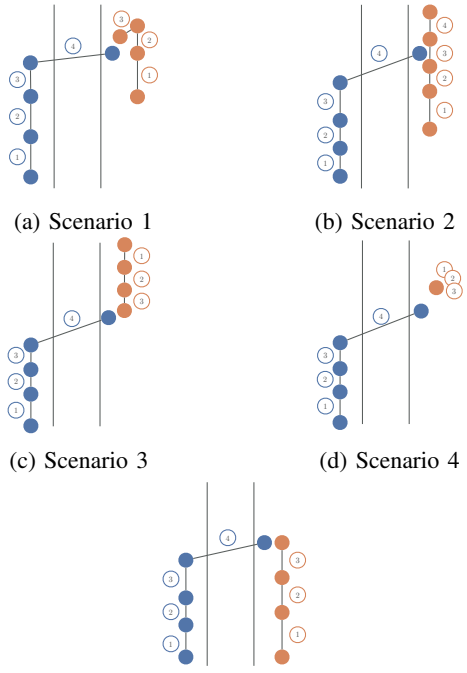
[1]OpenDS: https://opends.dfki.de

create an initial benchmark OpenDS-CTS2. This benchmark consists of about sixteen thousand traffic scenes with pairs of interacting pedestrians on opposite sidewalks of the street visible from the ego-perspective of the car. We consider seven different types of scenes or scenarios including five (Figure 3) where a pedestrian finally crosses the street and two (Figure 4) scenarios where no pedestrian crosses the street.

(a) Example of real scene (b) Synthetic scene in OpenDS

Fig. 2: Example real scene with corresponding synthetic one.

The scenarios cover different interactions with various kinds of social signals such as directed gaze and gestures that precede a sudden change of intention of pedestrians to cross the street (Figure 2); pedestrian paths in the synthetic scenarios are represented by means of waypoint segments, where the waypoints are delimiting changes in actions, i.e., from the second waypoint on the pedestrian turns his head towards the other and waves from the third waypoint on. The paths both pedestrians take in the scenarios are shown in Figure 3 and Figure 4, where one of them on the left sidewalk and his friend on the right sidewalk take the blue and the orange path, respectively.



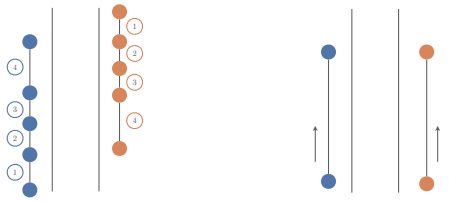
(a) Scenario 1 (b) Scenario 2

(c) Scenario 3 (d) Scenario 4

(e) Scenario 5

Fig. 3: Critical scenarios with two pedestrians

The scenarios 1-5 are critical scenarios where one pedestrian indeed crosses the street to meet the other. In particular, scenario 1 describes the situation, when a pedestrian sees someone he knows and wants to meet on the opposite side of the street from behind. This second pedestrian becomes aware of the first one, slows down, turns around, waves back to the first pedestrian on the left sidewalk, and waits for him to cross the street in order to catch up with him. Scenario 2 is similar to scenario 1 but instead of waiting, the friend on the right sidewalk only greets but then continues to walk into the same direction. Scenario 3 describes the situation where two pedestrians walk towards each other on different sides of the road, and one of them eventually crosses the street to meet the other. In scenario 4, one of both pedestrians is standing on one side of the street, waiting for the other, interacting across the street but neither of them actually crosses the street. In scenario 5, the two pedestrians notice each other and continue walking for some time in parallel to each other until one of them finally crosses the street to meet the other.



(a) Scenario 6 (b) Scenario 7

Fig. 4: Non-critical scenarios with two pedestrians

The scenario 6 and 7 are considered not critical, i.e., no pedestrian crosses the street. Scenario 6 describes the situation where people greet each other without having the intention to meet the other. In scenario 7, the pedestrians are walking on opposite sidewalks without any interaction and street crossing by either of them.

B. Pedestrian path-prediction problem

The task is to find a mapping f: $\mathbb{X} \rightarrow \mathbb{Y}$, where f in our case is a neural network. $\mathbb{X} = \{\mathrm{x}_{\mathrm{T}-\mathbf{n}}, \mathrm{x}_{\mathrm{T}-(\mathbf{n}-1)}, ..., \mathrm{x}_{\mathrm{T}}\}$, where x is the pedestrian state (as defined earlier) at a particular timestep. T is the current timestep and n is the number of observed timesteps. $\mathbb{Y} = \{\mathrm{loc}_{\mathrm{T}+1}, \mathrm{loc}_{\mathrm{T}+2}, ..., \mathrm{loc}_{\mathrm{T}+\mathrm{m}}\}$, where loc is the 2D location of the pedestrian and m is the number of predicted timesteps.

IV. SIMP3 Solution

A. Overview

The SIMP3 system makes use of dynamic Bayesian networks (DBN), called EMIDAS-DBNs, in combination with an encoder-decoder recurrent neural network in order to predict the future location of pedestrians (cf. Figure 5).

Given *n* + *m* pedestrians that are visible from the ego-view of the self-driving car, SIMP3 first determines through OpenDS6 all pairs of pedestrians on opposite sides of the street (bipartite graph for street with two opposite sidewalks), and

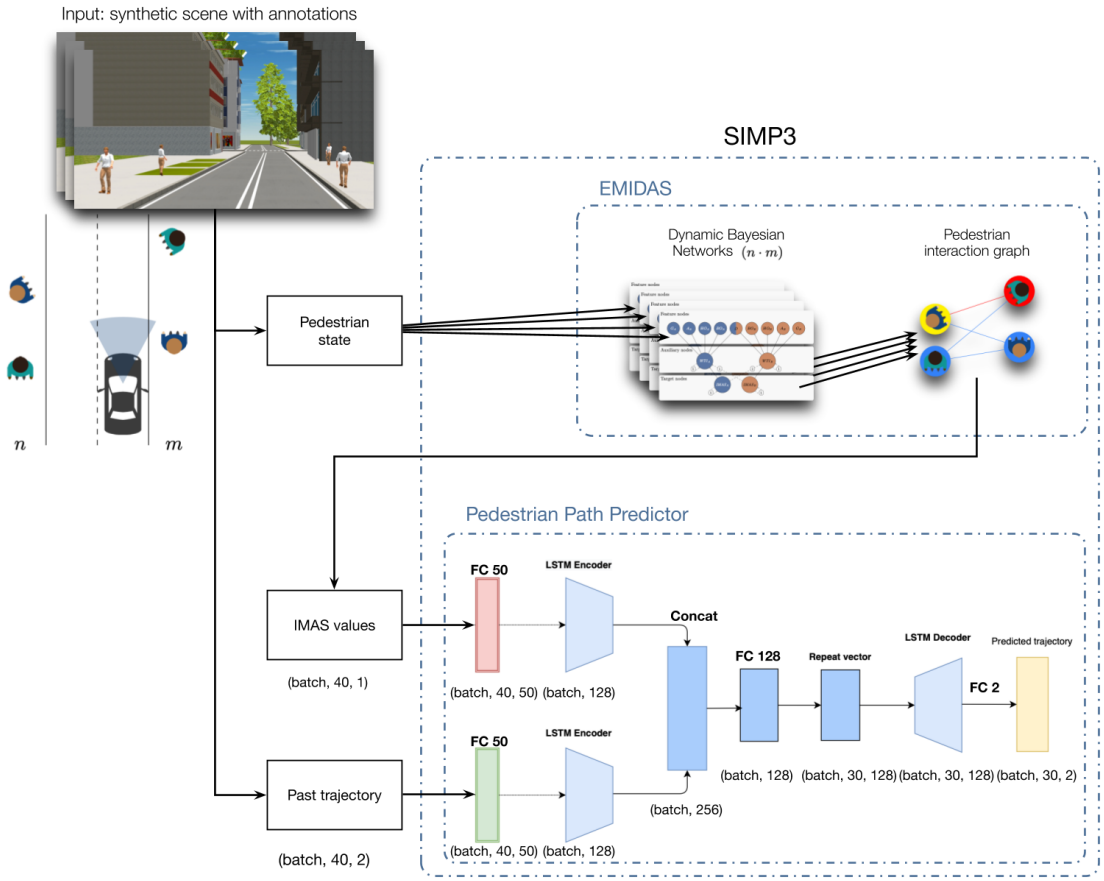


Fig. 5: SIMP3 architecture overview.

then processes them using $n \cdot m$ instances of trained DBN. Each DBN per pair takes as input the two pedestrian states (cf. Section III) and outputs the predicted intention of each of both to meet the other one across the street (IMAS values). The resulting dynamic bipartite pedestrian interaction graph (PInG) contains all predicted intentions of paired pedestrians in the scene at each time step. Each of the $n \cdot m$ pairs of pedestrians in the PInG is iterated through sequentially, extracting one intention estimation (IMAS) value for each pedestrian per time step; these values are then together with the past trajectory of the pedestrian passed to the neural network for path prediction.

B. Dynamic Bayesian Network EMIDAS-DBN

The EMIDAS-DBN (explainable multi-pedestrian interaction detection across street) is a dynamic Bayesian network that models dyadic social interaction for intention detection. In cognitive science and sociology, a dyad represents a group of two people, and the EMIDAS-DBN is the first cognitive causal model in support of detecting the intention of each of both being on opposite sides of the street to meet the other. The intention detection relies on observed social interaction signals of one or both pedestrians and the dynamic cause-effect relations between these signals, their individual willingness to interact and intention to meet up with each other.

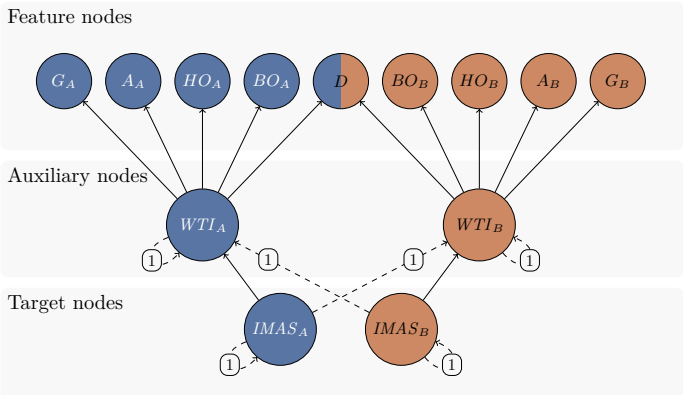


Fig. 6: Structure of an EMIDAS-DBN. Solid (dashed) edges represent instantaneous (temporal) causal effects.

In particular, the EMIDAS-DBN (cf. Figure 6) consists of leaf or feature nodes (observable pedestrian state from Section III), auxiliary nodes and target nodes for each of both pedestrians of a dyad; the node coloring in Figure 6 indicates to which of both the nodes belong. Feature nodes cover social signals of pedestrians including head orientation (*HO*) and body orientation (*BO*), currently performed gesture (*G*), whether

the other pedestrian is approached (*A*), and the distance (*D*) between both pedestrians. The target nodes represent the variable for the intention to meet across the street (IMAS) for each pedestrian of the dyad, and the auxiliary nodes model the willingness to interact (WTI) that may cause observable signals, while being considered as an effect of some intention IMAS. Both WTI and IMAS are not observable and take "very high", "high", "medium", "low", or "very low" as value. The solid (dashed) edges model instantaneous (temporal) causal influences between respective nodes of the DBN.

Note that a pedestrian may have the wish and willingness to interact with the other but nevertheless does not intend to cross the street to catch up with him. Detected intentions can be explained by the usual means of inference in a DBN such as for answering queries like "How likely is the prediction that pedestrian A has a very strong intention to meet pedestrian B in the next time step when pedestrian B was in the central field of view of pedestrian A and pedestrian A was strongly facing pedestrian B in the past nine time steps?".

The cognitive causal model of the EMIDAS-DBN has been developed with experts of cognitive science and psychology. Further, it has been evaluated on an initial benchmark OpenDS-CTS2 of scenes with IMAS and WTI ground truth annotations based on an initial user study and expert analysis of results. Due to the lack of an available benchmark and relevant studies on social signalling across street for intention detection, we created OpenDS-CTS2 as a first, initial benchmark of annotated scenes based on experience from our limited observations of social interactions between pedestrians with gestures across streets in Frankfurt. The EMIDAS-DBN parameters are learned by means of the Expectation–Maximization algorithm provided in GeNIe[2].

C. Pedestrian path predictor

As mentioned above, the pedestrian path prediction problem is solved sequence-to-sequence by use of a LSTM encoder-decoder network. The inputs to this predictor are the IMAS (intention to meet across the street) values of size (batch, 40, 1), and the past trajectory of the pedestrian of size (batch, 40, 2). These inputs have the format (*batch size*, *time steps*, *feature values*), where 1 timestep is equal to 0.1 seconds. Once the DBN is trained, it is used to compute the intentional factors as additional input to the path predictor during its training for each dyad of pedestrians on opposite sides of the street considered in the scene. That is, per time step, each of the respective $n \cdot m$ concurrently running instances of the trained DBN provides a scalar IMAS value for each pedestrian of the dyad to train the path predictor. Both inputs are passed through separate fully-connected layers of size 50 which serve as simple attention mechanisms, and are then encoded each with a dedicated LSTM encoder of size 128 with the following recurrent computation:

$$\begin{aligned}
\mathbf{f}_t &= \sigma_g(\mathbf{W}_f \cdot \mathbf{x}_t + \mathbf{U}_f \cdot \mathbf{h}_{t-1} + \mathbf{b}_f),\\
\mathbf{i}_t &= \sigma_g(\mathbf{W}_i \cdot \mathbf{x}_t + \mathbf{U}_i \cdot \mathbf{h}_{t-1} + \mathbf{b}_i),\\
\mathbf{o}_t &= \sigma_g(\mathbf{W}_o \cdot \mathbf{x}_t + \mathbf{U}_o \cdot \mathbf{h}_{t-1} + \mathbf{b}_o)\\
\tilde{\mathbf{c}}_t &= \sigma_g(\mathbf{W}_c \cdot \mathbf{x}_t + \mathbf{U}_c \cdot \mathbf{h}_{t-1} + \mathbf{b}_c)\\
\mathbf{c}_t &= \mathbf{f}_t \circ \mathbf{c}_{t-1} + \mathbf{i}_t \circ \tilde{\mathbf{c}}_t\\
\mathbf{h}_t &= \mathbf{o}_t \circ \sigma_h(\mathbf{c}_t)
\end{aligned} \tag{1}$$

where $\mathbf{x}$ is the input vector, $\mathbf{f}$ is the forget gate's activation vector, $\mathbf{i}$ is input gate's activation vector, $\mathbf{o}$ is the output gate's activation vector and $\mathbf{c}$ and $\mathbf{h}$ are cell and hidden state respectively. Initially, $\mathbf{c}_0 = 0$ and $\mathbf{h}_0 = 0$. The subscript $\mathbf{t}$ indicates the time step and σ_g (σ_h) the sigmoid (hyperbolic) tangent functions. Further, $\mathbf{W}$ and $\mathbf{U}$ are weight matrices and $\mathbf{b}$ is bias which are all learned during training. The outputs of the two LSTM encoders are then concatenated into a 256 dimensional vector. A fully-connected layer of size 128 extracts important features from this embedding before feeding it into the final LSTM decoder. All the fully-connected layers have a ReLu activation applied after them. The final fully-connected layer of size 2 brings the output of the decoder to the correct shape of (batch, 30, 2) for the prediction. Here, *batch* is the batch size, 30 is the number of future time steps and 2 is the 2D location of the pedestrian. For our task, we fixed the observed time steps to 40 (4 seconds) and the predicted timesteps to 30 (3 seconds).

Once the DBN is trained to provide IMAS values to the predictor, the training of the path predictor proceeds with minimising mean squared error (MSE) as loss function:

$$MSE = \frac{1}{N}\sum_{i=1}^{n}(Y_i - \widetilde{Y}_i)^2, \tag{2}$$

with Y the ground truth trajectory, $\widetilde{Y}$ the predicted trajectory, and N the total number of samples in the current batch. The batch size is set to 128, while the weights of the network are initialized using the Glorot (Xavier) uniform [8]. The *Adam* optimizer [13] is used with the initial learning rate of 10^{-3} which reduces by a factor of 10 every 10 epochs for a total of 30 epochs.

D. SIMP3 Inference

After the sequential training of both, that is, first a EMIDAS-DBN gets trained and then the path predictor based on the output of $n \times m$ instances of this trained DBN, the SIMP3 system can eventually perform its online inference (cf. Algorithm 1). Each scene in the benchmark (line 1) holds complete information about the state of both pedestrians, and holds for 4 seconds which is equal to 40 time steps in the past. The SIMP3 procedure takes this observed scene and outputs trajectory predictions for each pair of pedestrians for the next 3 seconds or 30 time steps into the future. SIMP3 iterates through each pair of pedestrians and through the observed number of time steps (40) and collects the IMAS value for each pedestrian by calling the trained DBN with pedestrian states from the

[2]GeNIe: https://www.bayesfusion.com/genie/

Algorithm 1 SIMP3 inference

global *scene* ▷ global variable to store all of the information about the observed scene
global *predictedTraj* ▷ global variable to store predicted trajectories for all pairs of pedestrians
Initialize a trained DBN instance (cf. Sect. IV.B)
Initialize a trained PathPredictor (cf. Sect. IV.C)

procedure SIMP3($scene$)
Input: $scene$ the current annotated observed scene for 4 seconds (40 timesteps)
Output: the predicted trajectories of each pedestrian in the scene for the next 3 s
▷ For each pair of pedestrians
  for all $(i, j) \in scene.pedLeft \times scene.pedRight$ **do**
    Initialize *pastTraj1, pastTraj2, pastImas1, pastImas2*
  ▷ For each timestep in the observed time period
    for $timestep \leftarrow 1$ to 40 **do**
    ▷ Get the pedestrian states for the current timestep
      $pedestrianStates$ = *scene(timestep)*
    ▷ Predict imas values for the pair using the current pedestrian states
      $imas1$, $imas2$ = DBN(*pedestrianStates(*i, j*)*)
    ▷ Store imas values for each pedestrian in the pair
      $pastImas1$.append($imas1$)
      $pastImas2$.append($imas2$)
    ▷ Get the location of each pedestrian in the pair
      $traj1$, $traj2$ = *pedestrianStates(i, j).location*
    ▷ Store the observed trajectory for each pedestrian
      $pastTraj1$.append($traj1$)
      $pastTraj2$.append($traj2$)
    end for
    ▷ Predict trajectory of each pedestrian sequentially
    $predictedTraj1$ = PathPredictor(*pastImas1, pastTraj1*)
    $predictedTraj2$ = PathPredictor(*pastImas2, pastTraj2*)
    ▷ Store the predicted trajectories of each pedestrian
    $predictedTraj$.append($predictedTraj1$)
    $predictedTraj$.append($predictedTraj2$)
  end for
  return $predictedTraj$
end procedure

current pedestrian pair. The past trajectory of each pedestrian is collected by just querying the current pedestrian state for the pedestrian's location. After the IMAS values and the past trajectory of each pedestrian is collected, the path predictor is called sequentially for each of the pedestrians in the pair. Finally, the SIMP3 returns the future trajectories for each pedestrian for the next 3 seconds (30 time steps). To optimize computational time, the calls to the path predictor can be ran in parallel, since they are independent of each other.

V. EVALUATION

A. Experimental Setting

We evaluated the predictions of SIMP3 against three state-of-the-art models in order to show how valuable the social interaction-based intention detection (IMAS values) is for predicting the trajectory of the considered pedestrians in critical scenarios. Each critical scenario covers an abrupt change of intention driven by prior social interaction between two pedestrian across the street. The evaluation is performed on the OpenDS-CTS2 benchmark dataset (see Section III) consists of 7 scenarios and 15,949 scenes in total (Table I) each of which contains only two pedestrians (one-dyad scene) on opposite sides of the street.

Scenario	**Scenes**	**Pedestrian states** (each 0.1 sec)	**Pedestrian Trajectories** (for $obs + pred = 7$ sec)
1*	2607	358 755	5214
2*	2134	257 736	4268
3*	7260	942 621	14 520
4*	375	49 311	750
5*	2985	377 248	5970
6	196	50 557	392
7	392	101 022	784
Total	15 949	2 137 250	31 898

TABLE I: Number of scenes and data points per scenario in OpenDS-CTS2 (extended). Scenarios marked with an asterisk are critical in the sense that a pedestrian crosses the street.

The OpenDS2 is used to train all models with leave-out-once cross validation, where each model is trained on scenes of six scenarios and validated on scenes of the seventh one. This process is repeated for all seven scenarios, such that all scenes of all seven scenarios are tested. None of the simulated scenes in OpenDS6 is longer than seven seconds, thus the sum of the observed (*obs*) and predicted (*pred*) horizon does not exceed seven seconds. The ad-hoc change of the pedestrian's intention to cross the street happens between the fourth and seventh second in the trajectory sequences. Since we are interested in predicting this ad-hoc change, we fix the observed timesteps of all models to four seconds and prediction horizon to three seconds.

The total amount of scenes was achieved by different augmentations of the original scenarios: The slow walking speed of the pedestrians was varied in the interval $\{1.3,1.5,1.7\}$ *m/s* as well as the fast walking speed in the interval $\{1.9,2.1,2.3,2.5\}$ *m/s*. Furthermore, additional scenes are generated by mirroring the pedestrians' path along the midline of the road and along the axis perpendicular to the midline of the road. Additionally, we randomly alter the waypoints within a circle of radius *r=50* cm around the original waypoint in each scene. The amount of scenes per scenario are not evenly distributed because all possible variations of the pedestrian velocities were generated such that the original scenario configuration is still obeyed.

The simulator (OpenDS) provides annotations for each pedestrian every 0.1s. The annotations include the current position, whether the pedestrian looks towards the other pedestrian and

FDE/ADE in meters							
Method	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6	Scenario 7
SIMP3	**2.02**/1.45	**1.92**/1.49	**1.6**/1.03	**1.62**/1.49	**1.88**/1.16	3.45/1.85	**1.7**/1.35
Social-GAN	2.53/*1.0*	2.24/*0.89*	2.21/*0.89*	2.52/*1.07*	**1.86**/*0.75*	**3.09**/*1.34*	2.76/*1.14*
Social-STGCNN	3.23/1.87	3.63/2.05	2.86/1.59	3.16/1.85	3.13/1.92	3.96/1.98	4.57/2.48
STGAT	2.51/1.06	2.43/1.0	2.48/1.04	2.52/1.08	2.76/1.15	3.40/1.46	3.62/1.43

TABLE II: Comparative performance evaluation between SIMP3, Social-GAN, and Social-STGCNN on OpenDS-CTS2 for each scenario by means of the Final displacement error (FDE) and average dispacement error (ADE) in meters.

whether the pedestrian currently performs a gesture (and if so, which gesture). The pedestrian states in Table I represent these annotations and were used to train the DBN. Each pedestrian trajectory in the last column of Table I is a seven second sequence of pedestrian's position which were used to train our path predictor as well as the baseline models. Main evaluation metric for our experiments is the final displacement error (FDE), which is the Euclidean distance between the ground truth and the predicted position at the last time step $t = T_{pred}$ averaged over all data points N:

$$FDE = \frac{1}{N}\sum_{i=1}^{N} \|Y^i_{T_{pred}} - \widetilde{Y}^i_{T_{pred}}\|_2 \tag{3}$$

Another standard metric is the Average Displacement Error (ADE), which is defined as the average Euclidian distance (ℓ_2-norm) between the ground truth Y and the predicted position $\widetilde{Y}$ over all time steps T_{pred} in the prediction horizon for all data points N:

$$ADE = \frac{1}{N \cdot T_{pred}} \sum_{i=1}^{N} \sum_{t=1}^{T_{pred}} \|Y_{i,t} - \widetilde{Y}_{i,t}|\_2 \tag{4}$$

We compare our SIMP3 solution to the following state-of-the-art baselines:

1) **Social-GAN** [9] uses a generative adversarial network combined with a recurrent neural network and a social pooling mechanism to model the pedestrian path prediction problem.
2) **Social-STGCNN** [1] uses Social Spatio-Temporal Graph Convolutional Neural Network to model people's interactions as a graph.
3) **STGAT** [10] uses graph attention mechanism to capture spatial interactions as well as an LSTM to encode the temporal interactions of the pedestrians.

All experiments were performed on a PC with an Intel Core i7 7th Gen @ 3.60 GHz, an Nvidia GTX 1070 GPU and 16 GB RAM.

B. Results

In this section, the results of the comparative experimental performance evaluation are summarized. First, the different models differ in terms of inference time as shown in Table III with mean (M) and standard deviation (SD) of the inference time (in milliseconds). Due to the sequential inference by SIMP3, that is, the EMIDAS-DBN feeds its derived intention factors as additional input to the RNN for prediction, the overall inference running time of SIMP3 is longer than that of Social-GAN and SocialSTGCNN, though it is faster than STGAT.

Model	**Inference runtime** (in ms)
SIMP3	$9.7899\ (= 8.332 + 1.457)$
EMIDAS	$M = 8.332, SD = 0.807$
Path prediction	$M = 1.457, SD = 0.037$
Social-GAN	$M = 6.894, SD = 0.072$
Social-STGCNN	$M = 1.088, SD = 0.0582$
STGAT	$M = 46.793, SD = 1.252$

TABLE III: Inference runtimes of SIMP3, Social-GAN, Social-STGCNN and STGAT with the setting *observation* 4 / *prediction* 3 for one pedestrian.

As shown in Table II, SIMP3 outperforms Social-STGCNN in all scenarios in terms of both FDE and ADE. In addition, SIMP3 outperforms Social-GAN in five out of seven, and STGAT in six out of seven scenarios in terms of FDE. In particular, the results of SIMP3 compared to Social-GAN are better in four out of five critical scenarios, with the results for both in scenario 5 being very close (0.02 mtrs).

In terms of ADE, Social-GAN outperforms SIMP3 in all scenarios. However, we are interested in predicting whether a pedestrian will step on the street or not after an abrupt change of intention in a critical situation. Therefore, we believe that the FDE metric captures this objective better as it measures the error of the final predicted position of the pedestrian. A method can achieve a better score in terms of ADE than in terms of FDE when a large part of the future trajectory does not exhibit sudden changes of intention to cross the street.

There are two remarks in order: First, one reason why the FDE of SIMP3 is very similar to that of Social-GAN in scenario 5 could be due to the paths of both pedestrians. Recall that in scenario 5, the pedestrians are walking synchronously on opposite sides of the street. As one consequence, the distance between the pedestrians is not reduced before either pedestrian crosses the street. From this we conclude that for such non-critical scenes the types of social signals considered by the EMIDAS-DBN do not alone suffice to distinguish the intention levels (IMAS value) of both pedestrians just before their stepping on the street. This, in turn, implies that SIMP3 could only rely on the past prediction of the pedestrians, just as Social-GAN. Therefore, the overall direction of the paths predicted by SIMP3 in scenario 5 are probably similar to the ones predicted by Social-GAN. Second, SIMP3 does not outperform Social-GAN and STGAT on Scenario 6, which is

the only scenario where pedestrians interact with each other but do not cross the street. Besides, it is the scenario with the lowest amount of data points in the dataset.

VI. Conclusion

We presented the first socially-aware multi-pedestrian path predictor SIMP3 that takes observed social interaction between pedestrians on opposite sides of the street into account. In particular, SIMP3 combines dynamic Bayesian networks for intention detection of pedestrians with a recurrent network for path prediction. The performance of SIMP3 has been evaluated against selected competitors on the initial benchmark OpenDS-CTS2 of about sixteen thousand traffic scenes with socially interacting pedestrians across the street simulated in OpenDS. The results revealed that SIMP3 can outperform the selected baselines for socially-aware pedestrian path prediction in terms of final displacement error in, while remaining competitive in terms of average displacement error. Future work is concerned with, among others, extending the OpenDS-CTS2 benchmark with simulated scenarios of n-ary ($n > 2$) social interaction between pedestrian groups on same and opposite sidewalks.

Acknowledgments

We very thankfully acknowledge the great support of (parts of) the reported work by Tanja Simeonovski, Anthony Heggen, Asha Upperla, and Dikshant Gupta.

References

[1] Abduallah, M.; Kun, Q.; Mohamed, E.; Claudel, C. (2020): Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction. Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[2] Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; Savarese, S. (2016): Social LSTM: Human trajectory prediction in crowded spaces. Proc. of IEEE conference on Computer Vision and Pattern Recognition (CVPR).

[3] Baur, T. (2018): Cooperative and transparent machine learning for the context-sensitive analysis of social interactions. Dissertation, CSD, University of Augsburg, Germany.

[4] Blaiotta C. (2019): Learning generative socially aware models of pedestrian motion. IEEE Robotics and Automation Letters, 4(4):3433–3440.

[5] Elnagar A.; Gupta K (1998): Motion prediction of moving objects based on auto-regressive model. IEEE Trans. on Syst., Man, and Cybernetics (SMC) - Part A: Systems and Humans 28(6):803–810. DOI:10.1109/3468.725351

[6] Ferrer G.; Sanfeliu A (2014): Behavior estimation for a complete framework for human motion prediction in crowded environments. Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA).

[7] Gebhardt, P., et al. (2018): MARSSI: Model of Appraisal, Regulation, and Social Signal Interpretation. Proc. 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)

[8] Glorot, X.; Bengio, Y. (2010). Understanding the difficulty of training deep feed-forward neural networks. Proc. of 13th Int. Conf. on Artificial Intelligence and Statistics (AISTATS), volume 9 ofJMLR Proceedings. JMLR.org

[9] Gupta, A.; Johnson, J.; Fei-Fei, L.; Savarese, S.; Alahi, A. (2018): SocialGAN: Socially acceptable trajectories with generative adversarial networks. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[10] Huang, Y.; Bi, H.; Li, Z.; Mao, T.; Wang, Z. (2019): STGAT: Modeling Spatial-Temporal Interactions for Human Trajectory Prediction. Proc. of IEEE Int. Conf. on Computer Vision (ICCV).

[11] Huynh M.; Alaghband G. (2019): Trajectory prediction by coupling scene-LSTM with human movement LSTM. Proc. of Int. Symposium on Visual Computing. Springer.

[12] Karasev V.; Ayvaci A.; Heisele B.; Soatto S. (2016): Intent-aware long-term prediction of pedestrian motion. Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA).

[13] Kingma, D.P.; Ba, J. (2015): Adam: A method for stochastic optimization. Proc. of 3rd Int. Conf. on Learning Representations (ICLR).

[14] Kosaraju, V.; Sadeghian, A.; Martín-Martín, R.; Reid, I.; Rezatofighi, SH.; Savarese, S. (2019):. Social-BiGAT: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. arXiv preprint arXiv:1907.03395.

[15] Lee, N.; Choi, W.; Vernaza, P.; Bongsoo Choy, C.; Torr, PHS.; Krishna Chandraker, M. (2017): DESIRE: Distant Future Prediction in Dynamic Scenes with Interacting Agents. CoRR abs/1704.04394; arXiv:1704.04394 http://arxiv.org/abs/1704.04394.

[16] Liang, J.; Jiang, L.; Niebles, JC.; Hauptmann, AG.; Fei-Fei, L. (2019): Peeking into the Future: Predicting Future Person Activities and Locations in Videos. In: CoRR abs/1902.03748 (2019). arXiv:1902.03748 http://arxiv.org/abs/1902.03748

[17] Makris, D.; Ellis, T.J. (2002): Spatial and Probabilistic Modelling of Pedestrian Behaviour. In: BMVC.

[18] Muscholl, N.; Poibrenski, A. (2020): SIMP3 sources with OpenDS-CTS 2.0 benchmark. https://github.com/atanas1054/SIMP3

[19] Nikhil N.; Tran Morris B. (2018): Convolutional neural network for trajectory prediction. Proc. of Europ. Conf. on Computer Vision (ECCV).

[20] Paris S.; Pettre J.; Donikian S. (2007): Pedestrian reactive navigation for crowd simulation: a predictive approach. Computer Graphics Forum, 26:665-674. Wiley Online Library.

[21] Pellegrini S.; Ess A.; van Gool L. (2010): Improving data association by joint modeling of pedestrian trajectories and groupings. Proc. of Europ. Conf. on Computer Vision (ECCV). Springer.

[22] Rhinehart N.; Kitani K.; Vernaza P. (2018): R2P2: A ReparameteRized Pushforward Policy for diverse, precise generative path forecasting. Proc. of Europ. Conf. on Computer Vision (ECCV).

[23] Robicquet A.; Sadeghian A.; Alahi A.; Savarese S. (2016): Learning social etiquette: Human trajectory understanding in crowded scenes. Proc. of Europ. Conf. on Computer Vision (ECCV). Springer.

[24] Rudenko A., Palmieri L.; Arras KO. (2017): Predictive planning for a mobile robot in human environments. Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), Workshop on AI Planning and Robotics.

[25] Rudenko, A.; Palmieri, L.; Herman, M.; Kitani, K.M.; Gavrila, D.M.; Arras, K.O. (2020): Human motion trajectory prediction: a survey. Journal of Robotics Research, 39(8):895–935.

[26] Sadeghian, A.; Kosaraju, V.; Sadeghian, A.; Hirose, N.; Rezatofighi, H.; Savarese, S. (2019): Sophie: An attentive GAN for predicting paths compliant to social and physical constraints. Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[27] van den Berg, J.P.; Guy, S.J.; Lin, M.C.; Manocha, D. (2009): Reciprocal n-Body Collision Avoidance. In: ISRR.

[28] Vasishta P.; Vaufreydaz D.; Spalanzani A. (2018): Building prior knowledge: A Markov based pedestrian prediction model using urban environmental data. Proc. of Int. Conf. on Control, Automation, Robotics and Vision (ICARCV).

[29] Vianciarelli, A.; Pentland, A. (2015): New Social Signals in a New Interaction World: The Next Frontier for Social Signal Processing. IEEE Trans. Systems, Mans and Cybernetic (SMC).

[30] Xue H.; Huynh D.; Reynolds M. (2019): Location-velocity attention for pedestrian trajectory prediction. Proc. of IEEE Winter Conf. on Applications of Computer Vision (WACV). IEEE.

[31] Yamaguchi K.; Berg AC.; Ortiz LE.; Berg TL. (2011): Who are you with and where are you going? Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). DOI:10.1109/CVPR.2011.5995468.

[32] Xue, H.; Huynh, DQ.; Reynolds, M. (2018): SS-LSTM: A Hierarchical LSTM Model for Pedestrian Trajectory Prediction. Proc. of IEEE Winter Conf. on Applications of Computer Vision (WACV), Lake Tahoe, NV. doi: 10.1109/WACV.2018.00135.

[33] Zanlungo F., Ikeda T.; Kanda T. (2011): Social force model with explicit collision prediction. Europhysics Letters EPL, 93(6):68005.

[34] Zhang, P.; Ouyang, W.; Zhang, P.; Xue, J.; Zheng, N. (2019): SR-LSTM: State refinement for LSTM towards pedestrian trajectory prediction. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[35] Zhu, Q. (1991): Hidden Markov Model for dynamic obstacle avoidance of mobile robot navigation. IEEE Trans. on Robotics and Automation (TRO), 7(3):390–397.