Contextual Recurrent Predictive Model for Long-Term Intent Prediction of Vulnerable Road Users

Khaled Saleh, Mohammed Hossny, and Saeid Nahavandi

Abstract— **Recently, the problem of intent and trajectory prediction of vulnerable road users (VRUs) in urban traffic environments has got some attention from the intelligent transportation research community. One of the main challenges that make this problem even harder is the uncertainty exists in the actions of pedestrians in urban traffic environments, as well as the difficulty in inferring their end goals. In this paper, we are proposing a data-driven framework based on inverse reinforcement learning (IRL) and the bidirectional recurrent neural network architecture (B-LSTM) for long-term prediction of VRUs' intention. We evaluated our framework on real-life datasets for agent behavior modeling in traffic environments, and it has achieved an overall average displacement error of only 2.93 and 4.12 pixels over 2.0 and 3.0 s ahead prediction horizons, respectively. In addition, we compared our framework against other baseline models based on sequence prediction models and planning-based approaches. We have outperformed these approaches with the lowest margin of average displacement error of more than 5 pixels. Furthermore, the performance of the proposed framework was evaluated on an additional vehicle-based video sequence dataset for path prediction of pedestrians and it continued to achieve robust results with higher generalization capabilities.**

Index Terms— **Intent prediction, vulnerable road users (VRU), autonomous vehicles, motion trajectory forecasting, reinforcement learning (IRL).**

I. Introduction

RECENTLY, the development of autonomous vehicles (AVs) has reached major milestones and witnessed a number of success in highway traffic environments [1], [2]. That being said, they are still however faced with a number of challenges in urban traffic environments. More specifically when it comes to interacting with VRUs such as pedestrians and cyclists [3]–[5]. Thus, the necessity for having predictive models within these vehicles that can infer and understand the VRUs intentions over longer time periods become inevitable. In the advanced driving assistance systems (ADAS) community, the intent prediction of VRUs problem, especially pedestrians, has been thoroughly investigated over the past few years [6]–[8]. In this respect, the intent prediction problem is commonly accomplished based on forecasting the motion trajectories of VRUs in traffic environments. Since the driver in ADAS context is still in command of the driving decisions most of the time, thus the predictive models for intent prediction proposed in the literature were only focused on predicting shorter time horizons of VRUs' intention. On the other hand, in the context of AVs, where the driver will be out of the decision-making loop, the need for long-term prediction is essential. As it was illustrated in [3] and [9], longer time horizons of intent prediction of VRUs were attributed to being one of the most strong cues for a trusted interaction between VRUs and AVs.

Commonly, the approaches for the intent prediction introduced in ADAS field rely either on single linear dynamical models such as Kalman filter [6] or a switching multiple linear dynamical models [7], [8]. One constraint of linear dynamical models they need an explicit modelling of the agent (i.e. pedestrian) in the traffic scene. They are also challenged by the variable non-linear and uncertain dynamics of VRUs over longer time periods. Another approach has been also investigated in the literature is planning based models. Unlike linear dynamical models, planning based models do not need an explicit modelling of the VRUs. Additionally, they have been proven to give a resilient prediction of VRUs over a longer time horizon [10]–[12]. However, one challenge planning-based models are still encountered with, is their inherent reliance on a prior known end goal. Data-driven approaches [13], [14] are also another stream of approaches proposed for tackling the intent prediction problem of VRUs. Similar to planning-based models they do not need an explicit modelling of the VRUs' motion dynamics. However, unlike planning-based models, they do not need a prior end goal for the VRUs to be known beforehand. Despite the promising results that data-driven approaches have shown for the intent prediction problem for VRUs, they still need some improvements. For instance, so far the proposed data-driven approaches in the literature do not take into account the inherent uncertainty of the VRUs' actions. Additionally, they are also neglecting the effect of the physical environment on the VRUs' actions.

Thus, in this paper, we are proposing a framework that combines the best of the two worlds of planning-based models and data-driven approaches. We first learn the reward function of the traffic environment by just observing the demonstrated trajectories of VRUs via inverse reinforcement learning (IRL).

Manuscript received December 7, 2018; revised April 30, 2019; accepted June 21, 2019. Date of publication July 19, 2019; date of current version July 29, 2020. This work was supported by the Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University. The Associate Editor for this paper was M. Da Lio. *(Corresponding author: Khaled Saleh.)*

The authors are with the Institute for Intelligent Systems Research and Innovation, Deakin University, Geelong, VIC 3216, Australia (e-mail: kaboufar@deakin.edu.au; mohammed.hossny@deakin.edu.au; saeid.nahavandi@deakin.edu.au).

Digital Object Identifier 10.1109/TITS.2019.2927770

Then using the learned reward function alongside the motion trajectory of VRUs we learn another RNN model. The RNN model in return infers a long-term trajectory without prior information about the VRUs' end goal. The main contribution of this work as an extension to our early results in [15] is four-folds:

- A data-driven framework for long-term prediction of vulnerable road users' intention via their motion trajectories.
- Contextual-aware predictive model that takes into account the VRUs preferences when navigating urban traffic environments.
- Resilient and modular approach for accommodating the various factors that influence VRUs actions based on inverse reinforcement learning.
- Robust framework that can provide robust generalisation capabilities across different traffic environments.

The rest of this paper is organized as follows. Section II presents a brief literature review of the problem. Section III describes the problem formulation and the proposed solution. Section IV describes the datasets used and the validation method. Finally, Section V concludes this paper.

II. Related Work

The intent prediction problem of VRUs has been commonly approached in the literature as a dynamical motion modelling problem solved using recursive Bayesian filters [6], [7], [16]. In [6], one of the early work on the intent prediction problem of VRUs, they used an Extended Kalman filter (EKF) to model the linear dynamical motion model of pedestrians in four distinctive crossing scenarios. Keller and Gavrila [7], relied on another dynamical motion model based on a Gaussian process to infer whether a pedestrian walking on a curb will cross or not. In their work, they developed two different motion models to identify the stopping and walking behaviour of the pedestrians. They relied on optical flow fields so that they could predict the trajectory of the pedestrian. More recently, Pool *et al.* [16] pursued the same approach introduced in [8], but for the cyclist trajectory prediction problem using a stereo-based camera from a moving vehicle. They presented two models; the first one is based on the standard Kalman filter (KF) with a CV underlying dynamical motion model for a linear cyclists' trajectory prediction. The other model is based on a mixture of five linear dynamical motion models based on KF as well.

Another category of approaches utilized in the literature for the intent and trajectory prediction of pedestrians is the planning-based models. These models are inspired by the path planning approaches that are heavily used in the robotics field. Unlike the traditional path planning approaches where an ego-centric trajectory to be performed by a robot in an environment, they used it to plan a trajectory of other agents (i.e, pedestrian). In planning-based approaches, the inherent assumption is that the end goal the agent is trying to reach is known in advance which might not be foreseeable in the case of pedestrians. In [17], a planning-based approach used for forecasting pedestrians' trajectories in traffic environments. Since the end goal of the pedestrians is not known beforehand, they firstly infer a set of possible goals using a combination of Gaussian Mixture Model (GMM) and Particle Filter (PF). Using these inferred end goals and an occupancy grid map of the environment, they can predict a probability distribution over the probable trajectories to these goals.

Yet another planning-based approach was introduced in [18] for on-road pedestrian avoidance system for an autonomous mobile robot. The pedestrian avoidance system took into account the pedestrians' intention and their associated uncertainty as part of the robot's motion planning framework. They formulated the problem as a Mixed Observable Markov Decision Process (MOMDP), where the motion model variables of the pedestrian are already given (fully observable) and the intention of the pedestrian is unknown (unobserved). They assume the pedestrian is directed towards his/her goal following the shortest path trajectory. They utilized a sampling-based approximate algorithm called Successive Approximations of the Reachable Space under Optimal Policies (SARSOP). Using SARSOP, they solved the MOMDP model and inferred a probability distribution over the possible directions of the pedestrians.

Data-driven approaches specifically those ones based on recurrent neural network architectures such as long short-term memory (LSTM) were also investigated for the intent and trajectory prediction problem of pedestrians [13], [14]. In data-driven approaches and similar to planning-based approaches, no explicit modelling for the dynamics of the motion of pedestrians is needed to be performed firstly. However, unlike planning-based approaches, they do not require prior information regarding the end goal of the pedestrians in the scene. In [14], recurrent neural network (RNN)-based approach was used for modelling human-to-human interactions in crowded environments from a surveillance camera's perspective. In [13], another RNN-based model for predicting trajectories of pedestrians in traffic environments was also introduced. In their presented RNN model, they relied only on past positional information of pedestrians in order to predict their future motion trajectories. Similarly in [19], an RNN-based model was proposed for predicting whether pedestrians would cross or not in specific urban traffic scenarios. Based on two different input sequence features extracted from 3D LIDAR scans at an intersection, they tackled the problem as a binary classification problem. The two types of input features are namely, temporal (pedestrian's position over time) and geometrical (pedestrians' distance to curb).

III. Proposed Methodology

In this section, the proposed methodology for the intent prediction problem for VRUs in urban traffic environment will be discussed. Firstly, we will begin with our formulation of the problem. Then, the building blocks of our proposed framework (shown in Figure 1) will be described.

A. Problem Formulation

In our formulation for the intent prediction problem of VRUs in a traffic environment, we cast the problem as a probabilistic sequence prediction problem. Given a sequence of past trajectory observations x as well as a reward map r that represents the VRU's preference in an urban traffic

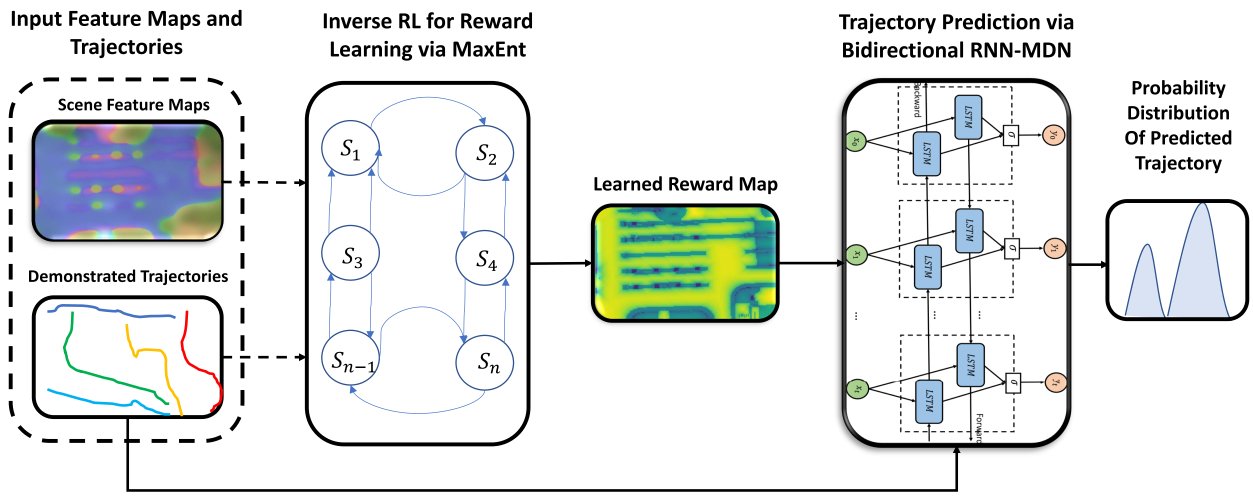

Fig. 1. The proposed framework for long-term trajectory prediction of pedestrians in urban traffic environment. Firstly, demonstrated trajectories and contextual features maps are used for learning the reward map of the scene via IRL MaxEnt. The demonstrated trajectories along with the learned reward map of the scene are passed as the input sequences for training a probabilistic trajectory prediction B-LSTM-MDN model. The output of the B-LSTM-MDN model are probability density of future sequence trajectories of input pedestrians.

environment. In return, we will anticipate the probability density $P(y|x, r)$ of the pedestrian's future trajectory y. In order to achieve a probabilistic sequence prediction model, we will utilize a bidirectional recurrent neural network model based on LSTM architecture [20] with a mixture density network on top of it [21]. For recovering a reward map that can accurately capture the VRUs' preferences, we will rely on an inverse reinforcement learning (IRL) technique [22].

B. IRL and Markov Decision Process

MDP is one of the most widely used frameworks for modelling the dynamics of a decision making process [23]. MDP can be defined as $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{T}, r\}$, where $\mathcal{S}$ is the state space of the system, $\mathcal{A}$ are the possible actions, $\mathcal{T}$ is the transition model that describes the system dynamics and r is the reward function. Typically acting in an MDP results in a sequence of states and actions $\{s_0, a_0, s_1, a_1, s_2, \ldots\}$. A policy π, is the mapping sequences $(\mu_0, \mu_1, \mu_2, \ldots)$, where at any time t the mapping $\mu_t(\cdot)$ determines the action $a_t = \mu_t(s_t)$ to take when in state s_t. The ultimate goal in an MDP, is to find an optimal policy π^*, that maximises the expected sum of rewards accumulated over time.

In IRL context, the specifications of MDP are available except the reward function r is unknown. Alternatively, a set of demonstrations $\mathcal{D} = \{\zeta_1, \zeta_2, \ldots, \zeta_N\}$ are provided by a demonstrator. Each sample trajectory ζ_i from the set of demonstration $\mathcal{D}$ is described by a pair of state-action according to $\zeta_i = \{(s_0, a_0), (s_1, a_1), \ldots, (s_T, a_T)\}$. Given the demonstrations $\mathcal{D}$, the goal of IRL is to recover the reward function r that can ultimately capture the preference of the agent. Since in real life applications, it would be difficult to observe a reward function for each action-state pair in the set of demonstration $\mathcal{D}$, especially if the state space is large. Thus, a common approach in IRL methods is collecting a feature vector f that best characterises each possible action from the set of demonstrations $\mathcal{D}$.

C. Reward Learning for VRU Intent and Trajectory Prediction

One of the most commonly used approaches for IRL is the MaxEnt proposed in [22]. MaxEnt was successfully utilised in a number of applications such as learning driver behaviours [11], planners for social robotics [22], [24] and activity forecasting from surveillance data [10], [25]. In the formulation for the MaxEnt, the reward function can be calculated as a weighted linear combination of the feature values vector f according to Eq. 1.

$$r = \theta^T f, \tag{1}$$

where θ is a vector of unknown weights.

In this work, we will be focusing on the contextual physical information in urban traffic environment as our feature values vector for parametrising the reward function. More specifically, we will utilise the vision-based contextual information extracted from the environment by employing image semantic segmentation techniques. The contextual physical information will be the common ones that could have a potential influence on the future actions of VRUs such as trees, buildings, sidewalks, and roads. Using demonstrated trajectories of VRUs in urban traffic environments along with contextual physical information, MaxEnt can be adopted for learning the reward function parameters. In MaxEnt, the probability distribution of a trajectory ζ_i is proportional to the exponentiated sum of rewards along the trajectory ζ_i. The calculation of ζ_i can be easily accomplished using Eq 3 after the substitution in Eq 1.

$$P(\zeta_i) \propto \exp \sum_{(s,a)\in\zeta_i} r_{s,a} \tag{2}$$

$$P(\zeta_i|\theta) = \frac{\exp \sum_{(s,a)\in\zeta_i} \theta^T f_{s,a}}{Z(\theta)} \tag{3}$$

where $Z(\theta)$, is the normalization function.

Algorithm 1: Maximum Entropy IRL

input : $\mathcal{S}, \mathcal{A}, \mathcal{T}, f, \bar{f}$
output: Optimal set of weights $\hat{\theta}$
$\theta^1 = IRL\_initWeights()$
for *n=1:N* **do**
  $\pi^n = IRL\_valueIteration(\mathcal{S}, \mathcal{A}, \mathcal{T}, f, \theta^n)$
  $\hat{f}^n_\theta = IRL\_stateVisitFrequency(\mathcal{S}, \mathcal{A}, \mathcal{T}, f, \pi^n)$
  $\nabla \mathcal{L}^n_\theta = \bar{f} - \hat{f}^n_\theta$
  $\theta^{n+1} = IRL\_updateWeights(\theta^n, \nabla \mathcal{L}^n_\theta)$
end

Algorithm 2: IRL_valueIteration

$V(s) = -\infty$
for *n=N:1* **do**
  $V(s_{goal}) = 0$
  $Q^n(s, a) = \theta^T f_{s,a} + \mathbb{E}_{\mathcal{T}(s,a,s')}[V^n(s')]$
  $V^{n-1}(s) = \text{soft}\max_a Q^n(s, a)$
end
$\pi(a, s) = \exp^{Q(s,a)-V(s)}$

Algorithm 3: IRL_stateVisitFrequency

$\phi(s_{start}) = 1$
for *n=1:N* **do**
  $\phi_{s_{goal}} = 0$
  $\phi^{n+1}_s = \sum_{s',a} \mathcal{T}(s, a, s')\pi(a, s')\phi^n(s')$
end
$\phi_s = \sum_n \phi^n_s$
$\hat{f}_\theta = \sum_s \phi_s f_s$

By maximising the entropy of Eq 3, learning from demonstration trajectories in MaxEnt can be accomplished. Additionally, the maximisation of the entropy of Eq 3 can be interpreted as minimising the gradient of the log-likelihood of the same equation, which in returns can be calculated using learning algorithms such as conjugate or stochastic gradient descent. That been said, we will be using the similar forward-backward algorithm introduced and discussed in [10] for training the MaxEnt framework and obtaining the weights θ of the reward function. In the forward-backward algorithm (outlined in Algorithm 1, 2 and 3), the objective is to minimize the gradient of the likelihood of Eq 3 using a gradient descent algorithm. The gradient is calculated based on the difference between the empirical cumulative feature count $\bar{f}$ and the expected cumulative feature count $\hat{f}_\theta$ in Eq 4.

$$\nabla \mathcal{L}_\theta = \bar{f} - \hat{f}_\theta \tag{4}$$

The empirical cumulative feature count, is the average accumulated features over the demonstrated trajectories $\mathcal{D}$, and can be computed according to Eq 5

$$\bar{f} = \frac{1}{\mathcal{D}} \sum_{i=1}^{\mathcal{D}} f_{\zeta_i} \tag{5}$$

On the other hand, the expected cumulative feature count is the average accumulated features according to the trajectories generated by the weights. The expected cumulative feature count can be expressed using the expected state visitation frequency property ϕ_s according to Eq 6.

$$\hat{f}_\theta = \sum_s \phi_s f_s \tag{6}$$

The sum of visits that have been to each state s are described over time by ϕ_s and can be calculated as shown in Algorithm 3. At the convergence, when $\bar{f}$ equals $\hat{f}_\theta$, an optimal set of weights $\hat{\theta}$ can be obtained as described in Algorithm 1. As a result, these weights can be used to have reward functions as the ones visualised in Figure 2.

D. Probabilistic Trajectory Prediction via Bidirectional LSTM

Due to their capabilities in modelling complex temporal dependency of their input sequence information as it was shown in [13], [14], RNNs have been achieving resilient results in sequence prediction tasks [26], [27]. Thus, in the proposed probabilistic framework for trajectory prediction of VRUs in the urban traffic environment, we will be capitalising on their powerful sequence-to-sequence modelling capabilities. In specific, we will be utilising the B-LSTM architecture [28]. In general, the operation of conventional LSTM architecture is governed by three main internal gates which dictate which information to be persisted over time and which to be forgotten. Therefore, LSTM is considered one of the best RNN architectures for memorising longer-term information. The aforementioned conventional LSTM is usually referred to as U-LSTM, that because they process the information in only one direction which is the forward direction. On the other hand, B-LSTM can process the information in two directions, namely forward and backward which make them more capable of understanding a much higher level of abstraction of their input information [28].

Both U-LSTM and B-LSTM architectures output predictions of deterministic real target values, however in real-life there is usually an inherent uncertainty especially concerning our VRUs trajectory prediction problem. As a result, we will be augmenting B-LSTM architecture with an output layer of a mixture density network (MDN) [21]. The MDN layer will generate a weighted sum of various probability distributions that can account for the uncertainty of VRUs trajectories in urban traffic environments.

In Figure 3, the proposed B-LSTM-MDN for VRUs trajectory prediction is shown. It is comprised of two stacked LSTM layers (LSTM-1 & LSTM-2) each with 64 hidden nodes. At the output layer, it outputs two weighted MDNs. The forward and the backward arrows denote the information flow of the forward and backward iterations over time. Given an input sample sequence $X = \{x_0, .., x_T\}$ of length T to our B-LSTM-MDN model. Where X is comprised of two main sources of information, the trajectory of the VRUs in 2D dimension $(x_{0:T}, y_{0:T})$ and the k-neighbour reward features at each position of this trajectory $(r^k_{0:T})$. Then, the output of the model is the probability distribution over the future trajectory Y of the VRUs.

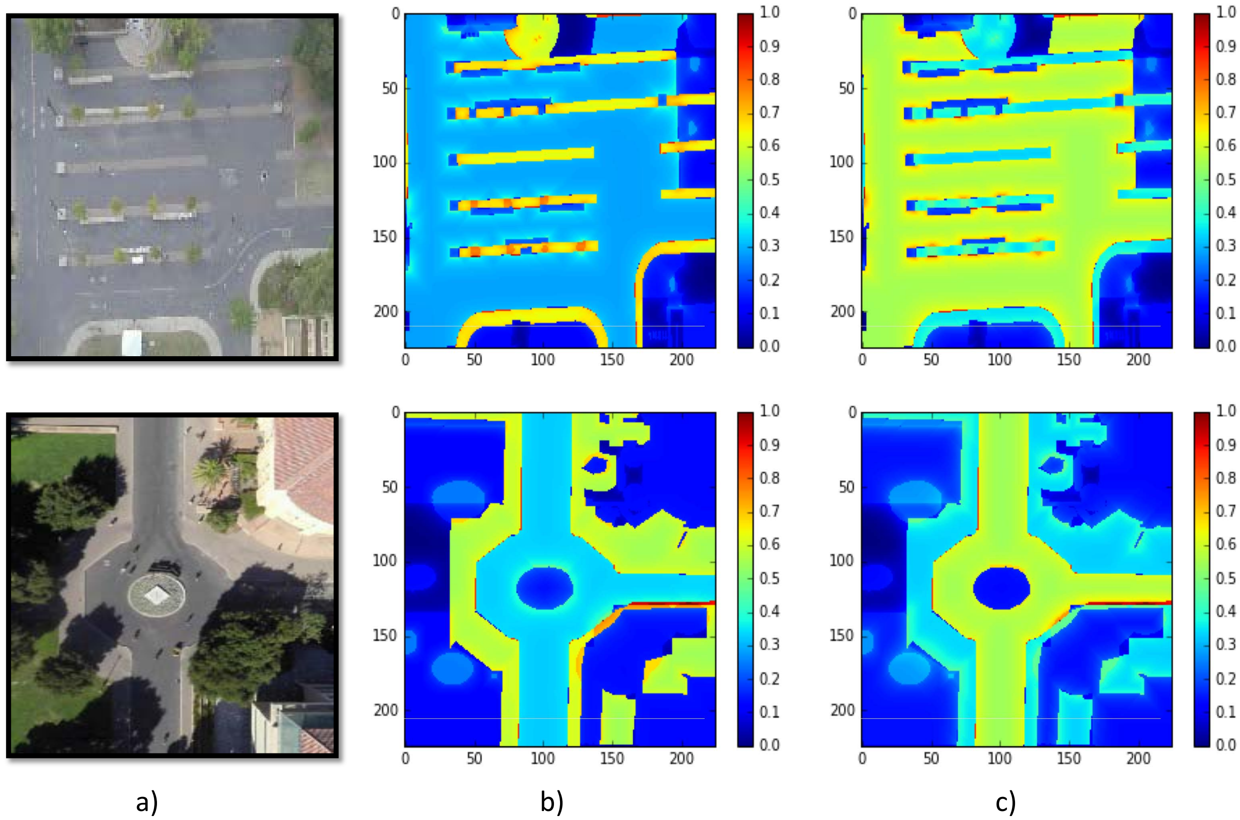

Fig. 2. Sample of the learned reward maps from the Stanford Drone Dataset (SDD) using MaxEnt. a) represents the RGB image of the scene, b) the learned reward for pedestrians and c) the learned reward for cyclists. Highest reward value is 1.0 and the lowest is 0.0.

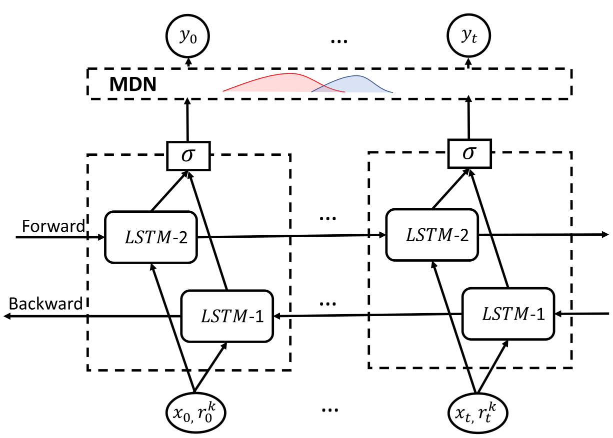

Fig. 3. The probabilistic B-LSTM model for trajectory prediction of VRUs in urban traffic environments.

As we mentioned before, the output h_t of every LSTM memory cell is controlled by three internal gates at each time step t which in the case of our B-LSTM-MDN model will have two of them. The $\overrightarrow{h_t}$ for the forward layer and $\overleftarrow{h_t}$ for the backward layer. In return, the final output y_t from each LSTM cell is as follows:

$$y_t = \sigma(\overrightarrow{h_t}, \overleftarrow{h_t}), \tag{7}$$

where σ is a function to combine the outputs from the two inner LSTMs and it is in our model a concatenation function with the rectified linear unit (ReLU) as the activation layer.

For the MDN output layer, we chose the mixture of Gaussian as our probability density function (PDF), which is calculated as follows:

$$P(y_t|\mathcal{N}_t) = \sum_{m=1}^{M} \alpha_t^m \mathcal{N}(y_t|\mu_t^m, \sigma_t^m, \rho_t^m) \tag{8}$$

where y_t is the real target value, M the number of mixtures for the PDF of Gaussian which was two in our case, α_t^m is the weight for the m-th mixture and $\mathcal{N}$ is the normal Gaussian distribution.

Since the output values from our B-LSTM model are real numbers, so transformations are needed before we use them as the parameters $\{\mu_t^m, \sigma_t^m, \rho_t^m\}$ for our normal distribution as follows:

$$\alpha_t^m = \frac{\exp(\tilde{\alpha}_t^m)}{\sum_{i=1}^{M}\exp(\tilde{\alpha}_i^m))}, \tag{9}$$

$$\sigma_t^m = \exp(\tilde{\sigma}_t^m) \tag{10}$$

$$\rho_t^m = \tanh(\tilde{\rho}_t^m) \tag{11}$$

where $\tilde{\alpha}_t^m$, $\tilde{\sigma}_t^m$ and $\tilde{\rho}_t^m$ are the PDF's weight, variance and the correlation values from the B-LSTM output layer of the m-th Gaussian mixture respectively.

Eventually, the training of the B-LSTM-MDN model can be accomplished via minimising the log likelihood of the normal Gaussian distribution against the input real-valued training data as follows:

$$L(X) = \sum_{t=1}^{T} -\log\left(\sum_{m=1}^{M} \alpha_t^m \mathcal{N}(y_t|\mu_t^m, \sigma_t^m, \rho_t^m)\right) \tag{12}$$

where T is the length of the input sequence. For optimising the aforementioned loss function we used, the Adam optimiser with a learning rate of 0.005.

IV. Experiments and Results

Given the nature of the proposed framework which mainly relies on deep sequence prediction model (i.e., B-LSTM), the necessity for a relatively large amount of VRUs' trajectory in traffic environments is essential. In the following, we will describe the two datasets we utilised for training and evaluating the performance of our models. Then, the details of the training procedure for our proposed framework will be described. Finally, the results and the evaluations of our proposed framework will be illustrated.

A. Stanford Drone Dataset

Recently the Stanford drone dataset (SDD), one of the largest datasets for agents behaviour modelling, has been made publicly available [29]. SDD was collected using a bird's eye view camera mounted on a drone hovering over the vicinity of Stanford University campus.

The dataset contains video images with frame by frame bounding-boxes annotations (at roughly frame rate of 28 FPS) for moving targets such as VRUs (pedestrians and cyclists) and cars. SDD was categorised into eight scenes, each with a number of targets annotated videos. In our experiments, we focused on the scenes that had more number of VRUs, which at the same time contain other static or dynamic objects similar to the ones found in urban traffic environments. These traffic objects are such as sidewalks, road/roundabouts, cars, grass, and buildings. Thus, we chose four scenes from the SDD for the training and testing of our framework, namely "bookstores, gates, deathCircle and little". As a first preparation stage, for each VRUs's annotated bounding box coordinates over time in each scene, they were converted into a trajectory of (x, y) positions by calculating the bounding boxes' centre position.

B. Daimler Path Prediction Dataset

Daimler path prediction dataset (DPPD) [6] (shown in Fig. 5), is one of the early datasets introduced in the literature for the problem of intent and trajectory prediction of pedestrians. The dataset consists mainly of total 68 video sequences of crossing scenarios of pedestrians in an urban traffic environment. The dataset was collected using a vehicle-based camera and was annotated with bounding boxes of the pedestrian in the scene. The annotations provided have two forms, one manually, while the other one relied on an object detector based on HoG/linSVM [30]. In our experiments we relied on the annotations from the object detector in order to accommodate its noisy observations as part of our model. Form the total 68 video sequences there are 13 sequences that were captured while the vehicle was stationary and the rest when the vehicle was moving. Since our framework is expecting to have a holistic overview of the environment that the VRUs is interacting with, we further prepared the video sequences as well as the bounding boxes annotations from the dataset to be in a bird's eye view (BEV) perspective using perspective projection technique. We utilised four-pixel positions of the road boundary in the original images from the video sequences to get a warped BEV image as shown in Figure 5-b. Additionally, as the first stage of the proposed framework expects to take as an input semantic labels map of the scene, we manually annotated a reference image from each scene from the 13 video sequences as shown in Figure 5-c.

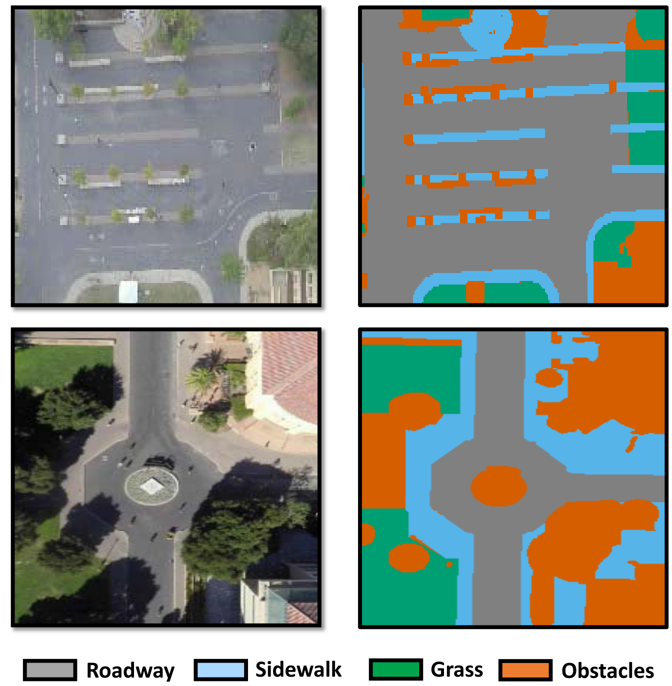

Fig. 4. Semantic labels of two scenes from the SDD namely, bookstore "top" and gates "bottom". The obstacles label is for any obstacle objects such as buildings or trees.

C. Framework Training

In the following, the training data of the two-stages of our framework will be described. The main dataset we will be utilising for training our framework will be the SDD since it contains more data and covers two different VRUs classes namely pedestrians and cyclists. During the testing, we will evaluate our trained framework on both the SDD and the DPPD datasets. More specifically for the DPPD, we will be evaluating the generalisation capabilities of the proposed framework on it since its setup is more similar to the expected one in highly automated vehicles. That being said, since the range of measurements of the VRUs trajectories in both SDD and DPPD is quite different, for the second stage of our framework we will train separate B-LSTM-MDN models for each dataset of them. The first stage, however, will be trained using only the SDD and we will be evaluating its performance

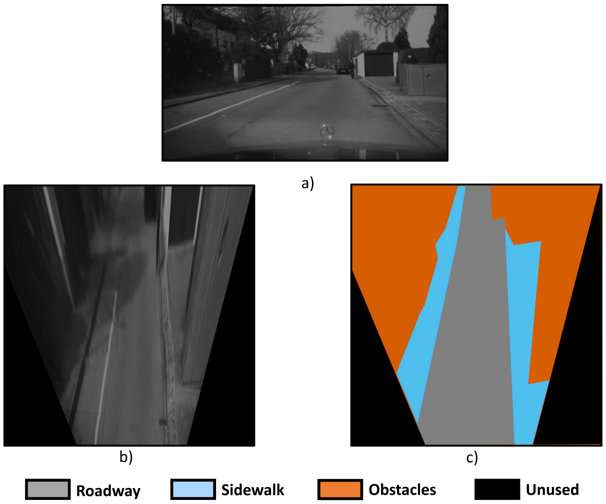

Fig. 5. Sample image from a video sequence from the Daimler path prediction dataset [6]. a) is the original image from the video sequence, b) the transformed BEV image and c) the corresponding input semantic label image for the reward inference.

on the testing split of the SDD and the totally unseen DPPD dataset.

1) Training IRL MaxEnt: For reward learning via MaxEnt sub-system, we have further manually annotated the reference image for each scene from the four scenes with pixel-wise semantic labels as shown in Figure 4. These semantic labels are to be the input feature values vector for the MaxEnt as discussed in Section III-C. The number of pixel-wise semantic labels were scene specific, but the common ones were: buildings, road, sidewalk, and generic obstacles. Since the resolution of pixel-wise semantic label image for each scene is relatively large, so for tractable computation of the MaxEnt algorithm, we resized all of the semantic label images of the four scenes into a size of (224H×224W). For training the MaxEnt, we used the entire VRUs (pedestrians and cyclists) trajectories and semantic label images from each scene from the entire four scenes.

As it can be shown from Figure 2, there are different reward maps for pedestrians and cyclists. Where pedestrians generally prefer to walk on the curbside and that is indicated in Figure 2-b by the warmer (higher reward) colour for curbside. However, since these scenes are inside a university campus, that is why other less warm colours (less reward) also exist in the roadway area. On the other hand, the learned reward maps for cyclists indicate the preference (high reward) of cyclists to avoid obstacles and move on the roadway.

2) Training B-LSTM-MDN: For the probabilistic sequence prediction B-LSTM-MDN sub-system, the entire VRUs' trajectories from the SDD's four scenes were split into 80% for training and the rest for testing using a 2-fold cross-validation technique. As we discussed in Section III-D, the input to the model is a sequence of length T containing past trajectory and reward features. In the case of SDD, we empirically chose T to be of size 28 which corresponds to roughly 1 second of the past trajectory of the pedestrian with its k-neighbour reward features. The value for k was also empirically chosen to be of size 8. Therefore, we preprocessed only the training trajectories split with their 8-neighbourhood learned reward maps at each position of the trajectory into equal chunks of 28 and were used as the input X sequence. For the target Y sequence, at the training phase, it was preprocessed into the same 28 lengths as the input X, but it contained only the future 28 trajectory positions for the trajectory positions of the input sequence X. At the testing phase and with the help of the output MDN layer of the model, we can sample any variable length for the future trajectories. We can do so by recursively feeding in the predicted sequence to the model to generate the next sequence. On the other hand, for the DPPD and since there were no annotations for the ego-motion of the vehicle in order to compensate that for the bounding box measurements of the pedestrians. Thus, we will only utilise the 13 video sequences from the DPPD where the vehicle was stationary in two different physical environments. Using the annotated bounding boxes coordinates we will first pre-process them to obtain its central position in the camera coordinate frame to be the relative position (x, y) of the pedestrian's trajectory in the scene. Since the frame rate of the DPPD is 16 FPS, therefore the input to the B-LSTM-MDN model will be of size 16 which corresponds to 1 second. Similarly the target Y sequence, at the training phase, it is of length 16 lengths as the input trajectory. After the pre-processing of the 13 videos sequences from the DPPD we got total 1300 sequence of size 16 each, we further split the resulted sequences into 60% for training and 40% for testing the B-LSTM-MDN model. It is worth mentioning that 40% of the testing split covers different unique trajectories from those in the training split.

D. Evaluation on SDD

In order to evaluate the performance of our proposed probabilistic framework quantitatively for the VRUs' trajectory prediction problem, we adopted two different evaluation metrics. The first one is the average displacement error which was used in [14]. The average displacement error is essentially the averaged Euclidean distance between the future VRU trajectory predicted and generated by the proposed framework and the future ground truth trajectory. The second metric is the Modified Hausdorff Distance (MHD) which was similarly adopted in [10]. MHD is used to evaluate the geometrical similarities between two non-linear sequences which in our case will be the predicted future trajectory from our framework and the future ground truth trajectory. It is worth noting, that as our framework predicts a probability distribution over each point of the future trajectory. Thus, we will use a random sampling technique to get the real numbers of the future predicted trajectories to evaluate it against the future ground truth trajectory similar to the work in [14].

In order to further evaluate the performance of the proposed framework, in Table I we compare against some variants of data-driven baseline models based on LSTM network as well as planning-based models over two different long-term future prediction horizons (2 seconds and 3 seconds ahead). More specifically, we are interested in exploring whether the learned

TABLE I

PERFORMANCE OF THE PROPOSED FRAMEWORK (B-LSTM-MDN-REWARD) AGAINST A NUMBER OF BASELINE MODELS FOR PEDESTRIANS. THE PROPOSED APPROACH WERE EVALUATED OVER TWO DIFFERENT PREDICTION HORIZONS (2 AND 3 SECS) OF THE PEDESTRIANS' TRAJECTORIES AND AGAINST TWO DIFFERENT EVALUATION METRICS. THE LOWER THE BETTER

Approach	**2.0 (sec) Ahead**		**3.0 (sec) Ahead**	
	Avg. Disp. Error (pixels)	MHD (pixels)	Avg. Disp. Error (pixels)	MHD (pixels)
hMDP [10]	11.54	4.33	8.01	**2.13**
U-LSTM	12.12	10.96	15.16	13.48
U-LSTM-MDN	9.16	7.48	13.49	11.12
B-LSTM-MDN	8.13	6.48	11.29	8.94
U-LSTM-Reward (proposed)	11.49	10.29	15.04	13.26
U-LSTM-MDN-Reward (proposed)	3.22	1.93	4.35	2.72
B-LSTM-MDN-Reward (proposed)	**2.93**	**1.95**	**4.12**	2.90

TABLE II

PERFORMANCE OF THE PROPOSED FRAMEWORK (B-LSTM-MDN-REWARD) AGAINST A NUMBER OF BASELINE MODELS FOR CYCLISTS. THE PROPOSED APPROACH WERE EVALUATED OVER TWO DIFFERENT PREDICTION HORIZONS (2 AND 3 SECS) OF THE CYCLISTS' TRAJECTORIES AND AGAINST TWO DIFFERENT EVALUATION METRICS. THE LOWER THE BETTER

Approach	**2.0 (sec) Ahead**		**3.0 (sec) Ahead**	
	Avg. Disp. Error (pixels)	MHD (pixels)	Avg. Disp. Error (pixels)	MHD (pixels)
hMDP [10]	13.91	8.11	9.56	**5.77**
U-LSTM	14.83	12.05	18.37	15.51
U-LSTM-MDN	10.15	6.78	15.19	10.24
B-LSTM-MDN	9.22	8.68	13.65	8.87
U-LSTM-Reward (proposed)	14.52	13.91	19.12	16.81
U-LSTM-MDN-Reward (proposed)	7.57	6.30	**8.17**	7.38
B-LSTM-MDN-Reward (proposed)	**6.75**	**5.87**	8.22	7.23

reward map features had made an actual difference in the predicted trajectories from our B-LSTM-MDN model. The compared baseline models are:

- hMDP [10]: a planning-based approach based on MDP proposed by Kitani et al. [10]. It also relies on the same MaxEnt paradigm described in Section III-C for learning a reward function of the scene which in addition to the known end-goal of the VRUs, it predicts their trajectories using a Dynamic programming algorithm.
- U-LSTM: traditional U-LSTM model similar to the one introduced in [31], that relies only on the past trajectories of VRUs in order to directly infer real-valued future trajectory.
- U/B-LSTM-MDN: U-LSTM or B-LSTM models with MDN at the output layer (with the same layers as in Section III-D) that relies only on past trajectory positions to infer probability distributions over the future trajectory.
- U-LSTM-Reward: a traditional U-LSTM model but augmented by reward function along with the past trajectories.
- U/B-LSTM-MDN-Reward: is the proposed probabilistic trajectory model described in Section III-D. In the case of U-LSTM-MDN-Reward, the LSTM layers are unidirectional instead of the bidirectional ones.

As it can be noticed from both Table I and Table II, the proposed framework has outperformed all the other LSTM-based and planning-based baseline models in terms of lowest average displacement errors. More specifically,

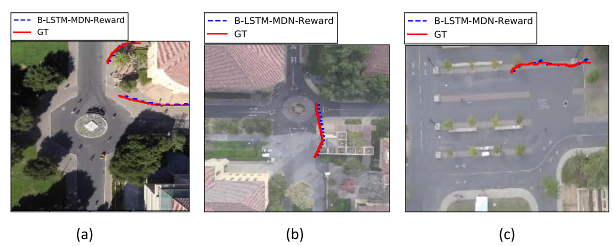

Fig. 6. Qualitative sample predictions of pedestrians from the B-LSTM-MDN-Reward framework (dashed blue) against the ground truth trajectory (solid red) over three scenes of SDD, (a) gates, (b) "deathCircle", (c) "bookstore".

the additional learned reward features were also proven to improve the performance of all the LSTM-based models that did not include them, namely (U-LSTM, U-LSTM-MDN and B-LSTM-MDN). We can notice also from Table I and Table II, both the average displacement error and the MHD scores for the cyclists are little bit higher. The rationale behind that is since cyclists are not constrained by the same rules as pedestrians (i.e. pedestrians usually prefer to walk sidewalk where cyclists are not), as a result this makes the predictions about their trajectories diverges more. For the MHD metric, the hMDP approach has achieved more accurate scores in the longer term prediction horizons (3 secs) because it has a prior knowledge regarding the end goal (ending point of the trajectory) for each VRUs. Another observation is that the LSTM-based models with MDN output layer tend to be giving more accurate predictions in comparison to the

TABLE III

PERFORMANCE OF THE PROPOSED FRAMEWORK (B-LSTM-MDN-REWARD) AGAINST A NUMBER OF BASELINE MODELS OVER THE DAIMLER PATH PREDICTION DATASET [6]. THE LOWER THE BETTER

Approach	**2.0 (sec) Ahead**		**3.0 (sec) Ahead**	
	Avg. Disp. Error (pixels)	MHD (pixels)	Avg. Disp. Error (pixels)	MHD (pixels)
U-LSTM	3.01	2.91	3.36	3.45
U-LSTM-MDN	1.41	1.05	1.50	1.08
B-LSTM-MDN	1.40	1.00	1.62	1.03
U-LSTM-Reward (proposed)	2.88	2.69	3.15	3.09
U-LSTM-MDN-Reward (proposed)	1.43	1.03	1.44	1.06
B-LSTM-MDN-Reward (proposed)	**1.39**	**0.97**	**1.42**	**0.98**

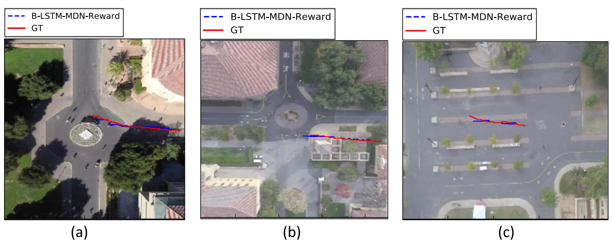

Fig. 7. Qualitative sample predictions of cyclists from the B-LSTM-MDN-Reward framework (dashed blue) against the ground truth trajectory (solid red) over three scenes of SDD, (a) gates, (b) “deathCircle”, (c) “bookstore”.

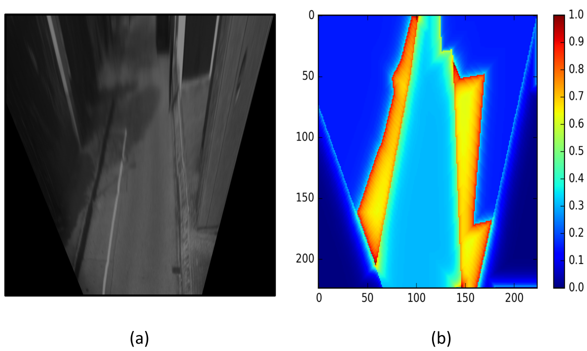

Fig. 8. Sample of the inferred reward map from the Daimler path prediction dataset [6] using the learned weights from the trained IRL-MaxEnt on SDD. a) the input BEV image and b) the inferred reward map. Highest reward value is 1.0 and the lowest is 0.0.

LSTM model that was without it (i.e. U-LSTM). Moreover, the main proposed framework (B-LSTM-MDN-Reward), also proved to be providing resilient results over two long-term prediction horizons (namely 2 and 3 seconds ahead). For an additional qualitative evaluation of the predictions of our proposed framework. In Figure 6 and Figure 7, some predicted trajectories of both pedestrians and cyclists from the proposed framework are plotted against the ground truth trajectories. As it can be shown, the framework can generate trajectories that are close enough to the ground truth trajectories. Moreover, the framework can capture the non-linear motion pattern of the VRUs in traffic environments while generating a collision-free trajectory that mimics the VRUs behaviours.

E. Generalisation on DPPD

In order to evaluate the robustness of our proposed framework, we additionally evaluate it on the DPPD. We leveraged

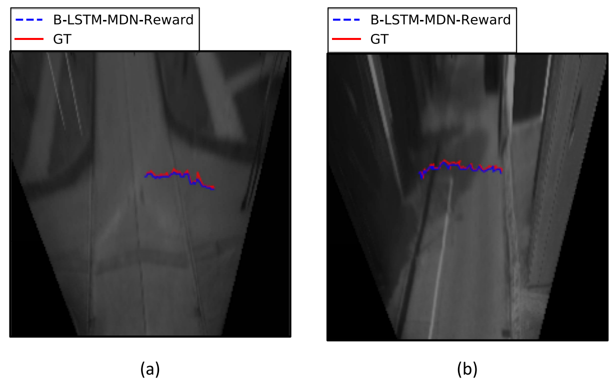

Fig. 9. Qualitative sample predictions of pedestrians from the B-LSTM-MDN-Reward framework (dashed blue) against the ground truth trajectory (solid red) over two different scenes from the Daimler path prediction dataset [6].

the same aforementioned evaluation metrics used for the SDD, namely average displacement error and the MHD. We used the same learned weights from the IRL-MaxEnt model discussed in Section IV-C.1 that was only trained using the scenes and trajectories from the SDD. For the B-LSTM-MDN, we trained it using the DPPD trajectories as described in Section IV-B, following the procedure discussed in Section IV-C.2. In Figure 8, is a sample reward map from the DPPD generated from the trained IRL-MaxEnt model on the SDD. As it can be shown, the inferred reward map is still generalisable despite the fact that the input BEV image is quite different from the BEV from the SDD. In Table III, we compare the performance of the proposed framework on the DPPD against the same baseline approaches we tested on the SDD. The only exception is the hMDP approach which requires prior information about the end goal for the pedestrians which is hard to acquire from the DPPD, whereas in the SDD it was easily obtained when VRUs are getting outside the scene. As it can be noticed the proposed framework continued to be achieving resilient results on the DPPD similar to the SDD. More specifically, the additional information from the reward map was proved to be efficient and generalisable across different unseen datasets. The scores in Table III proves that more quantitatively, where our proposed approaches with the additional reward maps were achieving better results in comparison to their counterparts without the reward maps. In Figure 9, some qualitative results from the proposed framework are shown, where it shows how close were our predicted trajectories to the actual ground truth trajectories.

V. Conclusion

In this work, a framework for long-term prediction of VRUs trajectories in urban traffic environment was proposed. Our proposed framework is based on a combination of planning-based models and sequence prediction models based on inverse reinforcement learning (IRL) and deep recurrent neural networks. With the help of IRL, a reward function of the physical environment can be learned that perfectly capture the VRUs preference in traffic environments. Then using the learned reward function alongside the motion trajectory of pedestrians in the environment, we learn another RNN model that infers a long term trajectory without prior information about the end goal at inference time. We evaluated the proposed framework against two different evaluation metrics and in comparison to other baseline models. Our framework has shown significant improvements over the baseline models in terms of lower average displacement errors and modified Hausdorff distance. Furthermore, we evaluated the generalisation capabilities of our proposed framework on another unseen vehicle-based video sequence dataset, and it continued to show robust results in comparison to the compared baseline approaches. Future directions would be investigating further factors or features to include within our reward function that could influence VRUs behaviours in traffic environments. One example could be the interactions among a group of pedestrians such as family members pedestrians.

References

[1] J. Ziegler *et al.*, "Making Bertha drive—An autonomous journey on a historic route," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 2, pp. 8–20, 2014. doi: 10.1109/MITS.2014.2306552.

[2] B. Huval *et al.*, "An empirical evaluation of deep learning on highway driving," 2015, *arXiv:1504.01716*. [Online]. Available: https://arxiv.org/abs/1504.01716

[3] M. Mara and C. Lindinger, "Talking to the robocar-new research approaches to the interaction between human beings and mobility machines in the city of the future," in *Proc. Post City-Habitats 21st Century Ars Electron. Festival Art Technol. Soc.*, 2015, pp. 86–91.

[4] M. Wagner and P. Koopman, "A philosophy for developing trust in self-driving cars," in *Proc. Road Veh. Automat.*, 2015, pp. 163–171.

[5] D. Rothenbücher, J. Li, D. Sirkin, B. Mok, and W. Ju, "Ghost driver: A field study investigating the interaction between pedestrians and driverless vehicles," in *Proc. 25th IEEE Int. Symp. Robot Hum. Interact. Commun.*, Aug. 2016, pp. 795–802.

[6] N. Schneider and D. M. Gavrila, "Pedestrian path prediction with recursive Bayesian filters: A comparative study," in *Proc. German Conf. Pattern Recognit.* Berlin, Germany: Springer, 2013, pp. 174–183.

[7] C. G. Keller and D. M. Gavrila, "Will the pedestrian cross? A study on pedestrian path prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 2, pp. 494–506, Apr. 2014.

[8] J. F. P. Kooij, N. Schneider, F. Flohr, and D. M. Gavrila, "Context-based pedestrian path prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 618–633.

[9] K. Saleh, M. Hossny, and S. Nahavandi, "Towards trusted autonomous vehicles from vulnerable road users perspective," in *Proc. Annu. IEEE Int. Syst. Conf. (SysCon)*, Apr. 2017, pp. 1–7.

[10] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, "Activity forecasting," in *Computer Vision—ECCV* (Lecture Notes in Computer Science), vol. 7575. 2012, pp. 201–214.

[11] B. D. Ziebart *et al.*, "Planning-based prediction for pedestrians," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 3931–3936.

[12] V. Karasev, A. Ayvaci, B. Heisele, and S. Soatto, "Intent-aware long-term prediction of pedestrian motion," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2016, pp. 2543–2549.

[13] K. Saleh, M. Hossny, and S. Nahavandi, "Intent prediction of vulnerable road users from motion trajectories using stacked LSTM network," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 327–332.

[14] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 961–971.

[15] K. Saleh, M. Hossny, and S. Nahavandi, "Long-term recurrent predictive model for intent prediction of pedestrians via inverse reinforcement learning," in *Proc. Digit. Image Comput., Techn. Appl. (DICTA)*, Dec. 2018, pp. 1–8.

[16] E. A. I. Pool, J. F. P. Kooij, and D. M. Gavrila, "Using road topology to improve cyclist path prediction," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Jun. 2017, pp. 289–296.

[17] E. Rehder and H. Kloeden, "Goal-directed pedestrian prediction," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW)*, Dec. 2015, pp. 139–147.

[18] T. Bandyopadhyay, C. Z. Jie, D. Hsu, M. H. Ang, D. Rus, and E. Frazzoli, "Intention-aware pedestrian avoidance," in *Experimental Robotics*. Heidelberg, Germany: Springer, 2013, pp. 963–977.

[19] B. Völz, K. Behrendt, H. Mielenz, I. Gilitschenski, R. Siegwart, and J. Nieto, "A data-driven approach for pedestrian intention estimation," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 2607–2612.

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

[21] C. M. Bishop, "Mixture density networks," 1994.

[22] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proc. AAAI*, 2008, pp. 1433–1438.

[23] R. Bellman, "A Markovian decision process," *J. Math. Mech.*, pp. 679–684, 1957.

[24] M. Kuderer, H. Kretzschmar, C. Sprunk, and W. Burgard, "Feature-based prediction of trajectories for socially compliant navigation," in *Proc. Robot. Sci. Syst.*, 2012, pp. 1–8.

[25] W.-C. Ma, D.-A. Huang, N. Lee, and K. M. Kitani, "Forecasting interactive dynamics of pedestrians with fictitious play," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4636–4644.

[26] A. Graves, N. Jaitly, and A.-R. Mohamed, "Hybrid speech recognition with deep Bidirectional LSTM," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand.*, Dec. 2013, pp. 273–278.

[27] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4346–4354.

[28] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[29] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 549–565.

[30] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 886–893.

[31] K. Saleh, M. Hossny, and S. Nahavandi, "Intent prediction of pedestrians via motion trajectories using stacked recurrent neural networks," *IEEE Trans. Intell. Veh.*, vol. 3, no. 4, pp. 414–424, Dec. 2018.

Khaled Saleh received the B.Eng. degree in electronic engineering (major in computer science and engineering) from the Faculty of Electronic Engineering, Menoufia University, Egypt, in 2013. He is currently pursuing the Ph.D. degree with the Institute for Intelligent Systems Research and Innovation (IISRI), Deakin University, Australia. His research interests include machine vision, machine learning, and robotics with an emphasis on intelligent autonomous systems.

Mohammed Hossny received the bachelor's degree in computer science from Cairo University and the master's degree in computer science in collaboration with the IBM Centre of Advance Studies (CAS). He joined the Institute for Intelligent Systems and Research Innovation (IISRI), Deakin University, where he developed an algebraic framework for multimodal image fusion as a part of his Ph.D. degree. His research is focused around human performance. He is approaching this from marker less motion capture and biomechanics perspectives.

Saeid Nahavandi received the Ph.D. degree from Durham University, Durham, U.K., in 1991. He is currently an Alfred Deakin Professor, the Pro Vice-Chancellor (Defence Technologies), a Chair of engineering, and the Director of the Institute for Intelligent Systems Research and Innovation, Deakin University, Geelong, VIC, Australia. He has published more than 600 papers in various international journals and conferences. His research interests include modeling of complex systems, robotics, and haptics. He is a fellow of Engineers Australia (FIEAust) and the Institution of Engineering and Technology (FIET). He is the Co-Editor-in-Chief of the IEEE SYSTEMS JOURNAL, an Associate Editor of the IEEE/ASME TRANSACTIONS ON MECHATRONICS, and an Editorial Board member of IEEE ACCESS.