

SFMGNet: A Physics-Based Neural Network To Predict Pedestrian Trajectories

Sakif Hossain¹, Fatema T. Johora¹, Jörg P. Müller¹, Sven Hartmann¹ and Andreas Reinhardt¹

¹Department of Informatics, Clausthal University of Technology, Julius-Albert-Str. 4, 38678 Clausthal-Zellerfeld, Germany

Abstract

Autonomous robots and vehicles are expected to become an integral part of our environment soon. Unsatisfactory issues (esp. for path planning) regarding interaction with existing road users, performance in mixed-traffic areas, and lack of interpretable behavior remain key obstacles. To address these, we present a physics-based neural network, based on a hybrid approach combining a social force model extended by group force (SFMG) with Multi-Layer Perceptron (MLP) to predict pedestrian trajectories considering its interaction with static obstacles, other pedestrians, and pedestrian groups. We quantitatively and qualitatively evaluate the model concerning realistic prediction, prediction performance, and prediction "interpretability". Initial results suggest that, even when solely trained on a synthetic dataset, the model can predict realistic and interpretable trajectories with better than state-of-the-art accuracy.

Keywords

trajectory prediction, trajectory forecasting, hybrid AI, explainable AI

1. Introduction


There has been growing research interest in autonomous technologies like autonomous vehicles, service robots, goods carriers, and surveillance robots. It is generally expected that such autonomous entities will soon be a part of our daily environment. However, before this, several hard issues remain to be resolved. Firstly, their performance in mixed-traffic zones and interaction with existing road users (e.g. cars, pedestrians, cyclists) requires further inspection [1], [2]. Secondly, uncertainty or lack of interpretability in their motion behavior and decisions make them less "socially acceptable" [2]. Again, in a mixed-traffic zone, autonomous robots/vehicles need to predict the future trajectories of other road users (cars, pedestrians, cyclists, etc.) to plan their trajectories and navigate safely. However, doing so is a complex task due to overlapping interactions among road users, road user groups, and static obstacles.


Figure 1 shows an example scenario. Service robot R1 is placed in a mixed-traffic zone, trying

In A. Martin, K. Hinkelmann, H.-G. Fill, A. Gerber, D. Lenat, R. Stolle, F. van Harmelen (Eds.), *Proceedings of the AAAI 2022 Spring Symposium on Machine Learning and Knowledge Engineering for Hybrid Intelligence (AAAI-MAKE 2022)*, Stanford University, Palo Alto, California, USA, March 21–23, 2022.

✉ sakif.hossain@tu-clausthal.de (S. Hossain); fatema.tuj.johora@tu-clausthal.de (F. T. Johora); joerg.mueller@tu-clausthal.de (J. P. Müller); sven.hartmann@tu-clausthal.de (S. Hartmann); andreas.reinhardt@tu-clausthal.de (A. Reinhardt)

ORCID 0000-0003-0800-8392 (S. Hossain); 0000-0003-4565-9645 (S. Hartmann)

 © 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

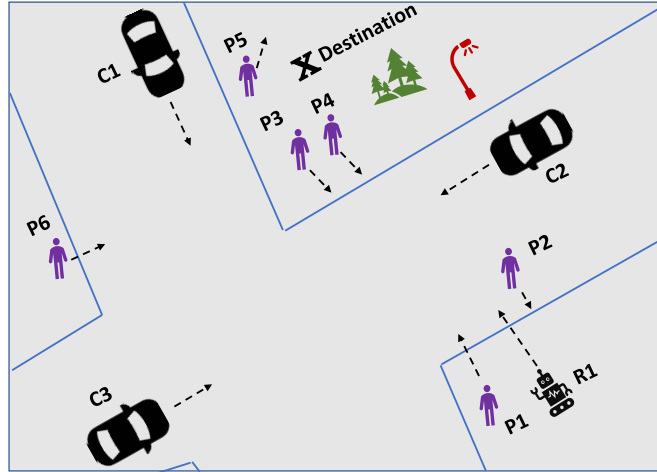


Figure 1: A robot placed in a mixed-traffic zone traveling towards a destination.

to navigate through other road users to reach its destination X. On its path, it will encounter cars (C2), pedestrians (P1, P2, P3, and P4), and static obstacles (e.g. trees, lamps). For that, it may need to predict the trajectories of these other road users and plan its path based on that. However, the behavior of an individual road user can also depend on other factors present in the environment. Factors like interaction with static obstacles, other road users, and (if the road user is traveling in a group) other group members. So, the robot/vehicle has to consider all these factors while predicting other road users' trajectories and planning its path.

As pedestrians are vulnerable road users and form the majority in most mixed-traffic zones, modeling pedestrian behavior warrants high importance. This task entails estimating a pedestrian's future motion within seconds based on its previous motion and environment-related information.

The main contribution of this paper is a novel pedestrian motion prediction model, a prerequisite for autonomous robot/vehicle path planning, which considers pedestrian interaction with static obstacles, other pedestrians, and pedestrian groups. We take a hybrid approach combining the classical social force model (SFM) [3] extended by pedestrian group behavior modeling [4], with a neural network for predicting pedestrian motion.

The remainder of this paper is arranged as follows: Section 2 reviews related work on trajectory prediction. Section 3 then briefly discusses SFMG; our model architecture is described in Section 4, followed by its evaluation in Section 5. Section 6 concludes and discusses future research opportunities.

2. Related work

Broadly, road user trajectory prediction algorithms can be divided into theory-based or physics-based approaches and learning-based approaches. The first group of approaches uses domain/theoretical knowledge and/or physics-based models to model road user motion. The social force model [3] and its various extensions ([6], [7], [8], [4], [9]) are widely used for this purpose.

At its core, SFM defines pedestrian motion as being motivated by external forces exerted upon it by static obstacles and other road users. These methods can model road user motion realistically, and they are interpretable. However, the complexity of estimating model parameters hinders their scalability, and they require too much explicit expert involvement, which in turn restricts their suitability to include new road user types [10].

Learning-based approaches using machine learning algorithms, especially artificial neural networks (ANN) have the potential to overcome the above-mentioned issues. With a sufficient amount of relevant data, a machine learning algorithm can be trained to estimate the underlying motion dynamics of a road user. Different types of neural network architectures have been used to predict pedestrian trajectory, including Multi-layer Perceptron [11], Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) ([12], [13], [14]), Conditional Variational Auto-encoders [15], Generative Adversarial Networks [16], Graph Neural Networks [17], and Graph Attention Networks [18] and so on. However, the size of the neural network needed to estimate complex human behavior and the number of training samples needed jointly become a major bottleneck. Moreover, neural networks tend to be like "black boxes", lacking explainability. Explainability is required for autonomous entities to analyze their behavior, attribute responsibility, and engender trust [19]. The lack of a prior model in an ANN makes explainability hard or nearly impossible. There have been some attempts to combine theory-based and learning-based approaches to avail the advantages of both and consequently design a "Grey-box" model [20].

Johora et al. [21], proposes combining GSFM, an agent-based model, and LSTM-DBSCAN, a learning-based model, in a hierarchical manner, i.e. GSFM takes over when LSTM-DBSCAN predicts conflicting trajectories. Kim et al. [22], combines an ANN with a cellular automata model to estimate traffic congestion. Here, an ANN helps to estimate the cellular automata model's parameters. Karpatne et al. [23] suggests a new paradigm of approaches that aim to combine theory-based knowledge with data-driven approaches and avail subsequent advantages. Antonucci et al. [24], suggests a combination of SFM and MLP to predict pedestrian trajectories. Here, the neural network architecture has been designed in such a manner that it estimates SFM parameters. This work [24] can estimate the acceleration force towards the desired destination and repulsive forces exerted by the nearby obstacles using two different MLPs. Interestingly, the second MLP consists of both the acceleration force part and repulsive force part together in a single network, jeopardizing its interpretable nature. Moreover, the authors do not consider interaction with other road users. Another approach to combining SFM and neural networks to design a differentiable simulation model is introduced in Kreiss et al. [25]. Here, the neural network is used to estimate the interaction potentials. Pedestrian interaction with obstacle/s and other pedestrian/s is considered by the authors. The authors did not test the model [25] on real data. **Noticeably**, none of these works consider interactions among pedestrians and pedestrian group/s, although pedestrian groups contribute toward 70% of the road users [26].

Therefore, in this work we combine SFMG and neural network to model pedestrian motion and, consider interactions with obstacles, other pedestrians, and pedestrian groups. Here, SFMG helps understand and model/predict a pedestrian's motivation behind making a certain motion behavior (i.e. through corresponding forces). The neural network architecture mimics the SFMG equations, and it is solely used to estimate the equation parameters. SFMG serves

as a prior model for the neural network. Thus, the overall approach retains the social force model's interpretability. Again, we employ individual networks to estimate individual forces and combine them in a *modular* manner to get total force. Refer to Section 4 for further details on the model architecture.

3. Social force model with pedestrian groups (SFMG)

In this section, we briefly discuss the social force model [3] to understand how we can incorporate it with Multi-Layer Perceptron (MLP). Suppose, a certain pedestrian α is moving towards its destination \vec{x}_a^o at a certain desired velocity $\vec{w}_\alpha(t) := v_\alpha^o \vec{e}_\alpha(t)$ at time t . Its motion path can be disturbed by static obstacles in the environment (e.g. walls, trees, etc.) and other road users. Then, according to SFM [3], the total force acting upon α is defined as:

$$\frac{d\vec{v}_\alpha(t)}{dt} = \vec{f}_\alpha^o + \sum_{\beta} \vec{f}_{\alpha\beta}^{SOC} + \sum_B \vec{f}_{\alpha B}^{SOC} + \sum_i \vec{f}_{\alpha i}^{SOC} + \xi \quad (1)$$

Again, the acceleration force towards the destination, \vec{f}_α^o is defined as:

$$\vec{f}_\alpha^o = (v_\alpha^o \vec{e}_\alpha(t) - \vec{v}_\alpha(t)) / \tau \quad (2)$$

Here, $\vec{v}_\alpha(t)$ is the current velocity, τ is relaxation time and $\vec{e}_\alpha(t) = \frac{\vec{x}_a^n - \vec{x}_a}{\|\vec{x}_a^n - \vec{x}_a\|}$ is the desired direction. \vec{x}_a^n and \vec{x}_a denote next position and current position of pedestrian α respectively. A pedestrian's tendency to keep a safe distance from static obstacles is denoted by the repulsive force $\vec{f}_{\alpha B}^{SOC}$ as:

$$\vec{f}_{\alpha B}^{SOC} = U_{\alpha B}^o e^{-d_{\alpha B}(t)/R} \vec{\eta}_{\alpha B} \quad (3)$$

Here, B denotes a static obstacle, and $U_{\alpha B}^o$ is the interaction strength between α and obstacle B . $d_{\alpha B}$ and $\vec{\eta}_{\alpha B}$ are the distance and the direction unit vector between α and B respectively. The repulsive force between α and another pedestrian β is denoted by:

$$\vec{f}_{\alpha\beta}^{SOC} = V_{\alpha\beta}^o e^{-d_{\alpha\beta}(t)/\sigma} \vec{\eta}_{\alpha\beta} F_{\alpha\beta} \quad (4)$$

where, $d_{\alpha\beta}$ and $\vec{\eta}_{\alpha\beta}$ are distance and unit direction vector between α and β . $F_{\alpha\beta}$ represents the anisotropic behavior of a pedestrian, and it is defined as:

$$F_{\alpha\beta} = \lambda_\alpha + (1 - \lambda_\alpha)(1 + \cos(\phi_{\alpha\beta})) / 2 \quad (5)$$

Here, λ_α is a constant and $\phi_{\alpha\beta}$ is the angle between the motion direction of pedestrian α and the vector pointing in the direction of α to β . i.e. $\cos(\phi_{\alpha\beta}) = \vec{\eta}_{\beta\alpha}(t) \cdot \vec{e}_\alpha(t)$. Combining equation 4 and 5, and simplifying it, we get:

$$\vec{f}_{\alpha\beta}^{SOC} = V_{\alpha\beta}^o (A_1 e^{-d_{\alpha\beta}(t)/\sigma} \vec{\eta}_{\alpha\beta} + A_2 \vec{\eta}_{\alpha\beta} \vec{\eta}_{\alpha\beta} \vec{e}_\alpha(t) e^{-d_{\alpha\beta}(t)/\sigma}) \quad (6)$$

where, $A_1 = \lambda_\alpha + (1 - \lambda_\alpha)/2$ and $A_2 = (1 - \lambda_\alpha)/2$ are constants.

The attractive force towards points of interest (e.g. friends, street artists, etc.) is given by $\vec{f}_{\alpha i}^{SOC}$. We ignore this force in our work to avoid further complexity. SFM has been extended by [4], [27] to include pedestrian group interaction by introducing the group force \vec{f}_{group} as:

$$\vec{f}_{group} = \vec{f}_{vis} + \vec{f}_{att} \quad (7)$$

In the above equation, \vec{f}_{vis} denotes the pedestrian's desire to keep other group members within his/her field of view (FOV). \vec{f}_{att} is an attraction force representing the pedestrian's motivation to maintain group coherence. The terms \vec{f}_{vis} and \vec{f}_{att} can be described as:

$$\vec{f}_{vis} = S_{vis} * \theta * \vec{V}_{desired} \quad (8)$$

$$\vec{f}_{att} = \begin{cases} S_{att} * \vec{n}(A_{ij}, C_i), & \text{if } dist(A_{ij}, C_i) \geq d \text{ and } \vec{V}_{desired} \neq 0 \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

Here, θ is the minimum rotation angle. S_{vis} and S_{att} are global strength parameters. $\vec{V}_{desired}$ is the desired velocity of any pedestrian A_{ij} in group G_i . $\vec{n}(A_{ij}, C_i)$ represents the normalized unit length vector between A_{ij} and group centroid C_i . Adding \vec{f}_{group} and omitting $\vec{f}_{\alpha i}^{SOC}$ equation 1 becomes:

$$\frac{d\vec{v}_{\alpha}(t)}{dt} = \vec{f}_{\alpha}^o + \sum_{\beta} \vec{f}_{\alpha\beta}^{SOC} + \sum_B \vec{f}_{\alpha B}^{SOC} + \vec{f}_{group} + \xi \quad (10)$$

We use the force equations of the social force model with pedestrian group behavior modeling described in this section to design our pedestrian motion prediction model architecture.

4. Physics-based neural network (SFMGNet)

We aim to design our neural network based framework in such a way that it estimates the total force acting upon a pedestrian according to Equation 10. Equation 10 aggregates four individual forces, namely acceleration force (\vec{f}_{α}^o), repulsion from boundaries ($\vec{f}_{\alpha B}^{SOC}$), repulsive force from other pedestrians ($\vec{f}_{\alpha\beta}^{SOC}$) and group force (\vec{f}_{group}). We employ four individual networks to estimate these four forces separately. We call these individual networks as *modules*. A final network combines these individual modules to give final force ($d\vec{v}_{\alpha}(t)/dt$). Figure 2 shows the overall model architecture and highlights the individual modules. The model specific hyperparameters (e.g. activation function, learning rate, etc.) were chosen empirically. Further details are presented in Section 4.2 and Section 4.3.

4.1. Problem statement

We formulate our trajectory prediction problem similar to standard approaches used in the literature [17], [28]. Given the trajectories of all pedestrians in a scene, we need to predict their future trajectories. Let $\mathbf{X} = [X_1, X_2, \dots, X_N]$ be the current trajectories of all pedestrians present in the scene. We need to predict future trajectories, $\hat{Y} = [\hat{Y}_1, \hat{Y}_2, \dots, \hat{Y}_N]$. Here, for a certain pedestrian α , the given trajectories are $X_{\alpha} = [x_{\alpha}^t, y_{\alpha}^t]$ at time-steps $t = 0, 1, 2, \dots, T$. The real or ground truth future trajectories for α are $Y_{\alpha} = [x_{\alpha}^t, y_{\alpha}^t]$ at time-steps $t = T + 1, \dots, T_e$.

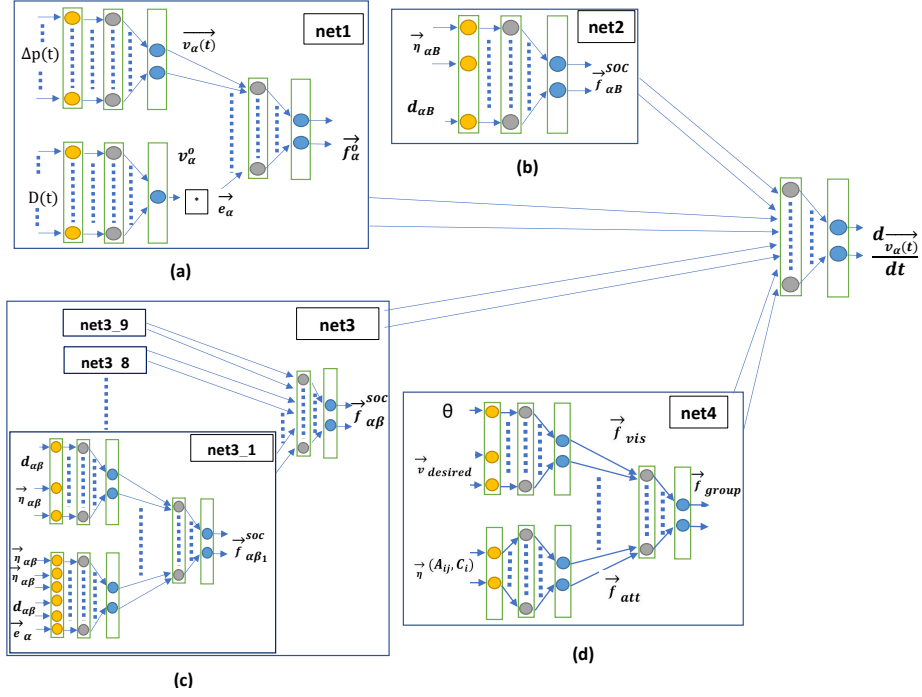


Figure 2: SFGNet architecture. Here, net1 (a), net2 (b), net3 (c) and net4 (d) are the individual networks corresponding to the forces in Equation 10. Note, net3 contains nine instances like net3_1.

4.2. Individual modules

net1: Acceleration This module estimates the goal-directed acceleration force (\vec{f}_α^o) as defined by Equation 2. The model architecture is designed according to Equation 2. This module is largely based on [24]. The module inputs are n previous positions (p) of the pedestrian α , till the current time-step t . The position values are normalized by subtracting the first position value of the window ($p^{(t-n)}$) to avoid spatial bias. The new position values are termed as $\Delta p(t)$.

net1 consists of two parts: one sub-network to estimate the instantaneous velocity $\vec{v}_\alpha(t)$ and second sub-network estimates the desired velocity $v_\alpha^0 \vec{e}_\alpha(t)$. The inputs to the networks are $\Delta p(t)$ and $D(t)$ respectively. Here $D(t)$ refers to the instantaneous velocity magnitudes, and it is calculated as: $D(t) = [||\Delta p'(t)_1||, \dots, ||\Delta p'(t)_{n-1}||]$. Where, $\Delta p'(t) = \Delta p(t) - \Delta p(t-1)$. The first sub-network consists of two MLPs, where the first MLP is given *sigmoid* activation and the second MLP rescales the outputs. The second sub-network also has two MLPs in it, with *tanh* activation for the first MLP and second MLP rescales the outputs. This output is multiplied with goal directed unit vector, $\vec{e}_\alpha(t)$, which is estimated by the approach described in Section 4.4. Finally, the two sub-networks outputs are aggregated by another MLP to give \vec{f}_α^o . Figure 2 (a) shows the net1 architecture. The net1 architecture can be expressed as:

$$\vec{f}_\alpha^o = sig(D(t)W_{vd})W_{vds} \vec{e}_\alpha(t) - tanh(\Delta p(t)W_{vi}) \cdot W_{vis} \quad (11)$$

Here W_{vd} , W_{vds} , W_{vi} and W_{vis} are weight matrices. This net1 architecture mimics Equation 2.

net2: Repulsive force from static obstacles This network is responsible for estimating the repulsive force from obstacles [24]. In this work, we only consider the nearest obstacle point. This force is defined by Equation 3. So, net2 architecture is designed according to this equation. Figure 2 (b) shows net2 architecture. The distance between pedestrian α and obstacle B ($d_{\alpha B}$), and the direction vector pointing towards B to α ($\vec{\eta}_{\alpha B}$) are network inputs. net2 consists of two MLPs, where the first one is followed by a *sigmoid* activation and another MLP rescales the outputs. net2 architecture can be expressed as:

$$\vec{f}_{\alpha B}^{SOC} = sig(W_U e^{d_{\alpha B}/W_R} \vec{\eta}_{\alpha B}) \cdot W_{fB} \quad (12)$$

Here, W_U , W_R and W_{fB} are weight matrices. This structure mimics SFMG force Equation 3.

net3: Repulsive force from other pedestrians This section focuses on estimating the repulsive forces experienced by a pedestrian from other pedestrians. Equation 6 defines this force. So, this corresponding network aims to mimic this equation. The input parameters are $\vec{\eta}_{\alpha\beta}$, $d_{\alpha\beta}$ and \vec{e}_{α} . A_1 , A_2 , σ and $V_{\alpha\beta}^o$ are the learn-able parameters. Figure 2 (c) shows the network architecture used to estimate repulsive force exerted by one neighboring pedestrian. This network consists of two sub-networks. First sub-network takes $\vec{\eta}_{\alpha\beta}$ and $d_{\alpha\beta}$ as inputs and it has one MLP which is followed by *relu* activation. This subsection estimates the first part of Equation 6. Second sub-network estimates second part of Equation 6 with inputs: $\vec{\eta}_{\alpha\beta}$, $d_{\alpha\beta}$ and \vec{e}_{α} . It also has similar architecture as the first sub-network. Finally, outputs of these two sub-networks are combined by another MLP, to get $\vec{f}_{\alpha\beta}$ force. The architecture is expressed as:

$$\vec{f}_{\alpha\beta}^{SOC} = (relu(W_{A1} \exp [d_{\alpha\beta}(t)/W_{\sigma}] \vec{\eta}_{\alpha\beta}) + relu(W_{A2} \vec{\eta}_{\alpha\beta} \vec{\eta}_{\alpha\beta} \vec{e}_{\alpha}(t) \exp [-d_{\alpha\beta}(t)/W_{\sigma 2}])) \cdot W_{\alpha\beta} \quad (13)$$

Here, W_{A1} , W_{A2} , W_{σ} , $W_{\sigma 2}$ and $W_{\alpha\beta}$ are weight matrices. This structure mimics Equation 6.

Again, one pedestrian can have multiple nearby pedestrian exerting repulsive force on it. So, we empirically consider a maximum of nine nearby pedestrians for a certain pedestrian. We employ instances of the network described above for finding repulsive forces from them. These networks are named as net3_i, where $i = 1, 2, 3, \dots, 9$. A final network, net3 is used to combine the outputs of these networks to get total repulsive force from other pedestrians.

net4: Group force net4 approximates the force exerted by pedestrian groups. The group force is given by Equation 7. Group force has two parts: visibility force (f_{vis}) and attraction force (f_{att}). So, net4 also has two sub-networks to estimate these two forces. Figure 2 (d) shows the net4 architecture. The first network inputs are θ and $\vec{v}_{desired}$. It has two MLPs: the first MLP is followed by *relu* activation, and second MLP rescales the *relu* output. The second sub-network has similar architecture, with $\vec{n}(A_{ij}, C_i)$ as input. The two outputs are combined by another MLP to estimate the final group force. This can be expressed as:

$$\vec{f}_{group} = (relu(W_{g1} \theta \vec{V}_{desired}) + relu(W_{g2} \vec{n}(A_{ij}, C_i))) \cdot W_G \quad (14)$$

Here, W_{g1} , W_{g2} and W_G are weight matrices. This structure is similar to Equation 7.

4.3. Module: Final recombination

This final recombination module aggregates the outputs of net1, net2, net3 and net4 to predict total force $\frac{d\vec{v}_\alpha(t)}{dt}$. This network consists of two MLPs. First MLP is followed by *relu* activation, and second MLP rescales this output. Figure 2 shows the overall architecture. Expressed as:

$$\frac{d\vec{v}_\alpha(t)}{dt} = \text{relu}\left(\underbrace{\vec{f}_\alpha^o}_{net1} + \underbrace{\vec{f}_{\alpha B}^{SOC}}_{net2} + \underbrace{\sum_{\beta} \vec{f}_{\alpha\beta}^{SOC}}_{net3} + \underbrace{\vec{f}_{group}}_{net4}\right) W_{IF} \cdot W_{FF} \quad (15)$$

W_{IF} and W_{FF} are weight matrices. Note, as we consider only the nearest obstacle point, we do not need to sum repulsive forces from static obstacles.

As we compare, the overall architecture described by Equation 15 and portrayed in Figure 2, strongly emulates the complete SFMG force described by Equation 10.

4.4. Goal prediction

An estimated goal or destination is required as a goal-directed unit vector (\vec{e}_α) is an input for net1 and net3. In an environment with other pedestrians and static obstacles, a pedestrian's previous positions cannot be exploited to estimate \vec{e}_α . So, we use Multiple Model Approach (MMA) [29] to estimate it. Specifically, we use Interaction Multiple Model (IMM) estimators. Different goal hypotheses based Kalman filters are chosen as filters for the IMM estimator. The goal hypotheses were generated using the Constant turn rate and velocity (CTRV) [30] model. Then, the IMM estimator compares the different filter-generated trajectories with the observations and selects the most likely one.

5. Evaluation

In this section, we perform a set of experiments to qualitatively and quantitatively evaluate SFMGNet in terms of realistic (Section 5.4), interpretable (Section 5.5) and accurate (Section 5.6 and Section 5.7) trajectory prediction. We discuss the experimental setup, dataset details and feature extraction process in Sections 5.1, 5.2 and 5.3 respectively.

5.1. Experimental setup

We train our model using a synthetic dataset to demonstrate our model's performance in cases where the relevant data is scarce or unavailable. Only the models trained with synthetic dataset were used during the evaluation process, even while testing on real-world datasets. We create the synthetic dataset using the trajectories simulated by an SFMG (SFM extended with group force) based simulator ¹ in a closed environment (crossing passageways). We run 1000 simulations of 30-second duration sampled at 0.1 seconds. The simulation run scenarios were chosen randomly

¹SFM: <https://github.com/svenkreiss/socialforce/tree/579543a3abe22716835acfa0b7d2d57fc1c199b6>. Extended for groups.

following a discrete uniform distribution. The following conditions were chosen to simulate scenarios rich with different types of interactions. The pedestrians' starting positions were chosen randomly among four possible starting zones within 7 to 10 meters distance to the destination point. The number of pedestrians in a certain scenario was also chosen randomly between 2 to 10. The presence of a group of sizes 2 to 4 is chosen randomly. Relaxation time $\tau = 0.5$, is constant. We divide the dataset with sample randomization into three sets: 50% for the training set, 25% for the development set, and 25% for the testing set.

We train the models net1, net2, net4, net3, and individual instances of net3 for different neighboring pedestrians (net3_1, net3_2, ..., net3_9) with an Adam optimizer with a learning rate of 0.001 and batch size 16. The number of epochs was chosen based on model performance on the development set. The final recombination layer was trained by Adam with a 0.1 learning rate. The window of motion observation is empirically chosen as $n = 10$. We train and optimize the model based on corresponding mean squared errors (MSE). The final MSE values for net1, net2, net3, net4 and SFMGNet are 0.0066, 0.5295, 0.1594, 0.0181 and 0.2640 units respectively.

5.2. Dataset overview

To assess SFMGNet's performance quantitatively, we test its performance on two benchmark datasets namely *ETH* [31] dataset and *UCY* [32] dataset. The *ETH* dataset consists of two scenes: *ETH* and *Hotel*. *ETH* scene consists of 360 pedestrians and 61 pedestrian groups. The hotel scene has 389 pedestrians and 41 groups. The *UCY* dataset has five scenes: namely *zara01*, *zara02*, *students01*, *students03* and *uni_examples*. *zara01* has 148 scenes and 45 groups. *Zara02* consists of 204 pedestrians and 58 groups. *students03* scene consists of 434 pedestrians and 104 groups. *students01* consists of 414 pedestrians and *uni_examples* has 118 pedestrians. *students01*, *students03*, and *uni_examples* scenes together are called *UNIV*. All the datasets are sampled at 0.4 seconds. All these scenes consist of multiple pedestrians being present in the same frame. We use the group and destination-related information given by the work in [33].

5.3. Feature extraction

For a certain pedestrian α at time-step t , the SFMGNet model requires: distance and direction unit vector from the nearest obstacle ($d_{\alpha B}$ and $\vec{\eta}_{\alpha B}$), distance and direction unit vector towards nearest (up to nine) pedestrian β ($d_{\alpha\beta}$ and $\vec{\eta}_{\alpha\beta}$). If in a group, then it needs a group Centroid directed unit vector ($\vec{\eta}(A_{ij}, C_i)$) and minimum rotation angle (θ). However, a trajectory dataset normally consists of time-step, velocity, and positions. Additionally, information about group formation [33] and obstacles are provided (or extracted). So, we follow the approach depicted in the flowchart in Figure 3 to extract features. A dataset (D), group (GroupList), and obstacle (ObstacleList) information are given to the algorithm. Then, at every time-step t in the dataset, we follow the following steps. First, the nearest obstacle is taken from ObstacleList, distance from pedestrian and $\vec{\eta}_{\alpha B}$ is calculated and stored in an array (ObstacleSet). Next, we check for multiple pedestrians' presence. If true, we treat as neighbors to ego pedestrian (α) and, calculate and store $d_{\alpha\beta}$ and $\vec{\eta}_{\alpha\beta}$ in NeighbourList. Next, we check the existence of groups from GroupList. If true, first we get the corresponding group member positions. Now, we need $\vec{\eta}(A_{ij}, C_i)$ and θ . For calculating $\vec{\eta}(A_{ij}, C_i)$, first, we find the group Centroid based on

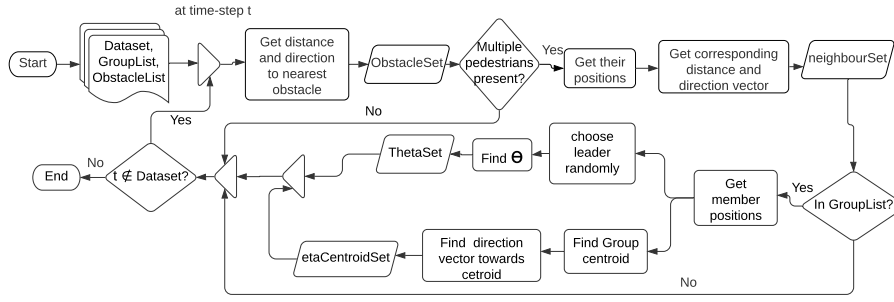


Figure 3: Flowchart for extracting features from a Dataset, given a list of groups and obstacles.

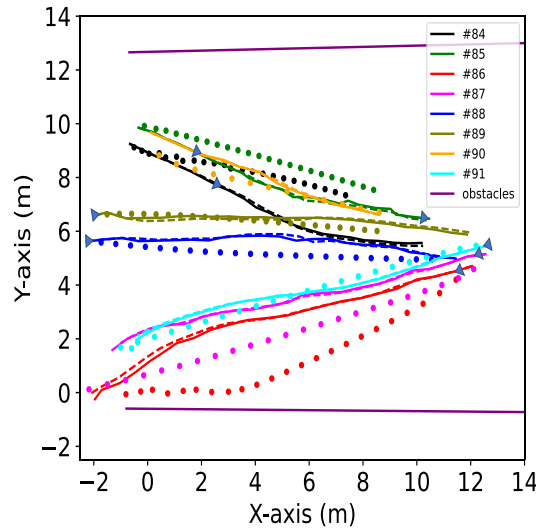


Figure 4: Real (solid lines), predicted (dashed lines), and SFMG simulated (dotted lines) trajectories of eight pedestrians. Here, arrows indicate the motion directions of the pedestrians.

group member positions and find their corresponding $\vec{\eta}(A_{ij}, C_i)$. Next, we store it in an array (etaCentroidList). For θ , first, we randomly choose a group leader, then we calculate θ based on the group leader and other member positions (see [4], [27]), and store it in an array (ThetaSet).

5.4. Realistic trajectory prediction

This section aims to evaluate SFMGNet’s ability to predict realistic trajectories through a case study. For this, we choose a sample scenario (time-step: 4630 - 4880) from the ETH scene from ETH dataset [31] with multiple numbers of pedestrians and pedestrian groups. This scenario consists of eight pedestrians namely: 84, 85, 86, 87, 88, 89, 90 and 91. Here, 86, 87, and 91 are traveling from right to left direction. And other pedestrians are traveling in the opposite direction. 84 and 85 are in a pedestrian group. Similarly, 88 and 89 are also in a group. We

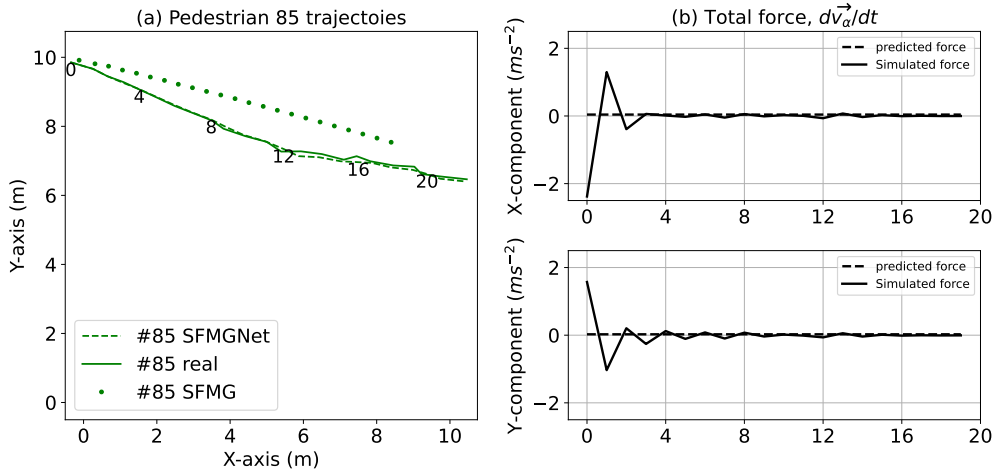


Figure 5: (a) Real, predicted and SFMG simulated trajectories of a pedestrian. Here, the numbers indicate the corresponding time steps. (b) Total predicted force acting on the pedestrian.

predict all the pedestrian trajectories with SFMGNet by taking 1.2 seconds of previous motion information as input and predicting the next 4.8 seconds of the trajectory (if possible) at a time. We also simulate their trajectories with SFMG [3] [4] for comparison. Figure 4 shows the real (solid lines), predicted (dashed lines) and SFMG simulated (dots) trajectories. Arrows show the corresponding pedestrian’s motion direction. We see that predicted trajectories are similar to the real pedestrian trajectories, barring a couple of time steps. The SFMGNet can predict and model slight deviations in every pedestrian’s motion accurately. Conversely, SFMG simulated trajectories are flatter, not covering the influences of environment related influences on the pedestrian. Also, SFMG simulated pedestrian trajectories fail to reach or overtake the destination point. As an example, we take a look at 85 in Figure 5 (a). The numbers represent the time steps. Our model predicted trajectory matches with the real trajectory and reaches the destination. Whereas, SFMG simulated trajectory is flatter and fails to reach the destination.

5.5. Interpreting trajectory prediction

In this section, we aim to interpret the SFMGNet predicted trajectories to a certain degree as a proof of concept. For that, we look at the predicted and simulated forces acting upon a certain pedestrian and try to establish a causal relationship with the environment elements/influences. We choose pedestrian 85 for this evaluation. Figure 4 shows the overall scenario, Figure 5 zooms into 85 trajectories (a) and, predicted and SFMG simulated total force (b). Looking into the predicted total force (Figure 5 (b)), we see an almost constant force acting upon the pedestrian, approximately 0.01 units on X-axis and 0.01 units on Y-axis. It points to the idea, that the pedestrian maintains a rather stable trajectory with subtle direction changes in its journey. The SFMG simulated force also remains stable apart from a spike at the beginning. However, the total force does not paint the whole picture. So, we explore the individual forces.

Figure 6 shows the individual forces, namely: acceleration force (a), the repulsive force from boundaries (b), repulsive from other pedestrians (c), and group force (d). The predicted attraction

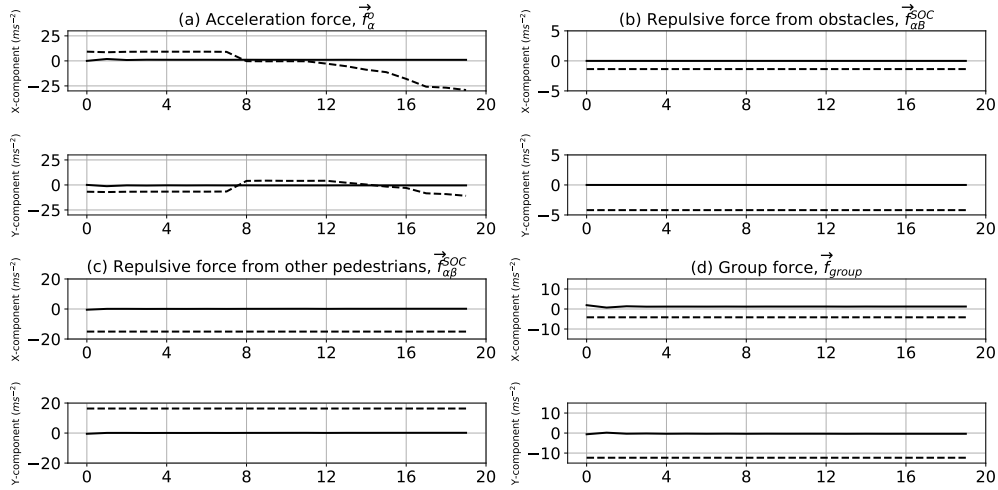


Figure 6: Predicted (dashed line) and simulated (solid line) acceleration force (a), repulsive force from obstacles (b), repulsive forces from other pedestrians (c) and group force (d) acting upon pedestrian 85.

force (Figure 6 (a)) changes over time, starting at positive values, falling towards zero at around time-step 12, and continues falling afterward. This could point toward the pedestrian’s need to change its velocity overtime to compensate for other environmental facts. That is, it would feel like a constant acceleration force in an open environment with no other road users and obstacles. But it’s not the case here. However, the SFMG simulated acceleration force remains stable throughout. Next, we look at the forces exerted by other actors; static and dynamic. Figure 6 (b) shows that, the pedestrian feels a constant repulsive force (both predicted and simulated) from boundaries throughout its journey. This corresponds to its desire to keep a safe distance from boundaries. As the nearest boundary point is quite far from it, the force values are low. Again, the pedestrian is surrounded by seven other pedestrians, so, it should feel a strong cumulative repulsive force from other pedestrians. This behavior is emulated by the predicted repulsive force from other pedestrians shown in Figure 6 (c). However, SFMG fails to emulate this properly, as it simulates force values nearing zero units. Moreover, 85 is traveling in a group with 84. Therefore, it feels like group forces influencing it, as we see in Figure 6 (d). The SFMGNet predicted group forces are much higher (in magnitude) compared to SFMG simulated group forces. The predicted group force better represents 85’s desire to stay closer to 84. The nature of group force depends on the combination of the visibility force, f_{vis} and attraction force towards group centroid, f_{att} . But, we do not consider that level of granularity in this work to avoid further complexity. To conclude, we can establish a high-level causality behind a pedestrian’s motion behavior motivation by inspecting individual force predictions.

5.6. Quantitative comparison

We quantitatively assess our model based on two commonly used [13], [28] distance-based parameters: average displacement error (ADE) and mean final displacement error (FDE). ADE refers to the mean of all euclidean distance values between the real and predicted values at each

Table 1

ADE/MDE error (in meters) comparison between SFMGNet and other baseline models.

Metric	Dataset	CV	FF	S-LSTM	S-GAN	SFM-NN	SFMGNet
ADE/FDE	Hotel	0.27/0.51	1.59/3.12	0.15/0.33	0.72/1.61	0.36/0.82	0.07/0.11
	ETH	0.58/1.15	0.67/1.32	0.60/1.31	0.81/1.52	0.68/1.63	0.10/0.14
	UNIV	0.46/1.02	0.69/1.38	0.52/1.25	0.60/1.26	0.46/1.12	0.35/0.64
	zara01	0.34/0.76	0.39/0.81	0.43/0.93	0.34/0.69	0.35/0.85	0.11/0.17
	zara02	0.31/0.69	0.38/0.77	0.51/1.09	0.42/0.84	0.38/0.95	0.13/0.23
Average		0.39/0.83	0.74/1.48	0.44/0.98	0.58/1.18	0.45/1.07	0.15/0.26

time-steps. FDE is the distance between the last predicted position and the last real position. The lower the values of both ADE and FDE parameters, the better the model performance.

For comparison, we choose constant velocity model (CV), a MLP or feed-forward network-based model [11] (FF), S-LSTM model [13], S-GAN model [16] and the model described in [24] (SFM-NN) as baselines. The baseline models (except SFM-NN) take 3.2 seconds (8 time-steps) of trajectory values as input and predict the next 4.8 seconds of trajectory, and report error values. They are also trained on the corresponding real-world datasets. The SFM-NN is trained on a synthetic dataset and takes 1 second of data as input to predict 4.8 seconds of trajectory. Our model (SFMGNet) is trained solely on a synthetic dataset. It takes 1.2 seconds (as $n = 10$ chosen) of data as input and predicts the next 4.8 seconds of data. Then, we calculate the error values. The error values are reported in Table 1. The bold values represent the lowest or best error values. As we can see, our model SFMGNet outperforms every baseline model in every scene dataset. Especially, the lowest ADE and FDE values (i.e. **0.07** and **0.11** respectively) are found for *hotel* scene data in ETH dataset. Our model shows the worst performance on UNIV dataset: ADE = **0.35** and FDE = **0.64**. Still, the model performance is noticeably better than the baseline models. The average ADE and FDE values are: **0.15** and **0.26**.

We evaluate SFMGNet’s ability to predict feasible and realistic trajectories by calculating the evaluation metric introduced in [34]. That is, we calculate the percentage of *near-collisions* among pedestrians for each frame or time-step. A near-collision takes place when the Euclidean distance between the pedestrians is below 0.1 meters [34]. We compute the average percentage of human near-collision in each frame of ETH and UCY datasets. Our model (SFMGNet) predicted trajectories do not contain any near-collisions in both datasets. That is, SFMGNet predicts 0.0% near-collisions for all frames in ETH, ETH hotel, UNIV, zara01, and zara02 scenes. This in turn strongly indicates that SFMGNet can predict feasible and realistic trajectories.

5.7. Ablation study

To assess the impact of different modules in our proposed model framework, i.e. acceleration force, the repulsive force from static obstacles, the repulsive force from other pedestrians, and group force, we test the performance of several ablative models. We choose three different model versions, namely: Att, ABr, and ABrPr for this study. Where the model Att considers only acceleration force towards the destination, model ABr considers both acceleration force and repulsive force from static obstacles, and model ABrPr considers all forces except group force. Finally, we compare them with the complete framework, SFMGNet. We test them on ETH

Table 2

Evaluation of the ablative models and the proposed model (SFMGNet) based on ADE/MDE (in meters).

Metric	Dataset	Att	ABr	ABrPr	SFMGNet
ADE/FDE	Hotel	0.63/1.11	0.13/0.23	0.13/0.21	0.07/0.11
	ETH	2.92/5.42	3.23/5.51	2.14/4.06	0.10/0.14
	UNIV	1.12/2.16	0.37/0.67	0.37/0.67	0.35/0.64
	zara01	0.90/1.71	0.20/0.34	0.15/0.25	0.11/0.17
	zara02	0.93/1.74	0.23/0.41	0.15/0.25	0.13/0.23
	Average	1.30/2.43	0.83/1.43	0.59/1.09	0.15/0.26

and UCY datasets and report the corresponding ADE/FDE error values in Table 2. We can see, the model performance improves with the addition of each force estimation module. Model Att performs the worst across all datasets, and performance increases significantly when repulsive force from obstacles is considered in model ABr. This trend follows for model ABrPr, which considers repulsive force from other pedestrians. Models Att, ABr, and ABrPr perform notably worse on the ETH scene compared to the other scenes. However, SFMGNet outperforms these ablative models significantly, indicating the strong importance of pedestrian groups.

Based on the above quantitative and qualitative evaluation, we conclude that SFMGNet can model realistic, interpretable pedestrian trajectories. Also, it shows better than state-of-the-art performance on two real-world benchmark datasets in terms of distance-based metrics.

6. Conclusion and outlook

This work proposes SFMGNet, a physics-based neural network-based framework to predict pedestrian trajectories while considering the influences of static obstacles, other pedestrians, and pedestrian groups. SFMGNet combines the social force model with group behavior modeling (SFMG) and multi-layer perceptron (MLP). SFMG provides knowledge about the motivation behind pedestrian motion behavior, which is then engineered in the MLP based network’s architecture. This combination allows the MLP based model to inherit the SFMG model’s interpretable nature. This model has the potential to contribute to better planning and understanding motion behavior of robots and autonomous vehicles. Based on the evaluation results, we conclude that the model predicts realistic trajectories and outperforms state-of-the-art models on the ETH and UCY datasets. We can also causally interpret model predictions. However, there remains potential for further improvement. We aim to include other types of road users (e.g., cars, cyclists) in the model to better emulate heterogeneous mixed-traffic zones. Again, we want to explore our model’s potential in modeling new types of road users (e.g., cargo-movers, tram buses), even when relevant data is scarce/absent.

References

- [1] A. Talebpour, H. S. Mahmassani, Influence of connected and autonomous vehicles on traffic flow stability and throughput, *Transportation Research Part C: Emerging Technologies* 71

(2016) 143–163.

- [2] P. Bhavsar, P. Das, M. Paugh, K. Dey, M. Chowdhury, Risk analysis of autonomous vehicles in mixed traffic streams, *Transportation Research Record* 2625 (2017) 51–61.
- [3] D. Helbing, P. Molnar, Social force model for pedestrian dynamics, *Physical review E* 51 (1995) 4282.
- [4] S. Ahmed, F. T. Johora, J. P. Müller, Investigating the role of pedestrian groups in shared spaces through simulation modeling, in: *International Workshop on Simulation Science*, Springer, 2019, pp. 52–69.
- [5] S. Hossain, Predicting pedestrian motion using a physics-based neural network, Master's thesis, Clausthal University of Technology, 2021.
- [6] F. Pascucci, N. Rinke, C. Schiermeyer, V. Berkhahn, B. Friedrich, Should I Stay or Should I Go? A Discrete Choice Model for Pedestrian–Vehicle Conflicts in Shared Space, Technical Report, 2018.
- [7] N. Rinke, C. Schiermeyer, F. Pascucci, V. Berkhahn, B. Friedrich, A multi-layer social force approach to model interactions in shared spaces using collision prediction, *Transportation Research Procedia* 25 (2017) 25 (2017) 1249–1267.
- [8] B. Anvari, M. G. Bell, A. Sivakumar, W. Y. Ochieng, Modelling shared space users via rule-based social force model, *Transportation Research Part C: Emerging Technologies* 51 (2015) 83–103.
- [9] F. T. Johora, J. P. Müller, Zone-specific interaction modeling of pedestrians and cars in shared spaces, *Transportation Research Procedia* 47 (2020) 251–258.
- [10] H. Cheng, F. T. Johora, M. Sester, J. P. Müller, Trajectory modelling in shared spaces: Expert-based vs. deep learning approach, in: *21st International Workshop on Multi-Agent Systems and Agent-Based Simulation (MABS 2020)*, Auckland, NZ, 2020. URL: <https://samarthswarup.github.io/mabs2020/accepted/>.
- [11] C. Schöller, V. Aravantinos, F. Lay, A. Knoll, What the constant velocity model can teach us about pedestrian motion prediction, *IEEE Robotics and Automation Letters* 5 (2020) 1696–1703.
- [12] X. Wang, R. Jiang, L. Li, Y. Lin, X. Zheng, F.-Y. Wang, Capturing car-following behaviors by deep learning, *IEEE Transactions on Intelligent Transportation Systems* 19 (2017) 910–920.
- [13] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, S. Savarese, Social lstm: Human trajectory prediction in crowded spaces, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 961–971.
- [14] H. Cheng, M. Sester, Modeling mixed traffic in shared space using lstm with probability density mapping, in: *21st International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2018, pp. 3898–3904.
- [15] H. Cheng, W. Liao, M. Y. Yang, B. Rosenhahn, M. Sester, Amenet: Attentive maps encoder network for trajectory prediction, *arXiv preprint arXiv:2006.08264* (2020).
- [16] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, A. Alahi, Social gan: Socially acceptable trajectories with generative adversarial networks, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.
- [17] A. Vemula, K. Muelling, J. Oh, Social attention: Modeling attention in human crowds, in: *2018 IEEE international Conference on Robotics and Automation (ICRA)*, IEEE, 2018, pp. 4601–4607.

- [18] Y. Huang, H. Bi, Z. Li, T. Mao, Z. Wang, Stgat: Modeling spatial-temporal interactions for human trajectory prediction, in: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6272–6281.
- [19] F. Pasquale, Toward a fourth law of robotics: Preserving attribution, responsibility, and explainability in an algorithmic society, *Ohio St. LJ* 78 (2017) 1243.
- [20] A. Kroll, Grey-box models: Concepts and application, *New frontiers in computational intelligence and its applications* 57 (2000) 42–51.
- [21] F. T. Johora, H. Cheng, J. P. Müller, M. Sester, An agent-based model for trajectory modelling in shared spaces: A combination of expert-based and deep learning approaches, in: *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, 2020, pp. 1878–1880.
- [22] B. S. Kim, T. G. Kim, Modeling and simulation using artificial neural network-embedded cellular automata, *IEEE Access* 8 (2020) 24056–24061.
- [23] A. Karpatne, G. Atluri, J. H. Faghmous, M. Steinbach, A. Banerjee, A. Ganguly, S. Shekhar, N. Samatova, V. Kumar, Theory-guided data science: A new paradigm for scientific discovery from data, *IEEE Transactions on knowledge and data engineering* 29 (2017) 2318–2331.
- [24] A. Antonucci, G. P. R. Papini, L. Palopoli, D. Fontanelli, Generating reliable and efficient predictions of human motion: A promising encounter between physics and neural networks, *arXiv preprint arXiv:2006.08429* (2020).
- [25] S. Kreiss, Deep social force, *arXiv preprint arXiv:2109.12081* (2021).
- [26] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, G. Theraulaz, The walking behaviour of pedestrian social groups and its impact on crowd dynamics, *PloS one* 5 (2010) e10047.
- [27] A. Kremyzas, N. Jaklin, R. Geraerts, Towards social behavior in virtual-agent navigation, *Science China Information Sciences* 59 (2016) 1–17.
- [28] H. Sun, Z. Zhao, Z. He, Reciprocal learning networks for human trajectory prediction, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7416–7425.
- [29] Y. Bar-Shalom, X. R. Li, T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*, John Wiley & Sons, 2004.
- [30] S. Blackman, R. Popoli, *Design and analysis of modern tracking systems*(book), Norwood, MA: Artech House, 1999. (1999).
- [31] S. Pellegrini, A. Ess, K. Schindler, L. Van Gool, You’ll never walk alone: Modeling social behavior for multi-target tracking, in: *2009 IEEE 12th International Conference on Computer Vision*, IEEE, 2009, pp. 261–268.
- [32] A. Lerner, Y. Chrysanthou, D. Lischinski, *Crowds by example*, in: *Computer graphics forum*, volume 26, Wiley Online Library, 2007, pp. 655–664.
- [33] J. Amirian, B. Zhang, F. V. Castro, J. J. Baldelomar, J.-B. Hayet, J. Pettre, Opentraj: Assessing prediction complexity in human trajectories datasets, in: *Asian Conference on Computer Vision (ACCV), CONF*, Springer, 2020.
- [34] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, S. Savarese, Sophie: An attentive gan for predicting paths compliant to social and physical constraints, in: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1349–1358. doi:10.1109/CVPR.2019.00144.