

Embedding group and obstacle information in LSTM networks for human trajectory prediction in crowded scenes

Niccoló Bisagno*, Cristiano Saltori*, Bo Zhang +, Francesco G. B. De Natale*,
Nicola Conci*

**Department of Information Engineering and Computer Science - DISI, University of Trento, 38123, Italy*

+ Department of Intelligence Science and Technology, College of Information Science and Technology, Dalian Maritime University, 116026, China

Abstract: Recurrent neural networks have shown good abilities in learning the spatio-temporal dependencies of moving agents in crowded scenes. Recently, they have been adopted to predict the motion of pedestrians by learning the relative motion of each individual in the crowd with respect to its neighbours. Crowded scenes present a wide variety of situations, which do not depend solely on the agents' positions, but also relate to the structure of the environment, the density of the crowd, and the social relationships between pedestrians. In this work we propose a framework to improve the state-of-the-art models of crowd motion prediction by enriching the learning model with the social relationships between pedestrians walking in the crowd, as well as the layout of the environment. We observe that socially-related people tend to exhibit coherent motion patterns. Exploiting the motion coherency, we are able to cluster trajectories with similar motion properties and improve the trajectory prediction, especially at the group level. Furthermore, we incorporate into the model also the layout of the environment, to guarantee a more realistic and reliable learning framework. We evaluate our approach on standard crowd benchmark datasets, demonstrating its efficacy and applicability, improving the accuracy in trajectory prediction.

1. Introduction

Crowd motion analysis is a key feature in surveillance and monitoring applications, providing useful hints about potential threats to safety and security in urban and public spaces. It is well known that crowds are generally difficult to model, due to the diversity of the agents composing the crowd. Each individual is unique, being driven not only by the destination, but also by personality traits and attitude. In fact, when choosing their paths, pedestrians are influenced by multiple social and environmental factors, which include, for example, the distance from neighbours, the local crowd density, the rules regulating specific environments (e.g. crossroads, streetlights), as well as social rules regulating public indoor or outdoor spaces (e.g., standing in line at a ticket counter), different relationships with pedestrians they are walking with (e.g., couples, friends).

When navigating in a crowded space, pedestrians tend to follow non-linear trajectories to avoid collisions with other pedestrians and obstacles. In a crowded environment, individuals tend to speed up or slow down their paces in order to avoid collisions; people prefer to preserve personal space, thus keeping a certain distance from their neighbours; pedestrians tend to follow people in their front especially in the presence of over-crowded situations. This non-linearity makes it difficult to predict their trajectories.

The literature has demonstrated that some generic principles can be defined to predict human motion within a social context. In particular, the Social Force Model (SFM) [Helbing and Molnar(1995)] and the Reciprocal Velocity Obstacle [Van den Berg et al.(2008)Van den Berg, Lin and Manocha] model, exploit the presence and velocity of subjects in the neighborhood to improve the trajectory prediction of each pedestrian. More recently, deep learning methods, in particular Recurrent Neural Networks (RNN) and Long-Short Term Memory (LSTM) cells [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese, Vemula et al.(2018)Vemula, Muelling and Oh, Lee et al.(2017)Lee, Choi, Vernaza, Choy, Torr and Chandraker], have shown good results in the trajectory forecasting task. While traditional methods can merely make one-step forecasting (e.g., Kalman filter, particle filters, Markov chains), deep learning has shown the ability of enabling longer-term prediction.

These approaches, however, neither exploit the existing social relationships (e.g. couples walking together) between pedestrians, nor the interactions with obstacles in the environment. We observed that socially-related people, such as friends/couples/families, tend to move in accordance with a coherent motion pattern. Moreover,

pedestrians navigating in the environment are highly influenced by the presence/absence of obstacles in the scene. Existing models, like the SFM, provide repulsive forces with respect to obstacles to model the avoidance behavior. However deep learning models like [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese, Vemula et al.(2018)Vemula, Muelling and Oh, Lee et al.(2017)Lee, Choi, Vernaza, Choy, Torr and Chandraker] do not provide any sort of obstacle awareness.

Stemming from the preliminary work presented in [Bisagno et al.(2018)Bisagno, Zhang and Conci], in this work:

- we define *social groups* as pedestrians who share similar motion properties across time by exploiting the motion coherency [Zhou et al.(2012)Zhou, Tang and Wang]. In this way, pedestrians within the same group can be highlighted and segmented.
- we show how the identified social relationships can be exploited to improve the prediction.
- we add obstacle annotation to the deep learning framework to refine and improve the results of the prediction task.
- we show how a combined representation of socially-related groups and fixed obstacles in the environment can improve the performance of trajectory prediction.

The paper is organized as follows. In Section 2, the most representative state-of-the-art works are reported and briefly analyzed. In Section 4, the proposed framework is presented. In Section 5, we present the experimental setup (Sec. 5.1), quantitative results (Sec. 5.2), and qualitative results such as the predicted trajectories (Sec. 5.3). Concluding remarks are summarized in Section 6.

2. Related work

A detailed overview on the recent literature in crowd analysis, and in particular crowd dynamics modeling, social activity forecasting, and group segmentation, can be found in the following survey papers: [Li et al.(2015)Li, Chang, Wang, Ni, Hong and Yan] [Grant and Flynn(2017)] [Kok et al.(2016)Kok, Mei and Chan]. Therefore, in the next paragraphs, we will focus more on the works related to group analysis, obstacle avoidance, and prediction.

2.1. Group analysis in crowds

The early approaches face the group analysis problem adopting trajectories as low-level motion features in the crowd. By clustering trajectories with similar motion trends, pedestrians can be associated to different groups. In the paper by [Zhong et al.(2015)Zhong, Cai, Luo and Yin], k-means is exploited to learn different motion modalities in the scene. In [Lawal et al.(2017)Lawal, Poiesi, Aguita and Cavallaro], the authors rely on support vector clustering to group pedestrians. In [Zhou et al.(2012)Zhou, Tang and Wang], a coherent filtering strategy is presented to detect coherent motion patterns in a crowded environment.

As far as the representation of collective activities is concerned, [Ge et al.(2012)Ge, Collins and Ruback] detect small groups traveling together. [Ryoo and Aggarwal(2011)] introduce a probabilistic representation of group activities, for the purpose of recognizing different types of high-level group behaviors. [Yi et al.(2015)Yi, Li and Wang] investigate the interactions between stationary crowd groups and pedestrians to analyze pedestrians' behaviors, including path prediction, destination prediction, personality classification, and abnormal event detection. [Shao et al.(2016)Shao, Loy and Wang] propose a series of scene-independent descriptors to quantitatively describe group properties, such as collectiveness, stability, uniformity, and conflict. [Bagautdinov et al.(2017)Bagautdinov, Alahi, Fleuret, Fua and Savarese] present a unified end-to-end framework for multi-person action localization and collective activity recognition using deep recurrent networks.

2.2. Obstacle avoidance

In the literature, obstacles are generally bounded to moving agents through forces as proposed by [Helbing and Molnar(1995)] [Pelechano et al.(2007)Pelechano, Allbeck and Badler], or as boundaries to a fluid, as in [Treuille et al.(2006)Treuille, Cooper and Popović].

Data-driven approaches have recently been employed to capture and model interactions among people in a crowd and obstacles. The overall idea consists of trying to learn and reproduce either local or global features related to the crowd collective behavior. Vector fields are applied in multiple works to learn velocity and navigation features of real scenes. [Patil et al.(2011)Patil, Van Den Berg, Curtis, Lin and Manocha] use vector fields, either learned or manually sketched, to guide the crowd flow in the environment. Goal-dependent velocity fields are also used to guide the simulation at a global level, as proposed by [Bisagno et al.(2017)Bisagno, Conci and Zhang]. [Hu et al.(2008)Hu, Ali and Shah] propose a solution to deal with the global and local crowd flow, as a feature to be

transferred from real to simulated scenes, learning crowd motion patterns on the basis of the instantaneous motion field.

2.3. Human behavior prediction

Forecasting social activities has lately gained significant attention, especially in the domain of crowd analysis. As far as the interaction modeling is concerned, [Helbing and Molnar(1995)] propose the well known SFM, which relies on Newtonian forces to model the interactions between pedestrians, and guide each agent towards his goal. Other models, such as the continuum crowds model ([Treuille et al.(2006)Treuille, Cooper and Popović]), are capable to reproduce human interactions using priors. The crowd is modeled as a fluid, and agents are influenced by their goal, position, preferred speed, and a discomfort factor. In [Alahi et al.(2014)Alahi, Ramanathan and Li], the Social Affinity Maps features and the Origin and Destination priors are proposed to forecast pedestrians' destinations using multi-view surveillance cameras. In robotics, the Reciprocal Velocity Obstacle (RVO) (see [Van Den Berg et al.(2011)Van Den Berg, Guy, Lin and Manocha]) and its extensions ([Chandra et al.(2019)Chandra, Bhattacharya, Bera and Manocha]) are widely used to model the collision avoidance between agents. All these methods are able to describe crowd behaviors assuming the availability of a strong prior associated with the model. They are based on energy potentials, relative distances, and hand-crafted rules.

More recently, neural networks have been employed to predict events in crowded videos. In particular, with the emergence of deep generative models (such as RNN, LSTM, VAE, etc.), the sequence-to-sequence generation problem can be solved effectively, making it possible to handle the long-term prediction task directly. [Jain et al.(2016)Jain, Zamir, Savarese and Saxena] adopt a structural RNN that combines spatio-temporal graphs and recurrent neural networks to model motion and interactions in the scene. [Varshneya and Srinivasaraghavan(2017)] apply both the soft attention and the hard-wired attention on the Social-LSTM, significantly promoting the trajectory prediction performance. [Varshneya and Srinivasaraghavan(2017)] present a soft attention mechanism to forecast individual's paths, exploiting the spatially-aware deep attention model. [Vemula et al.(2018)Vemula, Muelling and Oh] propose a novel social attention model that can capture the relative importance of each person when navigating in the scene. [Haddad et al.(2019)Haddad, Wu, Wei and Lam] also employ an attention model to improve the modeling of different situations. [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese] propose the Social-LSTM to model the interactions among people in a neighbourhood by adding a new social pooling layer; in [Lee et al.(2017)Lee, Choi, Vernaza, Choy, Torr and Chandraker], the authors present a deep stochastic IOC RNN encoder-decoder framework to predict the future paths of multiple interacting agents in dynamic scenes. [Ballan et al.(2016)Ballan, Castaldo, Alahi, Palmieri and Savarese] consider both the dynamics of moving agents and the scene semantics to predict scene-specific motion patterns. [Pei et al.(2019)Pei, Qi, Zhang, Ma and Yang] introduce social affinity maps to improve the trajectory prediction task by modeling social relationships. In [Gupta et al.(2018)Gupta, Johnson, Fei-Fei, Savarese and Alahi], generative adversarial neural networks (GANs) are employed to predict socially-plausible trajectories in a crowded environment.

3. Preliminaries: the Social LSTM model

In the domain of path prediction, LSTM networks have shown good capabilities in predicting the behavior of pedestrians. The Social-LSTM model proposed by [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese] is able to capture the status of the neighbourhood of each agent (represented by a *social* hidden-state tensor) to refine the trajectory prediction.

The j -th pedestrian, referred to as ped^j at time t in the scene, is represented by the hidden-state h_t^j of a LSTM network. Let the hidden-state dimension be D , and the neighbourhood size be N_0 .

The neighbourhood of the agent ped^i is described by the social pooling layer defined as the tensor H_t^i , with dimensions $N_0 \times N_0 \times D$:

$$H_t^i(m, n, :) = \sum_{j \in N_i} 1_{mn}[x_t^j - x_t^i, y_t^j - y_t^i] h_{t-1}^j \quad (1)$$

where h_{t-1}^j represents the hidden-state of the LSTM for ped^j ($\forall j \neq i$) at $t - 1$, N_i represents the set of neighbours of pedestrian ped^i , and $1_{mn}[x, y]$ is an indicator of presence at location (m, n) defined as:

$$1_{mn}[x, y] = \begin{cases} 0 & \text{if } [x, y] \notin \text{cell } mn \\ 1 & \text{if } [x, y] \in \text{cell } mn \end{cases} \quad (2)$$

A graphical representation of the pooling operation is shown in Fig. 4a.

The next position (x_{t+1}^i, y_{t+1}^i) in the prediction depends on the hidden-state at the previous time-step h_t^i . Inspired by [Graves(2013)], and as performed in [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and

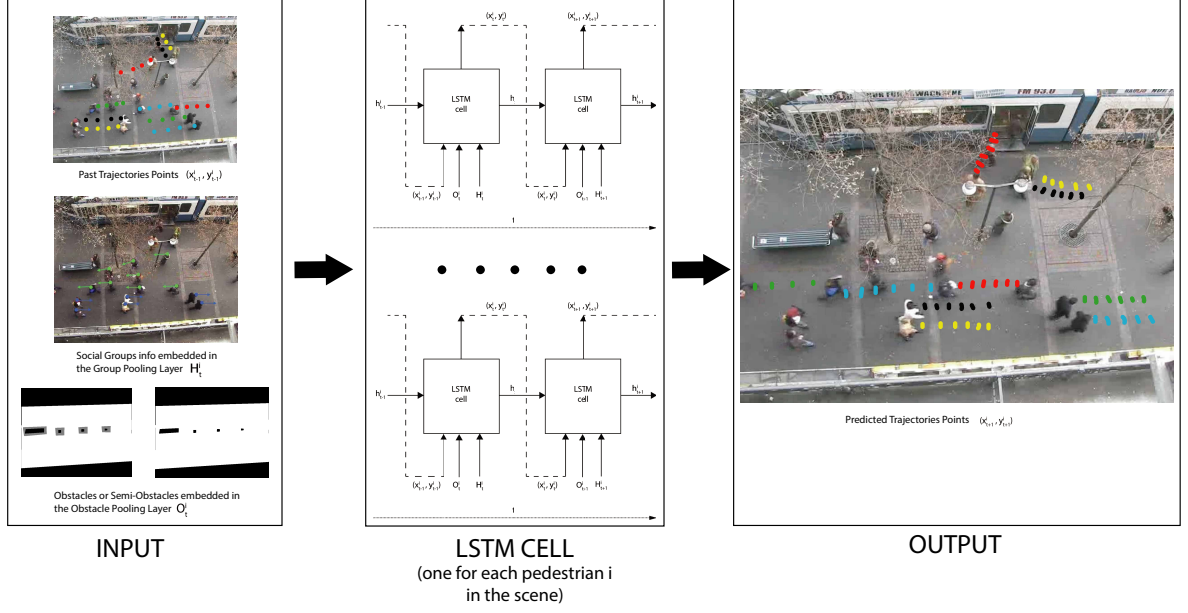


Fig. 1: The figure represent the workflow of our method. Starting from the observed past trajectories, combined with the group and obstacles/semi-obstacles information, we feed the LSTM cells to obtain as output the future predicted trajectories. The LSTM cell representation outlines the chain structure of the LSTM network between two consecutive time steps, t and $t + 1$. At each time step, the inputs of the LSTM cell are the previous position (x_{t-1}^i, y_{t-1}^i) , the Obstacle tensor O_t^i and the Social pooling tensor H_t^i . The output of the LSTM cell is the current position (x_t^i, y_t^i) .

[Savarese], the following parameters are predicted, which characterize a bi-variate Gaussian distribution: the mean $\mu_{t+1}^i = (\mu_x, \mu_y)_{t+1}^i$, the standard deviation $\sigma_{t+1}^i = (\sigma_x, \sigma_y)_{t+1}^i$ and the correlation coefficient ρ_{t+1}^i . The original model uses a $5 \times D$ weight matrix W_p to estimate the parameters. Thus, the coordinates at the next time-step $t + 1$ are computed as:

$$(x_{t+1}^i, y_{t+1}^i) \sim N(\mu_t^i, \sigma_t^i, \rho_t^i) \quad (3)$$

In order to estimate the parameters of the LSTM model, the negative log-likelihood loss L^i for agent ped^i is minimized for the current time t :

$$[\mu_t^i, \sigma_t^i, \rho_t^i] = W_p h_{t-1}^i \quad (4)$$

$$L^i(W_e, W_l, W_p) = - \sum_{t=T_{cur}+1}^{T_{step}} \log(\mathbb{P}(x_t^i, y_t^i | \mu_t^i, \sigma_t^i, \rho_t^i)) \quad (5)$$

The model is trained minimizing the log-likelihood loss for all the trajectories belonging to the dataset.

4. The proposed method

Pedestrians moving in a crowded scene adapt their motion according to the motion and positions of other pedestrians and the positions of obstacles in their neighbourhood. Thus, the prediction of the trajectory of an individual necessarily requires taking into consideration also the environment where the agent moves. In [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese], the state of the neighbourhood is modeled considering the interaction of the pedestrian with the neighbours. Compared to the original formulation of the Social-LSTM (briefly recapped in Section 3) we propose the use of a more comprehensive model, called Group-LSTM, which includes the social relationships between pedestrians in the crowd, thus improving the prediction performance. Furthermore, we propose a system module, called Obstacle-LSTM, which goal is to extend the description of the current state of the agent with the information related to fixed obstacles in the environment.



Fig. 2: Each pedestrian is represented by a single keypoint. Pedestrians walking in the same direction are clustered into one group s_i . In this example, two sets of pedestrians going in opposite directions are identified.

4.1. Group-LSTM

As mentioned in the previous paragraphs, the motion of pedestrians in crowded scenes is highly influenced by the behavior of other people in the surroundings and their mutual relationships. Stationary groups, groups of pedestrians walking together, people coming from opposite directions, will exert different effects on the action that one pedestrian takes. Thus, we propose a framework, which is able to consider whether the subject of interest is walking coherently with the pedestrians in his surroundings or not. By exploiting the coherent filtering approach [Zhou et al.(2012)Zhou, Tang and Wang], we first detect people moving coherently in a crowd, and then adopt the Social-LSTM to predict future trajectories. In this way, we are able to improve the prediction performance, accounting for the interactions between socially-related and unrelated pedestrians in the scene.

4.1.1. Pedestrian trajectory clustering

Coherent motion describes the collective movements of particles in a crowd. Coherent filtering [Zhou et al.(2012)Zhou, Tang and Wang] introduces a prior, meant to describe the coherent neighbour invariance, namely the local spatio-temporal relation between particles moving coherently. The algorithm is based on two steps. First, it detects the coherent motion of pedestrians in the scene. Then, points moving coherently are associated to the same cluster. Point clusters will continue to evolve, and new clusters will emerge over time. Eventually, each pedestrian i is assigned to a cluster s_i . The output of the coherent filtering consists of sets s_i ($i = 1, 2, \dots, n$) of people moving in a coherent manner.

The original implementation of the coherent filtering relies on the KLT tracker [Tomasi and Kanade(1991)], aiming at detecting candidate points for tracking and generating trajectories, which are then used as the input of the algorithm. Our objective is to cluster pedestrians into groups, where each individual in a group is represented using a single point, as shown in Fig. 2. Therefore, without loss of generality, we apply the coherent filtering algorithm directly on the ground truth of pedestrians' trajectories.

4.1.2. Group trajectory prediction

In the Social-LSTM model, each pedestrian is modeled using a LSTM network as displayed in Fig. 3. Each pedestrian is then linked with the other people in his neighbourhood via a so-called social pooling layer. The social pooling layer allows pedestrians to share their hidden states, thus enabling each network to predict the future positions of an individual based on its own hidden state and the hidden states in the neighbourhood.

The i^{th} pedestrian at time t in the scene is represented by the hidden state h_t^i in an LSTM network. We set the hidden-state dimension to D and the neighbourhood size to N_0 , respectively. The neighbourhood of the i^{th} agent ped^i is described using a tensor H_t^i as in Eq. 1, with dimensions equal to $N_0 \times N_0 \times D$:

$$H_t^i(m, n, :) = \sum_{j \in N} 1_{mn}[x_t^j - x_t^i, y_t^j - y_t^i] G_{ij}[s_i \neq s_j] h_{t-1}^j \quad (6)$$

where $1_{mn}[x, y]$ is an indicator function to select pedestrians in the neighbourhood defined as in Eq. 2.

If two pedestrians i and j belong to the same coherent set s_i , they will not be considered when computing the social pooling layer for each of them. The function $G_{ij}[i \in s_i, j \in s_i]$ is an indicator function defined as in Eq. 7:

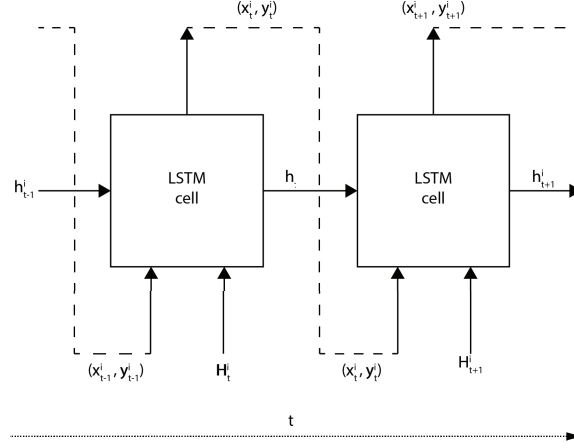


Fig. 3: The figure represents the chain structure of the LSTM network between two consecutive time steps, t and $t + 1$. At each time step, the inputs of the LSTM cell are the previous position (x_{t-1}^i, y_{t-1}^i) and the Social pooling tensor H_t^i . The output of the LSTM cell is the current position (x_t^i, y_t^i) .

$$G_{ij}[s_i \neq s_j] = \begin{cases} 0 & \text{if } i \in s_i, j \in s_i \\ 1 & \text{if } i \in s_i, j \notin s_i \end{cases} \quad (7)$$

Doing so, the social pooling layer of each pedestrian contains information only about socially unrelated pedestrians that are not moving coherently with the considered agent. If all the pedestrians in the neighbourhood belong to different coherent groups, our model performs similar to the Social LSTM [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese]. If all the pedestrians in the neighbourhood belong to the same coherent group, the tensor H_i will be a vector of 0s and the trajectory of pedestrian i is predicted as if no social interaction is occurring. The intuition behind this operation is that the network usually learns the repulsive forces between pedestrians avoiding collisions. If pedestrians are socially-related, they will tend to stay closer to each other, thus the repulsive forces will be smaller (almost neglected) and they will have a smaller influence on the future trajectory with respect to pedestrians not moving coherently. If socially-related pedestrians are considered in the social pooling, this would cause social groups to spread around instead of staying closer together as shown in Fig. 6 in Sec. 5. Fig. 6 shows how an absence of groups can lead to a performance degradation.

Once computed, the social hidden-state tensor is embedded into a vector a_t^i . The output coordinates are embedded in the vector e_t^i . Following the recurrence defined in Sec. 3, we can predict our trajectories gradually.

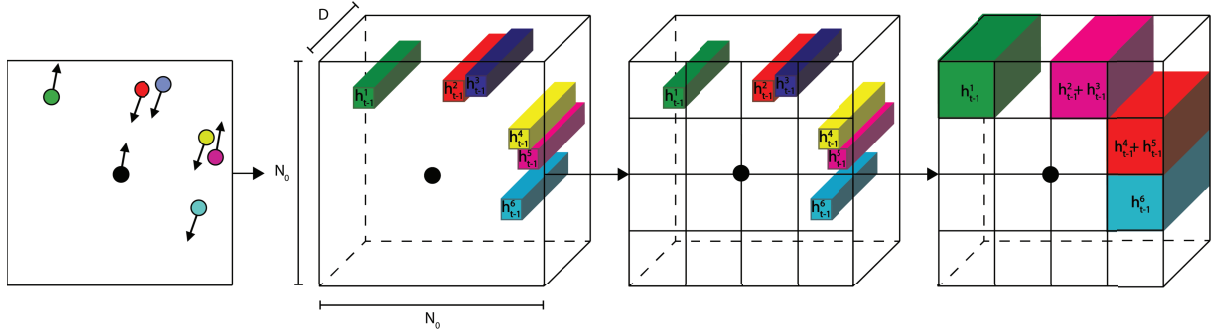
4.2. Obstacle-LSTM

Fixed obstacles in an environment have a non-negligible impact in shaping and determining the flow of a crowd. Depending on the nature of the obstacles, pedestrians actuate different strategies to prevent collisions, taking subjective decisions. We then want to embed this piece of information in the proposed model, thus considering the presence of an obstacle in the agent's neighbourhood. We exploit this additional annotation to improve the prediction performance by taking into account the interactions between pedestrians and obstacles in the scene. Since automatic obstacle detection is out of the scope of this paper, in the present work we manually annotate fixed obstacles in the scene, being a one-time operation. Despite the scene might contain also dynamic obstacles (e.g., cars, bicycles), the number of examples is too limited, thus making it not possible to train a corresponding model.

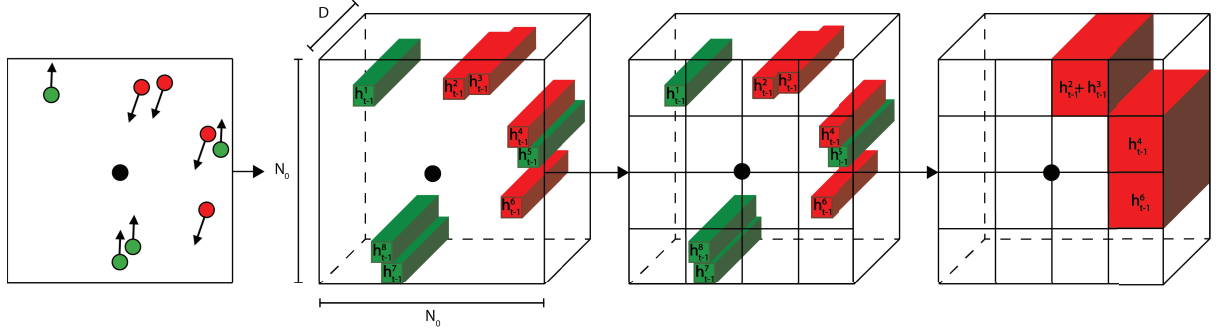
As far as the annotation is concerned, and for the sake of completeness, we have identified two separate categories of obstacles:

- *obstacle*, which has to be avoided, and that the pedestrian can not traverse (e.g., trees or light poles, walls);
- *semi-obstacle*, which is preferably avoided by a walking pedestrian (e.g., the flower bed around a tree, a meadow or a snowy part of the street).

This distinction will drive two different experiments, one considering the obstacles only, the second one considering both obstacles and semi-obstacles, so as to closely mimic how the pedestrians in the scene behave differently



(a) Representation of the Social hidden-state tensor H_t^i . The black dot represents the pedestrian of interest ped_i . Other pedestrians ped_j ($\forall j \neq i$) are shown in different color codes. The state of the neighbourhood of ped_i is described by $N_0 \times N_0$ cells, which preserves the spatial information by pooling together neighbours in the same cell.



(b) Other pedestrians ped_j ($\forall j \neq i$) are shown in different color codes, namely green for pedestrians belonging to the same set, and red for pedestrians belonging to a different set. The neighbourhood of ped_i is described by $N_0 \times N_0$ cells, which preserves the spatial information by pooling spatially-adjacent neighbours. Pedestrians belonging to the same set are not used for the final computation of the pooling layer H_t^i .

Fig. 4: Representation of the Social hidden-state tensor H_t^i . The black dot represents the pedestrian of interest ped_i .

with respect to the scene layout. Once these classes have been annotated, two different obstacle maps (one with obstacles only, and one with obstacles and semi-obstacles) are generated for each investigated scenario taken from [Pellegrini et al.(2009)Pellegrini, Ess, Schindler and Van Gool] and [Lerner et al.(2007)Lerner, Chrysanthou and Lischinski]. Similarly to the process of trajectory annotation, obstacles are annotated on the image plane and their coordinates are then projected on the ground plane in meters using the available homography for each dataset.

Figure 5 reports the obstacle and semi-obstacle map of the 5 scenes of interest. From now, if not explicitly mentioned, we will use the term *obstacles* including both obstacles and semi-obstacles. After obtaining the obstacle maps, the corresponding coordinates have been annotated by equally sampling points on each external boundary area.

4.2.1. Obstacle Tensor

The obstacle pooling layer allows pedestrians to understand the state of obstacles in their neighbourhood, thus improving their knowledge about the surrounding environment.

The configuration of obstacles is represented by the Obstacle Tensor. We set the neighbourhood size to N_0 . The obstacles in the neighbourhood of the i^{th} agent ped^i are described using a tensor O_t^i , with dimensions $N_0 \times N_0 \times D$.

We define two versions of the obstacle tensor:

- the Obstacle Presence (OP) tensor, where a constant value indicates the presence or absence of an obstacle in a given cell;
- the Obstacle Distance (OD) tensor, where the distance between the obstacle and the pedestrian of interest indicates the presence or absence of an obstacle in a given cell.

The OP tensor is represented as:

$$OP_t^i(m, n) = \sum_{j \in N_o} 1_{mn}[x^i - x^j, y^i - y^j], \forall i \in N_p \quad (8)$$

where 1_{mn} is the presence function at location (m, n) of the pedestrian i w.r.t. each obstacle j in the scene obstacle set N_o , and N_p is the number of people present in the frame. The OP tensor allows the model to learn whether a cell (m, n) in the neighbourhood of the pedestrian i is occupied or not.

Instead of representing just the occupancy, the OD tensor takes into account the Euclidean Norm $D(i, j)$, between the observed pedestrian i and the obstacle j , such that:

$$D(i, j) = \sqrt{(x^i - x^j)^2 + (y^i - y^j)^2}. \quad (9)$$

The OD_t^i tensor is represented as:

$$OD_t^i(m, n) = \sum_{j \in N_o} 1_{mn}[x^i - x^j, y^i - y^j]D(i, j), \forall i \in N_p \quad (10)$$

where 1_{mn} is the presence function in position (m, n) of the pedestrian i w.r.t. each obstacle j in the scene obstacle set N_o , $D(i, j)$ is the Euclidean Norm between i and j , and N_p is the number of people present in the frame.

4.2.2. Pedestrian trajectory prediction with obstacles

Like in [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese], the Obstacle tensor and the social hidden-state tensor are embedded into vectors b_t^i and a_t^i , respectively. The output coordinates are embedded in vector e_t^i .

The resulting recurrence is then defined by the following equations:

$$e_t^i = \Phi(x_t^i, y_t^i; W_r) \quad (11)$$

$$a_t^i = \Phi(H_t^i; W_e) \quad (12)$$

$$b_t^i = \Phi(O_t^i; W_o) \quad (13)$$

$$h_t^i = LSTM(h_{t-1}^i, e_t^i, a_t^i, b_t^i; W_l) \quad (14)$$



Fig. 5: Obstacle(black) and Semi-Obstacle(gray) classes.

Thus, in the Obstacle-LSTM, the input tensor to the LSTM cell is composed by three embedded tensors with the same size (as shown in Fig. 1): the Social tensor, the actual pedestrian position, and the Obstacle tensor. At deploying time, we can have 4 different Obstacle-LSTM configuration, depending on whether we are using OP or OD, coupled with either obstacles or semi-obstacles maps.

5. Experimental Results

5.1. Implementation Details

For the implementation, we relied on the PyTorch framework, and the RMSprop is used to optimize the objective function. Our network is trained using the following hyper-parameters: 200 epochs, 0.0015 learning rate, 0.95 decay rate. The final model has 3 embedding input layers one for each input: the position p_t at time t , the Social tensor [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese], and the Obstacle tensor.

The coherent filtering parameters are set up according to the original paper. So, we use $K = 10$, $d = 1$ and $\lambda = 0.2$ for our group clustering.

In line with all the state-of-the-art, our method is also evaluated on the UCY [Lerner et al.(2007)Lerner, Chrysanthou and Lischinski] and ETH [Pellegrini et al.(2009)Pellegrini, Ess, Schindler and Van Gool] benchmark datasets. The UCY datasets consists of 3 videos of two different scenes with 786 people. The ETH dataset consists of two videos of different scenes containing 750 pedestrians in total. The training and test procedure is performed using a leave-one-out approach, namely, training and validating on 4 sets of trajectories and testing on the remaining one.

5.2. Quantitative results

Similarly to other works in the literature ([Pellegrini et al.(2009)Pellegrini, Ess, Schindler and Van Gool, Lerner et al.(2007)Lerner, Chrysanthou and Lischinski]), we evaluate our approaches using the following two metrics:

- *Average Displacement Error (ADE)*, namely the average displacement error (in meters) between each point of the predicted path with respect to the ground truth path.
- *Final Displacement Error (FDE)*, namely the distance (in meters) between the final point of the predicted trajectory and the final point of the ground truth trajectory.

For both training and validation (again in line with the relevant works in the literature), we observe and predict trajectories using a time interval of 0.4 seconds. We observe trajectories for 8 time steps and predict for the next 12 time steps, meaning that we observe trajectories for $t_{obs} = 3.2$ seconds and predict for the next $t_{pred} = 4.8$ seconds. In the training phase, only trajectories that remain in the scene for at least 8 seconds are considered.

We compare our method with the Social-LSTM model ([Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese]) and its most recent variant proposed by [Gupta et al.(2018)Gupta, Johnson, Fei-Fei, Savarese and Alahi]. We also compare our model with a linear model, which uses the Kalman filter to predict future trajectories under the assumption of linear acceleration, as also reported in [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese]. The numerical results are shown in Table 1.

The proposed Group-LSTM (G-LSTM) performs on average better or equal than both Social-LSTM and Social-GAN, especially on the UCY dataset. This is due to the characteristics of crowd flows in the scene, which usually consist of easily identifiable groups walking in opposite directions. For the ETH dataset, the motion patterns are instead more varied and chaotic.

The achieved results show that the prediction performance can be improved when considering pedestrians that are not moving coherently. In fact, the change of motion and the evolution of trajectories are mainly influenced by pedestrians moving in different directions with respect to the pedestrian of interest. People walking together, instead, loosely influence each other, since they behave as a group.

The performance of our Obstacle-LSTM varies depending on whether we are using a combination of obstacles or semi-obstacles with either the Presence tensor or Distance tensor. The Obstacle-LSTM performs better than the other methods on the ETH dataset, thanks to the strong influence determined by obstacles in both scenes. In the Zara scene, instead, moving obstacles (e.g., car) have not been annotated and this causes a decrease of the network performance in the prediction task.

When combined together, Obstacle and Group-LSTM outperform other methods. The best results are obtained using a combination of the Group-LSTM with the Distance Tensor.

5.2.1. Ablation Study.

Our Group modules allows for better performances in every dataset, but UCY. The UCY dataset is characterised by a more chaotic behavior with groups merging and splitting, which causes the group module to worsen the

overall performances of our approach. Overall, the Group module is able to improve the performances of the network especially when the crowd flow is more regular. The Obstacle approach performs in scene characterised by fewer obstacles distributed on the edges of the walkable space (e.g. ETH, UCY). The Semi-Obstacles approach outperform the Obstacle one when there are obstacles in the middle of the walking space, thus causing the scene to be more difficult to navigate. In both obstacles and semi-obstacles cases, providing the Distance of the obstacle with respect to the pedestrian of interest outperform the Presence method because a more precise information is passed to the network. Performances are similar when the scene presents fewer obstacles in the middle of the walking space (e.g. ETH,UCY)

Metric	ADE						FDE					
Dataset	ETH	HOTEL	ZARA1	ZARA2	UCY	Average	ETH	HOTEL	ZARA1	ZARA2	UCY	Average
Lin. [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese]	1.33	0.39	0.62	0.77	0.82	0.79	2.94	0.72	1.21	1.48	1.59	1.59
S-LSTM [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese]	1.09	0.67	0.41	0.52	0.61	0.70	2.41	1.91	1.11	1.31	0.88	1.52
S-GAN [Gupta et al.(2018)Gupta, Johnson, Fei-Fei, Savarese and Alahi]	0.81	0.72	0.34	0.42	0.60	0.58	1.52	1.61	0.84	1.26	0.69	1.18
G-LSTM [Bisagno et al.(2018)Bisagno, Zhang and Conci]	0.48	0.47	0.23	0.34	0.56	0.42	1.12	0.89	0.91	1.49	1.48	1.18
OP-LSTM	0.40	0.44	0.42	0.40	0.39	0.41	0.87	1.10	1.01	1.68	1.89	1.31
OD-LSTM	0.40	0.43	0.38	0.40	0.40	0.40	0.85	0.83	0.81	1.57	1.67	1.14
SOP-LSTM	0.46	0.46	0.37	0.34	0.38	0.40	1.19	0.86	0.86	1.35	1.74	1.20
SOD-LSTM	0.44	0.40	0.36	0.37	0.37	0.39	1.20	0.88	0.81	1.57	1.67	1.23
G/SOP-LSTM	0.36	0.34	0.18	0.26	0.73	0.37	0.78	1.08	0.80	1.15	1.75	1.11
G/SOD-LSTM	0.40	0.31	0.17	0.20	0.81	0.38	0.92	0.83	0.69	0.91	2.04	1.08
G/OP-LSTM	0.38	0.33	0.20	0.26	0.75	0.38	0.79	1.06	0.90	1.12	1.95	1.16
G/OD-LSTM	0.36	0.33	0.21	0.22	0.64	0.35	0.75	0.93	0.92	0.95	1.96	1.10

Table 1: Legend: Lin. (Linear [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese]), S-LSTM (social LSTM [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese], S-GAN (social GAN [Gupta et al.(2018)Gupta, Johnson, Fei-Fei, Savarese and Alahi]), G-LSTM (Group LSTM), OP-LSTM (Obstacle Presence LSTM), OD-LSTM (Obstacle Distance LSTM), SOP-LSTM (Semi-obstacle Presence LSTM), SOD-LSTM(Semi-obstacle Distance LSTM), G/OP-LSTM (Group Obstacle Presence LSTM), G/OD-LSTM (Group Obstacle Distance LSTM), G/SOP-LSTM (Group Semi-obstacle Presence LSTM), G/SOD-LSTM(Group Semi-obstacle Distance LSTM)).

The Average Displacement Error (ADE) and the Final Displacement Error (FDE) are reported (in meters) for an observation interval $t_{obs} = 3.2$ seconds and a prediction of subsequent $t_{pred} = 4.8$ seconds.

5.3. Qualitative results

In Section 5.2 we have shown that considering only pedestrians not moving coherently can improve the prediction precision. In this section we will further evaluate the consistency of the predicted trajectories.

As a general rule, the LSTM-based approaches for trajectory prediction follow a data-driven strategy. Furthermore, the future planning of pedestrians in a crowd are highly influenced by their goals, their surroundings, and their past motion history. Pooling the correct data in the social layer can promote the prediction performance in a significant way.

In order to guarantee a reliable prediction, we do not only need to account for spatio-temporal relationships, but we also need to preserve the social nature of behaviors. According to the studies in interpersonal distances ([Hall et al.(1968)Hall, Birdwhistell, Bock, Bohannon, Diebold Jr, Durbin, Edmonson, Fischer, Hymes, Kimball et al., Conci et al.(2018)Conci, Bisagno and Cavallaro]), socially-related people tend to stay closer in their personal space and walk together in crowded environments as compared to pacing with unknown pedestrians. Pooling only unrelated pedestrians will focus more on macroscopic inter-group interactions rather than intra-group dynamics, thus allowing the LSTM network to improve the trajectory prediction performance. Collision avoidance influences the future motion of pedestrians in a similar manner if two pedestrians are walking together as in a group.

In Figure 6, Figure 8 and Figure 7, we show some examples of predicted trajectories, which highlight how the Group-LSTM is able to predict pedestrian trajectories with better precision, demonstrating how the prediction is improved when we pool in the social tensor of each pedestrian only pedestrians not belonging to the group.

In Figure 6, we show how the prediction of two pedestrians walking together in the crowd improves, when they are not pooled in each other's pooling layer. When the two pedestrians are pooled together, the network applies the typical repulsion force to avoid collisions. However, since they belong to the same group, each pedestrian *allows* the other to remain closer.

In Figure 7, we display the sequences of two groups walking toward each other. In Figure 8, we show how the prediction for the two groups is improved with respect to the Social-LSTM, demonstrating the ability to forecast how pedestrians belonging to the same group stay together when moving in the environment.

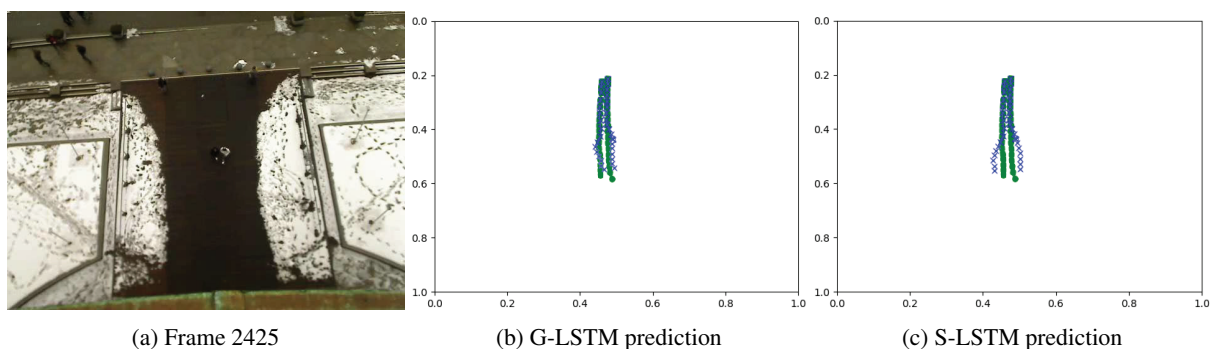


Fig. 6: ETH dataset Frame 2425: the prediction is improved when pooling in the social tensor of each pedestrian only pedestrians not belonging to the same group. The green dots represent the ground truth trajectories; the blue crosses represent the predicted paths. Using the grouping module allows the predictor to keep pedestrians belonging to the same group closer than in the case of S-LSTM.



Fig. 7: Sequences taken from the UCY dataset. An interaction example between two groups is shown. An in-depth analysis is further presented in Figure 8.

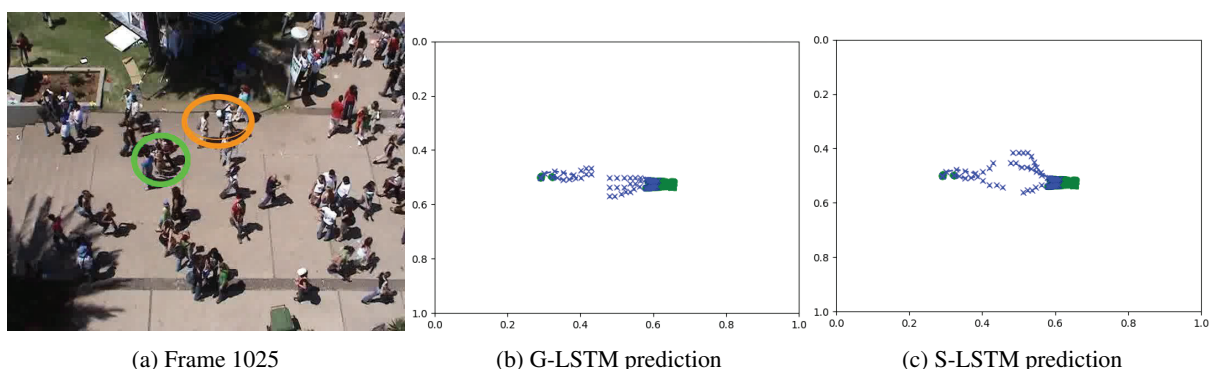


Fig. 8: UCY univ Frame 1025. We display how the prediction is improved for two groups walking in opposite directions. The green dots represent the ground truth trajectories, while the blue crosses represent the predicted paths. The G-LSTM predicts socially-related people to stay closer together than in the S-LSTM approach.

6. Conclusion

In this work, we tackle the problem of pedestrian trajectory prediction in crowded scenes. We propose a novel approach, which combines group detection and obstacle annotation to refine the prediction. With the Group-LSTM, coherent filtering is used to identify pedestrians walking together in a crowd, while the LSTM network is used to predict the future trajectories by exploiting inter and intra group dynamics. The Obstacle-LSTM displays how obstacles and non-trespassing areas can strongly influence future trajectories of pedestrian. Experimental results show that the proposed LSTM framework, which embeds group and obstacle information, outperforms the Social-LSTM in the prediction task on two public benchmarks (the UCY and the ETH datasets).

References

- [Alahi et al.(2016)Alahi, Goel, Ramanathan, Robicquet, Fei-Fei and Savarese] Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., Savarese, S., 2016. Social lstm: Human trajectory prediction in crowded spaces, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 961–971.
- [Alahi et al.(2014)Alahi, Ramanathan and Li] Alahi, A., Ramanathan, V., Li, F.F., 2014. Socially-aware large-scale crowd forecasting, in: Proceedings of the International Conference on Computer Vision and Pattern Recognition, IEEE. pp. 2203–2210.
- [Bagautdinov et al.(2017)Bagautdinov, Alahi, Fleuret, Fua and Savarese] Bagautdinov, T.M., Alahi, A., Fleuret, F., Fua, P., Savarese, S., 2017. Social scene understanding: End-to-end multi-person action localization and collective activity recognition., in: CVPR, pp. 3425–3434.
- [Ballan et al.(2016)Ballan, Castaldo, Alahi, Palmieri and Savarese] Ballan, L., Castaldo, F., Alahi, A., Palmieri, F., Savarese, S., 2016. Knowledge transfer for scene-specific motion prediction, in: European Conference on Computer Vision, Springer. pp. 697–713.
- [Van den Berg et al.(2008)Van den Berg, Lin and Manocha] Van den Berg, J., Lin, M., Manocha, D., 2008. Reciprocal velocity obstacles for real-time multi-agent navigation, in: IEEE International Conference on Robotics and Automation, pp. 1928–1935.
- [Bisagno et al.(2017)Bisagno, Conci and Zhang] Bisagno, N., Conci, N., Zhang, B., 2017. Data-driven crowd simulation. Advanced Video and Signal Based Surveillance (AVSS). IEEE , 1–6.
- [Bisagno et al.(2018)Bisagno, Zhang and Conci] Bisagno, N., Zhang, B., Conci, N., 2018. Group lstm: Group trajectory prediction in crowded scenarios. ECCV workshop: Anticipating Human Behavior .
- [Chandra et al.(2019)Chandra, Bhattacharya, Bera and Manocha] Chandra, R., Bhattacharya, U., Bera, A., Manocha, D., 2019. Denseped: Pedestrian tracking in dense crowds using front-rvo and sparse features. arXiv preprint arXiv:1906.10313 .
- [Conci et al.(2018)Conci, Bisagno and Cavallaro] Conci, N., Bisagno, N., Cavallaro, A., 2018. On modeling and analyzing crowds from videos, in: Computer Vision for Assistive Healthcare. Elsevier, pp. 319–336.
- [Ge et al.(2012)Ge, Collins and Ruback] Ge, W., Collins, R.T., Ruback, R.B., 2012. Vision-based analysis of small groups in pedestrian crowds. IEEE transactions on pattern analysis and machine intelligence 34, 1003–1016.
- [Grant and Flynn(2017)] Grant, J., Flynn, P., 2017. Crowd scene understanding from video: A survey. ACM Transactions on Multimedia Computing, Communications, and Applications 13.
- [Graves(2013)] Graves, A., 2013. Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850 .
- [Gupta et al.(2018)Gupta, Johnson, Fei-Fei, Savarese and Alahi] Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A., 2018. Social gan: Socially acceptable trajectories with generative adversarial networks, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
- [Haddad et al.(2019)Haddad, Wu, Wei and Lam] Haddad, S., Wu, M., Wei, H., Lam, S.K., 2019. Situation-aware pedestrian trajectory prediction with spatio-temporal attention model. arXiv preprint arXiv:1902.05437 .
- [Hall et al.(1968)Hall, Birdwhistell, Bock, Bohannon, Diebold Jr, Durbin, Edmonson, Fischer, Hymes, Kimball et al.] Hall, E.T., Birdwhistell, R.L., Bock, B., Bohannon, P., Diebold Jr, A.R., Durbin, M., Edmonson, M.S., Fischer, J., Hymes, D., Kimball, S.T., et al., 1968. Proxemics [and comments and replies]. Current anthropology 9, 83–108.
- [Helbing and Molnar(1995)] Helbing, D., Molnar, P., 1995. Social force model for pedestrian dynamics. Physical review E. APS 51, 4282.
- [Hu et al.(2008)Hu, Ali and Shah] Hu, M., Ali, S., Shah, M., 2008. Learning motion patterns in crowded scenes using motion flow field, in: International Conference on Pattern Recognition, pp. 1–5.
- [Jain et al.(2016)Jain, Zamir, Savarese and Saxena] Jain, A., Zamir, A.R., Savarese, S., Saxena, A., 2016. Structural-rnn: Deep learning on spatio-temporal graphs, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5308–5317.
- [Kok et al.(2016)Kok, Mei and Chan] Kok, V., Mei, K., Chan, C., 2016. Crowd behavior analysis: A review where physics meets biology. Neurocomputing 177, 342–362.
- [Lawal et al.(2017)Lawal, Poiesi, Aguita and Cavallaro] Lawal, I., Poiesi, F., Aguita, D., Cavallaro, A., 2017. Support vector motion clustering. IEEE Transactions on Circuits and Systems for Video Technology 27, 2395–2408.
- [Lee et al.(2017)Lee, Choi, Vernaza, Choy, Torr and Chandraker] Lee, N., Choi, W., Vernaza, P., Choy, C.B., Torr, P.H., Chandraker, M., 2017. Desire: Distant future prediction in dynamic scenes with interacting agents, in: Proceedings of the

- IEEE Conference on Computer Vision and Pattern Recognition, pp. 336–345.
- [Lerner et al.(2007)Lerner, Chrysanthou and Lischinski] Lerner, A., Chrysanthou, Y., Lischinski, D., 2007. Crowds by example, in: Computer Graphics Forum, Wiley Online Library. pp. 655–664.
- [Li et al.(2015)Li, Chang, Wang, Ni, Hong and Yan] Li, T., Chang, H., Wang, M., Ni, B., Hong, R., Yan, S., 2015. Crowded scene analysis: A survey. *IEEE Transactions on Circuits and Systems for Video Technology* 25, 367–386.
- [Patil et al.(2011)Patil, Van Den Berg, Curtis, Lin and Manocha] Patil, S., Van Den Berg, J., Curtis, S., Lin, M.C., Manocha, D., 2011. Directing crowd simulations using navigation fields. *Transactions on Visualization and Computer Graphics*. IEEE 17, 244–254.
- [Pei et al.(2019)Pei, Qi, Zhang, Ma and Yang] Pei, Z., Qi, X., Zhang, Y., Ma, M., Yang, Y.H., 2019. Human trajectory prediction in crowded scene using social-affinity long short-term memory. *Pattern Recognition* 93, 273–282.
- [Pelechano et al.(2007)Pelechano, Allbeck and Badler] Pelechano, N., Allbeck, J.M., Badler, N.I., 2007. Controlling individual agents in high-density crowd simulation, in: Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation, Eurographics Association. pp. 99–108.
- [Pellegrini et al.(2009)Pellegrini, Ess, Schindler and Van Gool] Pellegrini, S., Ess, A., Schindler, K., Van Gool, L., 2009. You'll never walk alone: Modeling social behavior for multi-target tracking. *International Conference on Computer Vision (ICCV)*. IEEE , 261–268.
- [Ryoo and Aggarwal(2011)] Ryoo, M., Aggarwal, J., 2011. Stochastic representation and recognition of high-level group activities. *International journal of computer Vision* 93, 183–200.
- [Shao et al.(2016)Shao, Loy and Wang] Shao, J., Loy, C.C., Wang, X., 2016. Learning scene-independent group descriptors for crowd understanding. *IEEE Transactions on Circuits and Systems for Video Technology* .
- [Tomasi and Kanade(1991)] Tomasi, C., Kanade, T., 1991. Detection and tracking of point features .
- [Treuille et al.(2006)Treuille, Cooper and Popović] Treuille, A., Cooper, S., Popović, Z., 2006. Continuum crowds, pp. 1160–1168.
- [Van Den Berg et al.(2011)Van Den Berg, Guy, Lin and Manocha] Van Den Berg, J., Guy, S.J., Lin, M., Manocha, D., 2011. Reciprocal n-body collision avoidance. *Robotics research*. Springer , 3–19.
- [Varshneya and Srinivasaraghavan(2017)] Varshneya, D., Srinivasaraghavan, G., 2017. Human trajectory prediction using spatially aware deep attention models. <https://arxiv.org/abs/1705.09436> .
- [Vemula et al.(2018)Vemula, Muelling and Oh] Vemula, A., Muelling, K., Oh, J., 2018. Social attention: Modeling attention in human crowds, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), IEEE. pp. 1–7.
- [Yi et al.(2015)Yi, Li and Wang] Yi, S., Li, H., Wang, X., 2015. Understanding pedestrian behaviors from stationary crowd groups. *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE , 3488–3496.
- [Zhong et al.(2015)Zhong, Cai, Luo and Yin] Zhong, J., Cai, W., Luo, L., Yin, H., 2015. Learning behavior patterns from video: a data-driven framework for agent-based crowd modeling. *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)* , 801–809.
- [Zhou et al.(2012)Zhou, Tang and Wang] Zhou, B., Tang, X., Wang, X., 2012. Coherent filtering: Detecting coherent motions from crowd clutters. *European Conference on Computer Vision* , 857–871.