Explaining Pedestrian Intention from Behavior

Zhi Qiao and Ross A. Knepper, *Member, IEEE*

***Abstract*—Efficient and competent robot navigation among human pedestrians requires prediction of the intentions of other agents (humans or robots). Those intentions are unobservable, but under the rational actor assumption, a focus pedestrian's behavior provides clues to his or her intentions. Those clues in turn permit an observer to infer circumstances that are known by the pedestrian but are unknown to the observer, such as the existence of another pedestrian approaching from around a corner. In this paper, we examine the question of how to identify the cause of observed changes in trajectory as a window into intention. We propose the influencer recognition model (IRM), a data-driven model to infer a distribution over possible causes of observed bends. We identify each possible influencer in the dataset using a 2-D kernel density estimator. With this model, we can calculate the relative likelihood with which other pedestrians influence the focus agent. We train and evaluate IRM on recorded pedestrian datasets. Finally, we present two sample applications of this model to predict a pedestrian's goal and infer the existence of obstacles outside of the observer's view.**

I. Introduction

Social navigation can be characterized as the interplay of two objectives: reaching one's goal and maintaining socially competent behavior. Competent pedestrian navigation requires robots to reason about human intentions [1], which can be characterized as the pair of an eventual destination and a plan for getting there. A human observer employs teleological reasoning [2] to infer the intentions of another human based on the choice of actions they perform. Conversely, when the intention is known, humans anticipate a *predictable* action that minimizes energy [3] – a straight line path.

As the density of pedestrians in a space increases, the social competency objective increasingly conflicts with minimum-energy goal-attainment. To reconcile the conflict, humans plan more complex paths comprising sequences of subgoals, each its own destination attainable from the last via a straight-line path. When behaving predictably then, people roughly follow a sequence of straight lines connecting subgoals (Fig. 1). During social navigation in a pedestrian context, people employ teleological reasoning based on the observed sequence of motions in order to infer one another's intentions. This inference, in turn, allows people to anticipate the motions of others and select compatible motions in advance for collision avoidance.

*This material is based upon work supported by HIT Robot Group and by the National Science Foundation under Grant No, 1526035.

Z. Qiao is with the Harbin Institute of Technology, Harbin, China qz709430186@163.com

R. A. Knepper is with Cornell University, Ithaca, USA rak@cs.cornell.edu

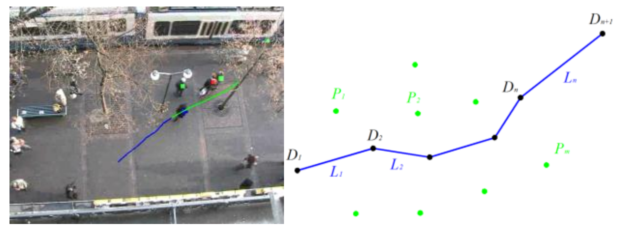

Figure 1. In a low-feature environment, a pedestrian's turning is usually caused by other pedestrians (the person with red square in the left figure). The right figure shows this problem abstractly, black dots D_i are deflection points of the focus agent and green dots P_i are other pedestrians when D_2 happens.

In this paper, we propose the *influencer recognition model* (IRM) to address the problem of identifying subgoals of a *focus agent* and predicting the *cause* of each subgoal – that is, the pedestrian *influencer* that necessitated it – based on observation of their path in context. We contribute:

- a data-driven, probabilistic, causative model for attributing bends in a pedestrian's path,
- a predictive model that identifies obstacles or agents in the environment responsible for bends,
- a method of predicting where and when the causative agent is obscured from view, and
- results evaluating the predictive power of our approach.

II. Related Work

Work in social navigation primarily addresses the related problems of prediction and control of social agents. These methods can be divided broadly into global and local methods. The local method focuses on the analysis and simulation of single agent with local interactions and usually makes decisions based on little information. The global method in contrast considers the history and context of the interaction.

The best-known local method is the social force model proposed by Helbing and Molnar [3]. This method constructs attractive and repulsive forces to guide agents' motions; pedestrians are repelled by one another and by static obstacles, whereas an attractive force towards the goal attempts to maintain progress. Other authors improved and expanded it variously. Tamura, et al. [4] assume a goal in the front of pedestrian and add a social force model to predict their trajectory. Linear trajectory avoidance (LTA) [5] is a pedestrian prediction method like the social force model in various aspects, but it differs in how to deal with agents. In LTA simulation, pedestrians have decisive directions and optimize paths by a collision-free principle, rather than just being reactive particles driven by force. Being derived from the social force model, all these methods ignore some valuable information given by past trajectories of the agents.

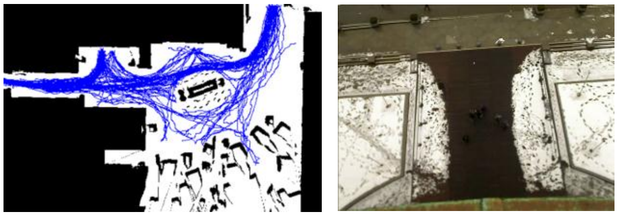

Figure 2. Left: In an environment [7] cluttered with obstacles, pedestrians have fixed trajectories in most time. Right: In an environment [5] with few static obstacles, pedestrians have more flexible trajectories.

Next, we consider global methods, in which the entire trajectory history of a pedestrian is used when predicting. One could model a pedestrian's trajectory as a Markov process in the sense that a single state (current plus initial position) is sufficient to predict their motion in an environment with many static obstacles, where their trajectory is influenced greatly by the environment (left in Fig. 2) [7]. Pedestrians always choose similar path from a common place to another one so that their trajectories are limited in such space. The Markov model breaks down in a *low-clutter environment* because pedestrian motions are influenced primarily by other pedestrians rather than the static environment, such as the scene in the BIWI walking pedestrian dataset (right in Fig. 2) [5]. This environment still has two obvious static obstacles om in the camera's view (the left and right snow boundaries), but the center region is nearly obstacle-free, so pedestrian-obstacle interactions are rare.

For conscious pedestrians, their historical trajectories and velocities are important information, and we can extrapolate their future trajectory from historical trajectories. Learning the motion patterns of persons and utilizing them is a common way to predict action of people [6]. This approach applies the EM algorithm to collect persons' motion data by laser-range sensors in an enclosed environment and introduces a method for robot navigation that complies to human behavior. In an indoor environment (left in Fig. 2), Ziebart, et al. [7] also collect pedestrians' trajectories over time and make path predictions with the principle of maximum entropy. The method obtains a cost-map of the current environment and estimates a distribution to predict possible future path from only the initial and current positions.

These approaches are limited to known environments; pedestrian behaviors are also subject to factors that are unique to a given environment. The SBCM framework [8] predicts a pedestrian's future trajectory by collecting trajectories in various conditions with a pedestrian ego-graph (PEG). This method takes advantage of general spatial effects (GSEs), which are conventions of pedestrian behavior that apply everywhere, and it discovers specific spatial effects (SSEs), which apply only in a specific context.

The drawback of identifying GSEs and SSEs is that they neglect the interaction between pedestrians, which leads to missed information implied by subgoals. As the complexity of environment grows, pedestrian behavior deviates increasingly from predictable trajectories by adding more subgoals. In order to explain the mechanism that results in subgoals and deflections, we segment the trajectory and try to infer the cause of each trajectory deflection. Most pedestrians will choose to walk a straight line in an open space because people always choose the shortest path to get to their destination. In a crowded environment, pedestrians will change their trajectories because of interaction with others. Most instances fall into two categories. Pedestrians change their trajectories to avoid someone in front of them (such as the left of Fig. 1) or to return to their original trajectories after avoiding a pedestrian.

III. Problem Statement

We are given a scene with a set of m pedestrians, $P_j = \{P_1, P_2, \ldots, P_m\}$, one of whom is a focus agent F. Each pedestrian j is observed to follow a time-parametrized trajectory $f_j : \mathbb{R} \rightarrow \mathbb{R}^2$ in the interval $0 \leq t \leq t_f$. The focus agent's trajectory f_F deflects one or more times for unidentified reasons.

We articulate the following problems. First, we consider in Section IV how to segment F's trajectory properly to capture subgoals. Second, in Section V we seek a model to explain F's deviations from a straight-line path at each subgoal by attributing them to a pedestrian *influencer*. Third, in Section VI we evaluate the model's performance. Fourth, we ask in Section VII how the model can be applied to the problems of predicting a focus agent's destination and predicting influencing agents that are outside of the observer's field of view. Finally, we discuss the limitations of this method and summarize our research in Section VIII and Section IX.

IV. Trajectory Segmentation and Labeling

We focus here on pedestrian navigation in environments with low levels of obstacle clutter. Therefore, any deflection of a pedestrian's trajectory is usually due to collision avoidance with other pedestrians. Thus, the key question is which pedestrian is influencing the focus agent's trajectory when the deflection occurs. A consistent and rigorous method is needed for determining where the deflection occurs in an arbitrary focus agent's trajectory.

There are many general methods for trajectory fitting. For example, Lee and Xu [9] present a derivation for a spline smoother that considers local velocity information. Such a method captures both the signal and measurement noise. We therefore fit a trajectory with straight lines, in order to fit the social model of piecewise-predictable trajectories. The sequence of lines in turn exposes the focus agent's intention.

In this section, we describe our technique of curating the ETH overhead pedestrian data from Pellegrini, et al. [5], but the techniques apply to any similar data set.

A. Trajectory Segmentation

In the segmented representation, the time-parameterized trajectory of F will be approximated by n piecewise linear segments $L_i : 1 \leq i \leq n$ bounded by n+1 points $D_i \in \mathbb{R}^2$ at times T_i for $1 \leq i \leq n+1$, each corresponding to a subgoal or deflection point (Fig. 1 right). Meanwhile at each point D_i, the pedestrian i occupies position $f_j(t_i)$.

The continuum trajectory $f_j(t)$ of agent j is sampled in the ETH dataset at 2.5 Hz, giving positions in meters of all agents in the scene for each time step. The objective is to segment the trajectory of each focus agent such that piecewise linear segments approximate the trajectory well. If we had access to the continuum trajectories, the process would be simple:

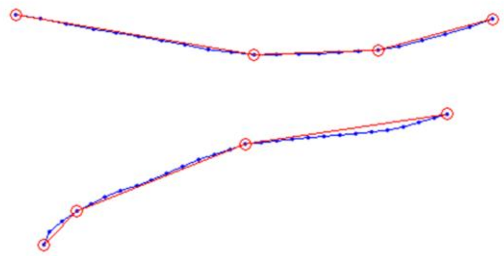

Figure 3. Two fitting-lines of pedestrians' trajectories. Blue dots are tracking points and red circles represent subgoals calculated by fitting method. When the trajectory has a part with small radian, this method will fit it inaccurately.

1. find all the inflection points, i.e. $\frac{d^2}{dt^2} f_j(t) = 0$
2. filter the inflection points to ensure they are sufficiently different, by removing one neighbor whenever adjacent inflection points differ by less than a threshold (we used 6°)
3. perform k-means clustering of points in $f_j(t)$ to the filtered inflection points
4. define a subgoal wherever two clusters meet, and connect consecutive subgoals by lines.

Here we briefly explain how we approximate the above algorithm on the dataset with sampled positions. To filter out small noise, we fit a straight line in a sliding-window fashion to every three consecutive points using linear regression, yielding a vector $A = [A_1, A_2, \ldots, A_{d-2}]$, comprising the sequence of angles between each line and the global x-axis. The idea is to find the local extrema in the angles. To do so, we construct skew-symmetric matrix U of the differences between pairs of angles $\left(A_p, A_q\right)$. We mark any element in this matrix, whose four neighbors are either all greater or all less than it is a local extremum. Selecting these cells from the matrix is the discrete equivalent to solving for a derivative equaling zero. Each element U yields a pair of indices (row and column) each of which is a potential extremum. The set of all such candidate extrema indices is collected and sorted. From here, indices are eliminated by thresholding if they differ by less than 6°. Finally, all position samples in the trajectory are clustered using k-means, and the clusters form straight lines connecting the subgoals.

This approach solves the trajectory's fitting-line problem successfully if the trajectory is approximately piecewise linear (upper Fig. 3). As a global fitting method, it won't be influenced by subtle noise and we can adjust fitting degree by controlling threshold Q. But if the trajectory changes gradually with small curvature, the approach won't work well because it is difficult to fit arc with line segments (lower Fig. 3).

B. *Labeling*

We labeled by hand the influencers of focus agents by watching the video. Two rules guide our interpretation of cause and effect for agents whose intentions could be easily discerned. Outliers such as wandering agents, static agents, incompetent agents and groups were excluded.

The primary goal of labeling is to discern the reason that pedestrians change their trajectories. For every pedestrian, a

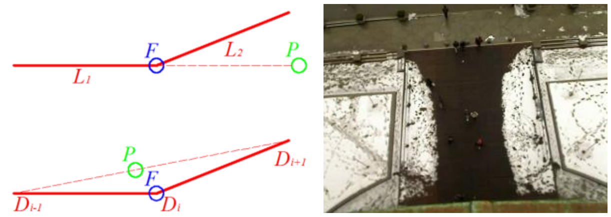

Figure 4. Left: Two labeling scenarios: (top) in the avoiding case, *F* deflects to avoid pedestrian *P* (bottom) in the returning case, *F* finishes avoiding *P* and moves to the next goal. Right: the training dataset [10].

deflection may be caused by a static obstacle in the environment, by distraction, or a special pedestrian. If there's a special pedestrian influences the trajectory of focus agent, then record the number of the pedestrian and take him or her as an influencer.

The hardest part of labeling process is how to ensure the objectivity and uniformity through it. When the deflection is caused by another agent, it typically falls into one of two situations: (a) focus agents turn to avoid collision when they find another pedestrian in front of them (avoiding case); (b) focus agents deflect after passing another pedestrian who was blocking their way (returning case). These two cases are most often separate aspects of the same situation -- people usually avoid first when they find pedestrians obstructing their path and then return to the original direction after passing the blocking pedestrian.

From these two cases, we can conclude two rules for judging influencers (Fig. 4): (a) if a pedestrian *P* is in front of the focus agent F and the original fitting-line L_1 would lead to a collision, then we regard the front pedestrian as the influencer; (b) if, when deflection D_i happens, another pedestrian blocks the straight line from subgoal D_{i-1} to subgoal D_{i+1}, then we regard the blocked pedestrian as the influencer.

According the two rules above, we label the ETH data [10] by watching the behaviors of every pedestrian in the video. There are 420 agents in the dataset. Of these, 178 trajectories are valuable samples. The others are missing data or have no interaction with other pedestrians.

C. *Coordinate Transformation*

To observe the relative position between the focus agent and influencer at the same time with subgoal D_2, we need to transform all the focus agents with one infection point to a same coordinate system. We transform the first fitting line L_1 to negative Y-axis to make sure focus agent come from -Y to +Y and move infection point D_2 to (0,0). Influencers' positions will be transformed with the same matrix and the agents with left bend need to be mirrored to right bend.

Now it's a map with respect to focus agents' trajectories and the influencers' positions when the subgoal D_2 happens (Fig. 5). Many influencers gather around (2,0). People always turn after passing another pedestrian in the direction they're going to turn, just in case they will confuse others when they turn. Some other influencers appear in the front of focus agent with obvious explanation. If we explain these two outcomes according to the avoiding and returning cases in labeling process, they should behave comparative quantity as they come from a same situation. The critical reason is the limited

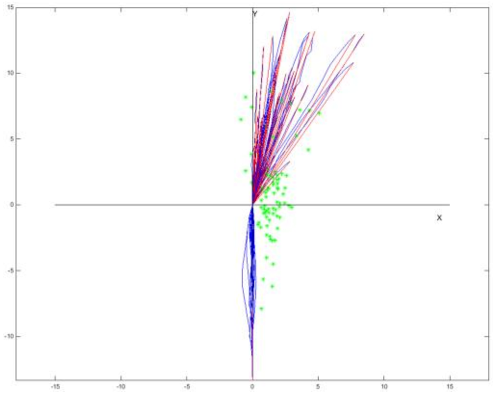

Figure 5. The positions of influencers when the corresponding focus agents turn. The green star is the influencer, the blue line is the focus agent's trajectory, and the red line is the fitting-line. Most influencers gather around (2,0) and others cluster in front of the focus agent.

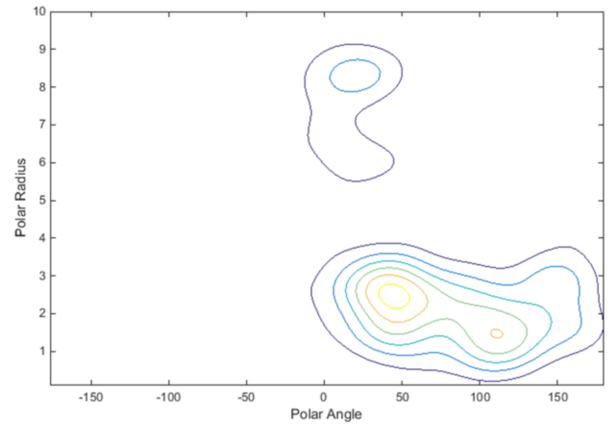

Figure 6. The contour of the influencer probability distribution when the focus agent turns. The horizontal axis is polar angle, and vertical axis is polar radius. Most influencers gather in the inner side of the turn, but a minority of other influencers gather in the front of the focus agent.

view of our sample video, the influencer's position in the avoiding case may be off screen, whereas the returning influencer always appears on screen. In other words, the distance between focus agent and influencer in the avoiding process is so long that we often cannot observe the influencer in the view. It's also an inevitable defect of this model caused by finite view of fixed camera.

V. Influencer Recognition Model

After the transformation, we obtain a distribution over the influencers' location in the frame of the focus agent, as represented by a 2-D kernel density estimator (KDE). The two dimensions are bearing and range. Furthermore, a video for visualization show why pedestrians deflect in their trajectories apparently.

We cannot regard pedestrians as moving obstacles on the account of intention, and they react intelligently [1]. Their trajectories have a few deflection points to divide trajectories. The deflections usually arise from the existence of spatial influencers (e.g. moving pedestrians). In its simplest form, the question we seek to answer is which pedestrian is the influencer of the focus agent in an obstacle-free environment.

A kernel density estimator (KDE) is a non-parametric method to estimate the probability density function of a random variable. KDEs are applied widely when no parametric model distribution is known to be a good fit. A one-dimensional KDE is defined as

$$f_h(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \quad (1)$$

However, pedestrian data are positions, which are two-dimensional variables. For two-dimensional KDE, the hardest problem is how to select the metric $x - x_i$. We can divide this problem to two parts: selecting a coordinate system, and designing an equivalent two-dimensional metric. Cartesian coordinates are conventional and intuitive. However, the Euclidean distance metric in Cartesian space incorrectly places influencers on all sides of the origin close together. Consider that such a metric would regard influencers near the origin in the first and third quadrants to be proximal, whereas our experience and results indicate they should not be confused. Typically, pedestrians with rightward deflections will not be influenced by a pedestrian in the left rear. (No such examples were found in the ETH dataset). We instead adopt a polar coordinate system, taking a modified Euclidean metric,

$$m(\rho_h, \theta_h) = (\,\alpha(\rho_h - \rho_i)^2 + \beta(\theta_h - \theta_i)^2)^{1/2} \quad (2)$$

Where ρ_i and θ_i are polar radius and polar angle of grid points, and ρ_h and θ_h are polar radius and polar angle of sample points. α and β are weight of polar radius and polar angle. By controlling α and β, we can adjust the proportion of the two variables.

Meanwhile, the selecting of the kernel function K and smoothing parameter h also influence the KDE. Based on our experience, we chose a Gaussian function as kernel function, and we set the smoothing parameter h to 0.05. The two-dimensional KDE formula is

$$f(\rho_i, \theta_i) = \frac{1}{0.05n}\sum \frac{1}{\sqrt{2\pi}} exp(-\frac{\alpha(\rho_h - \rho_i)^2 + \beta(\theta_h - \theta_i)^2}{0.1}) \quad (3)$$

In addition, the edge effect in a KDE can impair performance. The estimate distribution near the frontier of the variable's range will be incomplete if there are many data close to the boundary. The periodicity of angle eliminates the boundary in one dimension. To eliminate the influence of edge effect, we expand the data range to [-360°, +360°] and keep the estimating range as [-180°, +180°]. Note that there is seldom an influencer located directly behind a pedestrian, so the discontinuity at 180° is not a problem. Similarly, the edge effect for polar radius is insignificant because there are few samples near 0 m or 10 m.

After plugging in the position of the sample points in Eq. 3, we can obtain influencers' probability distribution model in the polar coordinate system and its contour. In Fig. 6, the horizontal axis is the bearing to the influencer in the focus agent's reference frame (as in Fig. 5). The vertical axis is the range from the focus agent to the influencer. The bulk of the probability mass is around two meters and 45 degrees. The contour's behavior is consistent with the cases previously discussed.

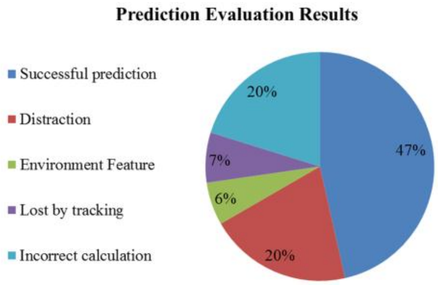

Figure 7. Prediction results as percentages. The most frequent outcome was a successful prediction. Distraction indicates failures because the focus pedestrian was not paying attention to surroundings or was otherwise not acting rationally. In roughly 20% of test cases (incorrect calculation), the model failed to predict the correct cause. Environment features are static obstacles not captured by IRM.

VI. Evaluation

To calculate the prediction accuracy of model, we contrast another dataset of ETH with this model. First, we select eligible pedestrians as focus agents with following condition:

a. The displacement should be bigger than 4 meters;

b. Their turning angle should be bigger than 5 degree;

c. They shouldn't gather to a group;

d. Their fitting line should be correct.

For every eligible focus agent, we put bearing and distance of other pedestrians into Eq. 3. The pedestrian with biggest probability should be the influencer to focus agent. There are 99 eligible samples totally (Fig. 7). Of these, 46 samples are predicted successfully; 20 samples are unpredictable as distraction; 6 samples fail because they are influenced by static obstacles; 7 samples' real influencers are not in the view or tracked by the dataset; 20 samples are predicted incorrectly due to inaccuracies in our trained model.

The result is not ideal for several reasons. First, most errors caused by calculation derive from treating equally between avoiding and returning cases. But avoiding case happens when interaction begins while returning case happens when interaction ends. The result of the two cases may be influenced by condition changing (finite view of camera), so this outcome will make the model inaccurate and non-adaptive in different environment. Second, people always predict the future position of front pedestrians in avoiding case. Actually avoiding pedestrian is affected by future trajectory of front pedestrians, IRM doesn't consider time sequence and it will lose influencers in the front sides of focus agent. Last, the training and testing datasets of IRM must be carried in a low-obstacle environment. We haven't found sufficient dataset for our method, and it brings inadequate robustness for IRM.

We also visualize this model on testing video with symbols. The focus agent is shown by blue cross and different parts of trajectory are shown by different colors. The squares mean other pedestrians and their lengths correspond to their probabilities to be influencers. Finally, the biggest probability pedestrian is labeled by red square and others are labeled by green squares (left of Fig. 8).

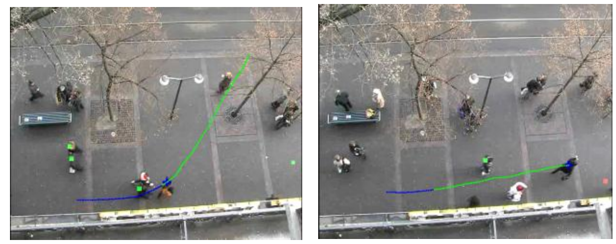

Figure 8. The visualization of the influencer model. The blue and green lines indicate two segments of the trajectory, each with consistent intention. The red square indicates the most likely influencer calculated by the influencer model (left). Sometimes, the model predicts an out-of-view pedestrian as designated by a red square at the edge of the screen (right).

VII. Applications

A. Destination Prediction

According to the method proposed above, we illustrate the reason leading to pedestrian's bend in low-clutter environment. Furthermore, there is a deeper inspiration to explain what drives pedestrian to avoid a special pedestrian or obstacle. Presume that a focus agent's trajectory is influencing by another pedestrian, this pedestrian must block the way of focus agent. As the pedestrians are always driven by their intention, we can infer that focus agent intend to reach the area blocked by the pedestrian.

Combining this idea and our influencer recognition model, it's possible that we can predict the destination by influencer's position. As the avoiding case shown by Fig. 9, the most crucial part is to ensure when the focus agent decides to turn. When the bend occurs, the influencer can be recognized by IRM. According to influencer's position, setting a personal space of the influencer can make clear which area is avoided by focus agent. Finally, the possible destination area can be inferred by the subgoal of focus agent and the influencer's position. In addition, introducing other variables (e.g. bearing) will be beneficial to make the result more accurate.

At present, it's an incomplete approach as uncertainty brought by real-time requirement, but it will be an outstanding prediction model in open space if we could improve the precondition and combine other information. In terms of time duration, destination prediction is a long-term and global method compared with short-term trajectory simulation.

B. To Discover GSEs or SSEs

In a surrounding with people and static obstacles, two kinds of effect interact with pedestrians: general spatial effects (GSEs) and specific spatial effects (SSEs) [8]. GSEs are detectable and concrete obstacles (e.g. trees) while SSEs is abstract implication in the mind (e.g. sidewalk). The analysis of abnormal trajectories with no apparent reason to bend will inspire us to obtain latent information in current environment.

After observing a focus agent with changing trajectory, assume an invisible GSE in the front of focus agent and out of the view. If this invisible GSE has biggest probability under the influencer recognition model, then we can infer that there is an invisible pedestrian influencing focus agent's trajectory out of view (Fig. 8).

Furthermore, some SSEs' characters can be inferred by long-term learning about abnormal trajectories. For example, pedestrians always bend to avoid dangerous area when they are passing cross, where pedestrians pay more attention to cars rather than persons. Collecting such trajectories without reasonable explanation may help understand where is rejected by pedestrian's subconscious. But it's difficult to conduct long-term learning only with IRM. For SSEs' research, another method needs to be incorporated to understand environment feature better. Learning motion patterns of people or building cost-map of space [6, 7] maybe good directions.

VIII. Limitations

The method presented here provides an important first step towards addressing the problem of understanding pedestrian intent during obstacle avoidance. Even so, the method presents several limitations.

Most significantly, the model is currently trained only on pedestrian influencers, not static obstacles. The model may still predict static obstacles as influencers if aware of them, but the point approximation no longer applies to larger obstacles. People tend to walk along large obstacles (e.g. a curb or wall) with smooth curves (Fig. 4). An environment with such features will greatly influence the trajectory in a manner that IRM cannot capture. A similar problem arises when a pedestrian is walking while distracted.

IRM only picks one influencer at a time to attribute the focus agent's behavior. People standing in groups often collectively influence the focus agent, as do combinations of unrelated people inadvertently standing in configurations that create complex influences on the focus agent. Such configurations would require IRM to account for multiple subgoals and recognize multiple influencers simultaneously to properly model the situation.

IX. Conclusion and Future Work

In this paper, we propose the influencer recognition model (IRM), a method to explain why pedestrian changes their trajectory in low-clutter environment based on a 2-D KDE learned from real pedestrian data. We show that IRM is able to identify influencers correctly in approximately half of encounters. It can also impute influence to a hypothetical pedestrian that is out of view of the observer.

As this method confuses two different cases without time series, its evaluation is not satisfactory. We also propose two promising applications: analyze GSEs and SSEs by abnormal trajectories and predict possible destination of focus agent by this model and the position of the influencer.

There are many directions for improvement. The primary objective of this research going forward is to collect a large

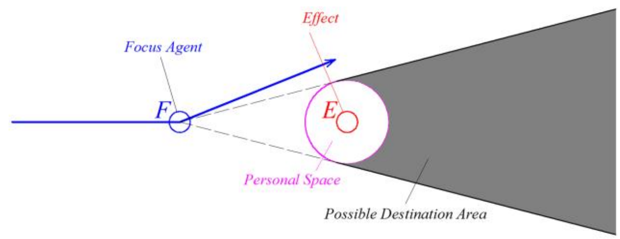

Figure 9. When the focus agent decides to turn, it's the approach to predict possible destination area by influencer recognition and its position. Blue line means fitting-line of trajectory, and purple circle illustrates the supposed avoidance area in focus agent's mind.

pedestrian dataset and annotate it with trajectories calibrated in ground coordinates (not screen coordinates, as many datasets do). Given the scale of data needed to train more sophisticated models using modern machine learning methods, this objective may be best achieved by computing homographies for web cams that stream pedestrian scenes.

Another objective is model improvement. The focus agent's returning and avoiding processes must be analyzed separately. To enhance the predictive power of the model, we would like to introduce additional variables such as the turning angle of focus agent. Another promising application of IRM is the prediction of a focus agent's destination.

References

[1] Mavrogiannis, Christoforos I. and Ross A. Knepper, "Decentralized Multi-Agent Navigation Planning with Braids", *Workshop on the Algorithmic Foundations of Robotics*, San Francisco, December, 2016.

[2] Gergely, György, and G. Csibra. "Teleological reasoning in infancy: The naıve theory of rational action." *Trends in cognitive sciences* 7.7 (2003): 287-292.

[3] Helbing, Dirk, and P. Molnár. "Social force model for pedestrian dynamics." *Physical Review E Statistical Physics Plasmas Fluids & Related Interdisciplinary Topics* 51.5(1998):4282.

[4] Tamura, Y, et al. "Development of pedestrian behavior model taking account of intention." *IEEE/RSJ International Conference on Intelligent Robots and Systems* IEEE, 2012:382-387.

[5] Pellegrini, S, et al. "You'll never walk alone: Modeling social behavior for multi-target tracking." *IEEE International Conference on Computer Vision* IEEE, 2009:261-268.

[6] Bennewitz, Maren, et al. "Learning Motion Patterns of People for Compliant Robot Motion. " *International Journal of Robotics Research*24.24(2005):31-48.

[7] Ziebart, Brian D, et al. "Planning-based prediction for pedestrians." *IEEE/RSJ International Conference on Intelligent Robots and Systems, October 11-15, 2009, St. Louis, Mo, Usa* DBLP, 2009:3931-3936.

[8] Chung, Shu Yun, and H. P. Huang. "A mobile robot that understands pedestrian spatial behaviors." *IEEE/RSJ International Conference on Intelligent Robots and Systems, October 18-22, 2010, Taipei, Taiwan* DBLP, 2010:5861-5866.

[9] Lee, C., and Y. Xu. "Trajectory fitting with smoothing splines using velocity information." *IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA* IEEE Xplore, 2000:2796-2801 vol.3.

[10] http://www.vision.ee.ethz.ch/datasets/