Real-Time Certified Probabilistic Pedestrian Forecasting
Author's Response

Henry O. Jacobs, Owen Hughes,
Matt Johnson-Roberson, and Ram Vasudevan

May 5, 2017

We would like to begin by thanking the reviewers and the editor for their careful reading and review of our paper. The comments provided were incredibly helpful and insightful, and have strengthened the paper considerably. Based on the recommendations of the reviewers and the editor, we have made several modifications to the paper. The changes made are summarized in detail below and a highlighted manuscript is attached that indicates additions in red.

Reviewer 1

1. **This paper presents a novel real-time probabilistic forecasting method for pedestrian trajectories via observing the historical trajectories in a particular scene. The problem is well motivated, formulated and handled. The reported experimental evaluations show pivotal improvement against the current state-of-the-art.**

   Thank you for the positive assessment.

2. **With the current form of the results it is hard to compare the improvements in terms of accuracy gained. This can be overcome by including results such as physical distance between predicted paths and ground truth observations. Such matrices are provided in the baseline models [7, 9, 13] with accuracy values in meters.**

   Thank you for noticing the difficulty in comparing the improvements. We calculated the MHD from each point in the trajectory to sampled points from each distribution which is summarized in Figure 1 below.

   We feel that both plots add something to the analysis, so we've included the following passage to justify both our metrics.

   > In our analysis we sought a metric that measured the similarity between the predictor and the ground truth as well as the "safety" of the prediction. We used the area under the curve of the ROC curve as our measure of this. The ROC curve plots the rate of false positive detections against the probability of detecting a pedestrian, or in essence the quality of the detection versus the safety. The area under this curve is a standard measure of the quality of a predictor. Figure 2 shows the analysis of the AUC of each algorithm versus time. In addition, we used the Modified Hausdorff Distance (MHD) from the ground truth trajectory to a sample from the predictions at each time in order to provide a geometric measure of how accurate the predictions are. Figure 1 shows MHD plotted against time.

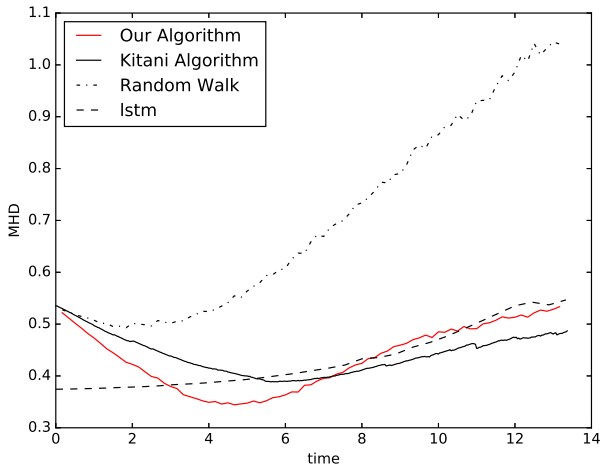

Figure 1: A comparison of the Modified Hausdorff Distance from the ground truth of the pedestrian to a 1000 point sample from each distribution. The method from [13] does very well at short time scales due to its confidence, but we outperform all other methods at intermediate times. At long timescales the MHD to the trajectory of most algorithms converges. Our method increases positional uncertainty with time, and so [7] outperforms us because they know the end point and we do not, and [13] places too much significance on the positions of other pedestrians at large time scales.

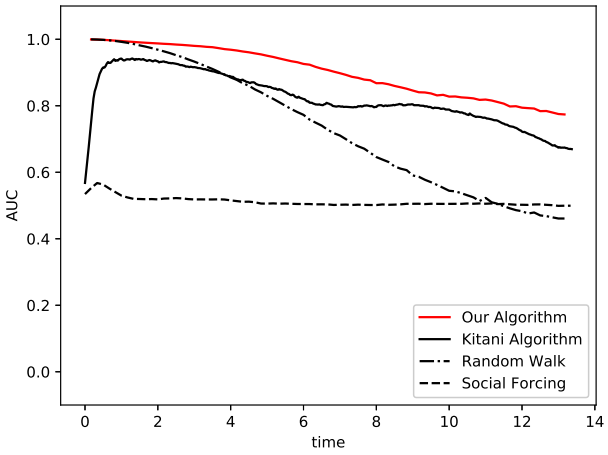

Figure 2: A comparison of the AUC of the various algorithms. Note that the initial dip in the performance of [7] is due to their confidence in their initial estimate. We sampled the S-LSTM model 100 times in order to give them the best chance in this analysis, but their confidence combined with the over-reliance on social forces at moderate-to-large lowered their performance.

3. **How the model can incorporate the interactions among pedestrians (i.e group motion) and the influences from neighbouring pedestrians (for instances such as collision avoidance)?** Thank you for this insightful question. Social forces are a natural avenue for future work, and so we included the following paragraph in the paper.

> Though including social forces into the model is out of the scope of this paper, incorporation could begin as follows. In the current motion model with a single vector field, the acceleration of the ith agent is given by $\ddot{x}_i = s^2 DX(x_i) \cdot X(x_i)$. Incorporating a social force F_i acting on the ith can be done by instead asserting $\ddot{x}_i = s^2 DX(x) \cdot X(x) + F_i$. Usually $F_i = \sum_j \nabla U(x_j - x_i)$ where U is an interaction potential [Helbing and Molnar].

4. **Presentation and organisation The motivation behind choosing start/end points based clustering is not clear. Does clustering based on start/end points help to identify different motion models? Isn't it better to cluster considering the entire trajectory?** This is a relevant question, thank you. As stated in Morris and Trivedi (below), our choice of clustering technique is not nearly as important as the chosen distance metric. Our success with distance functions that took a whole trajectory was limited on the data we tested, particularly when compared against our method. We have several hypotheses as to why this is true. For one, the spatial scale of the data we test against showcases important features (e.g. curb cuts and storefronts), which lead to clustered start points and end points. Similarly, we tested against high quality data in which trajectories with irreconcilable occlusions were rare and thus could be omitted. See the comment below for the text we added to the paper.

5. **Do you cluster different scenes provided in campus dataset [13] (i.e Valley, Fangron) separately or all together? When motion models are learnt, are they learnt in scene specific manner or altogether?** Thank you for noticing that this is unclear. Our algorithm trains on each scene independently, so clustering, vector field learning, and learning the potential functions are all done for each Valley, Fangron, etc. We've added the following text to the paper:

> We did a 2-fold cross validation by using 20% of the data for testing and the remainder for training within each scene. We learned separate collections of vector fields and model parameters for each fold on all of the four scenes on which we tested.

6. **The motivation behind choosing the affinity propagation clustering algorithm is not clearly stated. What is the distance measure used (i.e Euclidian, Ö) ? Not specifying such information makes reproduction of this work not feasible.**

This was an oversight on our part, thank you for noticing it. The distance used is actually a custom distance function on $\mathbb{R}^4$, which disregards the involution $(x_1, x_2, x_3, x_4) \to (x_3, x_4, x_1, x_2)$, and so doesn't care about parity of trajectories. We have included the following clarification:

> We then cluster in $\mathbb{R}^4$ using Affinity Propagation[17] and a custom distance measure defined by $d((x_1, x_2, x_3, x_4), \mathbf{y}) : \mathbb{R}^4 \times \mathbb{R}^4 \to \mathbb{R} = \min\{d_e((x_1, x_2, x_3, x_4), \mathbf{y}), d_e((x_3, x_4, x_1, x_2), \mathbf{y})\}$, for the euclidean distance d_e. This function measures the distance between the endpoints irrespective of their ordering. The scale of the datasets we tested on had large enough spatial scale that clustering based on endpoints captured people moving from destination to destination, e.g. from a storefront to the sidewalk at the edge of a scene. On our data, other distance measures from [Morris and Trivedi] and [Lee 2007] didn't identify coherent motion models. come up with a good story for why these fail so spectacularly.

7. **Adequacy of Citation I believe literature review section can be improved using a sub section on social force models, which is extensively applied for pedestrian trajectory forecasting. Helbing and Molnar, 1995; Koppula and Saxena, 2013; Pellegrini et al., 2010; Yamaguchi et al., 2011; Xu et al., 2012; Wang et al. 2008; Hospedales et al. 2009; Emonet et al. 2011; Yi et al. 2015.**

   We agree that this adds to the paper, thank you for the suggestion. We added a section to our literature to reflect the relevant work. Other classes of algorithm focus on predicting social behaviour based on social interactions of people [yamaguchi] incorporates social grouping into a discrete-time dynamical system model of pedestrian movement using a learned energy function (minimizing over energy function to find velocity, local minimum) to determine the dynamics that incorporates collision/group details, and learns social groups using an improvement on [Pellegrini]. Learns weight parameters by throwing simplex method at an LP version of the problem and hopes it converges on a local min. Scene obstacles are modeled as pedestrians. Destination forecasted by learned SVM on destinations of observed trajectories. Uses a classifier to determine groups. Mentions: Mentions Helbing, similarity to physics

   Notes on CRFs: "The CRF formulation indirectly encodes a behavioral model. Our focus is to build an explicit behavioral model which can exploit the rich behavioral context in social interactions,"

   "The assumption of discretized choice allows efficient prediction with analytical solutions despite the large number of factors considered in the model [1, 10]. However, due to the nature of the discretization, the behavioral prediction tends to show artifacts when metric is continuous."

   [pellegrini: you'll never walk alone] "An advantage of continuous model is the flexibility of constructing complex models, however, previous work focuses on individual motivation of the behavior [you'll never walk alone, 11], and the only social context is collision avoidance." [As in [you'll], our model assumes that each pedestrian makes a decision on the velocity based on various environmental and social factors in the scene] Yamaguchi uses collision avoidance from [you'll]

   [pellegrini: improving]: Yamaguchi attempts a simpler method of social grouping, with better computation time.

   [Yi] does it by learning a scene energy map, a moving pedestrian energy map, and a stationary energy map by maximizing likelihood, then computes trajectories by minimizing this map between the start and the end point. These are learned by using gradient ascent to maximize the likelihood of the data under the energy maps. Requires endpoint. No mention of real-time, or run-time.

   Mentions: [emonet]: Spatio-temporal dependency on motion patterns could be included in topic models [emonet] - [31]. [hospedales] included in last [helbing] and abnormal event detection [helbing]. [zhou] It can be used for various applications including pedestrian walking path prediction [1], [zhou], "Previous studies [zhou], [17][19] have shown that the walking behavior of an individual can be influenced by a variety of factors including scene layout (e.g. entrances, exits, walls, and obstacles), inter-person variations on the choice of source and destination, and interactions with other moving pedestrians. " "A mixture model of dynamic agents (MDA) is proposed by Zhou et al. [2], which can learn parameters automatically. Typical agent-based models also include the self-driven particle model [43], and the reciprocal velocity obstacle model [44]." "Most previous methods [2], [17], [18], [40][42] are local models. Decisions are made based on local environments and the interactions with nearby people. These models may make reasonable decisions when there are not so many people in the scene and the scene is not

that complex." ". Two existing approaches, i.e., the MDA model [zhou]and an unsupervised visual prediction approach (UVP) [54], are used for comparison." " This is because only the motion pattern information are modeled for MDA [2] and UVP [54], while the influences of stationary objects (e.g. stationary crowd groups) are ignored" [wang: unsupervised] "traffic flow segmentation [3][5],"

[Emonet] uses heirarchical dirichlet processes on optical flow of downsampled video, discretize to 8 cardinal directions, then find the most common low-level words, and do heirarchical dirichlet processes to generate topics there. Can't predict individual pedestrians? [Hospedales 2009] Uses an improvement on LDA, does online topic discovery in real time. Adds heirarchical modeling, to reflect how simple actions can be turned into behaviours. Doesn't have explicit motion model though, Doesn't necessarily work for a single pedestrian? [helbing and Molnar] describes pedestrian behaviour according to an interaction potential, causing pedestrians to avoid each other. Doesn't include a complex motion model.

[alahi] uses LSTM to learn pedestrian behaviours, and predict based on them. [zhou] [wang1: Trajectory analysis] [wang2: Unsupervised Activity Perception] [Xu] [Pellegrini1:Improving data] [Pellegrini2: wrong turn] [Pellegrini3 : you'll never walk alone] [Koppula]

Along different lines, many authors have focused on using social interaction as a main part of their motion model. [yamaguchi] incorporates social grouping into the energy function used to determine the velocity of a discrete-time dynamical system, while [Yi] incorporates pedestrians into an energy map, and calculate the prediction between two points as the lowest cost path. Both of these approaches [helbing and molnar] introduced the idea of an actual Social Force, which was shown to be effective on modern data by [], and [pellegrini 2010]. [pellegrini 2010] uses a stochastic linear trajectory avoidance. [emonet] and [hospedales] both operate on raw video data, using Heirarchical Dirichlet Processes and an extension of LDA to find behaviours in scenes. Though these methods are good at detecting anomalous behaviour, without an actual underlying motion model their individual predictive power is limited. robicquet uses a Long Short-Term Memory network to learn and predict social behaviours.

8. **Furthermore a comprehensive review on trajectory clustering algorithms is required. This should be used to motivate the reason of choosing affinity propagation algorithm in section 4. Morris and Trivedi 2009; Giannotti et al. 2007; Lee et al. 2007; Ester et al. 1996**

   Thank you for this suggestion and for the list of relevant titles. We included a note about different algorithms as well as a justification for why we chose our metric. The note can be seen in point 6 above.

Reviewer 2

1. **Keep the good work going. Interesting approach.**

   Thank you for the kind comments.

2. **Need to be clearer and dive a little bit deeper into the technical approach.** Thank you for this criticism. We hope we adequately addressed it by strengthening the description of our method for learning the motion model, as well as quantifying the way that we tested our data. We also changed the language below to better reflect that the specific implementation we chose is an example of how to use the larger framework.

Given the model established in the previous section, we describe an implementation to showcase one way the model can be applied to observational data.

3. **Need to use and apply more deep learning approaches such as social LSTM or other machine methods.** Thank you for the suggestion. We hope that we have addressed it by including a review of literature of social force models, including S-LSTM, as well as a section on how social factors can be integrated into our model. In addition, we compared against S-LSTM in addition to [7] from the paper. Thefigures drawing the predictions of the S-LSTM model are ommitted from the paper because like many of the agents, the S-LSTM predictions quickly exit the bounds of the scene, and so the visualization is identically zero. We changed the subtitle of the figure to reflect this, which is shown below.

An illustration of the predictions generated by the various algorithms. In this figure, the dot is the start point of the test trajectory, the diamond is the position at time t, and the X is the end of the trajectory. The likelihood of detection is depicted using the virdis color palette. Notice that the Random Walk is imprecise, while the predictions generated by the algorithm in [7] suffer from the inability of their motion model to adequately match the speed of the agent. The algorithm from [13] quickly passes beyond the boundaries of the scene, so the plots are ommitted here.

An illustration of the predictions generated by the various algorithms. In this figure, the dot is the start point of the test trajectory, the diamond is the position at time t, and the X is the end of the trajectory. The likelihood of detection is depicted using the virdis color palette. Notice that the Random Walk is imprecise, while the predictions generated by the algorithm in [7] are unable to match the speed of the agent and choose the wrong direction to follow the agent around the circle. The algorithm in [13] passes beyond the boundaries of the scene, and so the plots are ommitted here.

Reviewer 5

1. **It would be more interesting if the authors show whether the prediction algorithm would transfer the learned "knowledge" of forecasting in novel scenes.** Thank you for your comment. As it is, this is a shortcoming of our model, but we see it as an immediate avenue for future work. As can be read below, the environmental decomposition that the segmentation algorithm in [12] returns as well as an analogue of the methods [11] proposes for characterizing and transferring our vector fields from one scene to another.

As future work, we propose using scene segmentation from [12], as well as the semantic context descriptors and routing scores from [11] to show how vector fields can be transferred to novel scenes. It appears that the low-order parameterization of our model, and unit-length vector field assumption make it particularly amenable to the methods developed in [11].

2. **More statistically result is expected rather than analyzing "four different scenes", and comparing run-time by "averaged across several agents and scenes" (how many?).**

Thank you for noticing this. We've clarified the papers in the passages listed below to clarify the list of scenes we ran in comparison, and how many agents total were run. We clarified that the set of agents used to evaluate were the same that were used to determine run times. Note that we attempted to compare on all of the scenes in the Stanford Drone Dataset, but the

difficulty in getting the other models to converge in a reasonable amount of time prevented us from doing so.

Our analysis was conducted on the Coupa, Bookstore, Death Circle, and Gates scenes from the dataset from [13], with a total of 142 trajectories analyzed.

The run time per frame for each algorithm was generated using the mean run time for 400 frames, averaged across all of the data used in the quality analysis.

3. **Besides, it is better to explain the meaning of ROC and AUC in section IV.**

This is a necessary clarification, thank you for the suggestion. We have included a description of ROC and AUC, as well as a justification for why we chose to evaluate the methods using this criterion. You can read the text that we added to address this in the second point made by Reviewer 1.