Atlas: a Benchmarking Tool for Human Motion Prediction Algorithms

Andrey Rudenko[1*], Wanting Huang[2*], Luigi Palmieri, Kai O. Arras
Robert Bosch GmbH – Corporate Research
Stuttgart, Germany
ga86was@mytum.de, {andrey.rudenko, luigi.palmieri, kaioliver.arras}@de.bosch.com

Achim J. Lilienthal
Mobile Robotics and Olfaction Lab
Örebro University, Sweden
achim.lilienthal@oru.se

***Abstract*—Human motion trajectory prediction, an essential task for autonomous systems in many domains, has been on the rise in the recent years. With a multitude of new methods proposed by different communities, the lack of standardized benchmarking and objective comparison between them has been a major limitation for assessing the capabilities of the state-of-the-art systems. In this paper we present the Atlas benchmark which encompasses a large variety of heterogeneous datasets, representing usual human motion behaviors in different places and cultures. The Atlas benchmark offers tools, such as metrics, data preparation and filtering, calibration and visualization to overcome several limitations of existing benchmarking, thus sustaining the enduring development of better algorithms.**

I. Introduction

Benchmarking motion prediction algorithms is a challenging task. The evaluation outcome can be affected by various factors, and the properties of the methods can sometimes be exposed only in elaborate experiments. For instance, such factors include prediction horizon, which defines how far into the future predictions are made, and the procedure used to extract testing scenarios from raw datasets (labeled detections). Even when evaluating the simplest constant velocity model using the same dataset, metrics and prediction horizon, the evaluation results still vary in [1] and [2] due to the differences in testing scenario generation and data pre-processing.

In this paper we present the Atlas benchmark as the first step towards automated benchmarking and evaluation of the motion prediction methods with systematic variation of parameters. Atlas includes heterogeneous datasets of human motion trajectories, and is capable of automatically extracting valid testing scenarios, interpolating, downsampling and smoothing the missing and noisy detections. Compared to the prior art (e.g. TrajNet++ [3]), it offers many tunable parameters like the observation period and prediction horizon, import of semantic maps and other relevant information such as the coordinates of goals in the map, evaluation of the probabilistic prediction results, and robustness testing with added noise to the original data. Unlike TrajNet++, our benchmark is especially suited for studying how prediction parameters influence the results, in contrast to fixing the main parameters for producing ranking scores in a specific *challenge*.

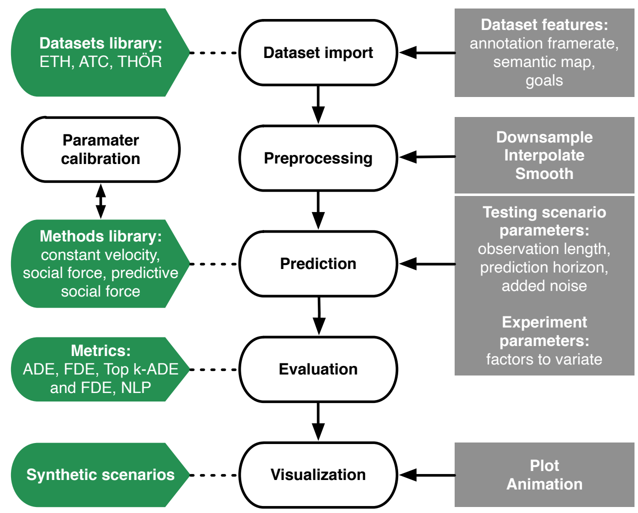

Fig. 1. Atlas benchmark design

II. Background

Generally speaking, trajectory prediction aims to estimate future positions of a moving agent within a certain time horizon with a deterministic or stochastic state hypothesis. Typically, a motion predictor uses as input the current (x, y) state of the agent (or a history of observed states), possibly augmented with the current state of the environment (or history thereof). The environment is represented by the states of other moving agents, a 2D map of static obstacles $\mathcal{M}$ and possibly also surface semantics $\boldsymbol{f}(\mathcal{M})$. For evaluation of a motion predictor, a continuous flow of detections from the dataset [6]–[11] is converted into *testing scenarios*, where all detections between two frames are used as the observation history of length O, and the following T frames should be predicted and compared to the ground truth (GT) data. Metrics used to this end include geometric and probabilistic estimations of the distances between predicted and GT positions [12]. This outlines the main parameters of the evaluation: the dataset used, the extraction strategy for a testing scenario, observation and prediction intervals O and T, and finally the metrics.

[1] A. Rudenko is also with the Mobile Robotics and Olfaction Lab, Örebro University, Sweden
[2] W. Huang is also with the TU München, Germany
[*] Equal contribution

This work was supported by the European Union's Horizon 2020 research and innovation program under grant agreement No. 101017274 (DARKO)

The evaluation strategy, proposed by Alahi et al. [1] and formalized in the first *TrajNet* benchmark for motion prediction [4], has been adopted by many authors [13]–[21]. It uses the ETH and UCY datasets with fixed observation history $O = 3.2$ s and prediction horizon $T = 4.8$ s and the ADE and FDE geometric metrics. TrajNet does not include variability in the main parameters O and T, obstacles in the environment and any notion of prediction uncertainty or robustness.

An improved *TrajNet++* benchmark [3] uses several further datasets, and potentially can be extended with new ones (stored in json format). It includes the possibility to predict several discrete positions for each pedestrian in each step, but does not support other probability distribution representations. The main limitation here, however, is the rigidly defined testing parameters, which restrict the evaluation to the fixed $O = 3.2$ s and $T = 4.8$ s. Furthermore, the scenario extraction strategy only guarantees that in each scenario *one* target pedestrian has a complete track of requested $O + T$ consecutive positions. This contradicts the assumption, commonly made by many authors, that the history tracks for *all* pedestrians are available at the time of prediction [21]–[24]. TrajNet++ does not support obstacles and semantic information about the environment.

Based on these insights, we develop the *Atlas benchmark* with an automated procedure to extract testing scenarios from an arbitrary dataset with flexible O and T parameters. Atlas accepts occupancy and semantic maps as input, supports analytical and discrete uncertainty representation, and includes robustness experiments with added noise to the observed trajectories.

III. Our Benchmark Description

Fig. 1 outlines the design of our benchmark. The benchmark includes five main elements: data import, preprocessing, a prediction phase, evaluation and visualization tools. By explicitly interfacing the prediction module and scripting the experiments, our benchmark is suited for flexible and highly automated assessment of the motion prediction algorithms.

As the first step, the datasets and possibly additional information like the known goals in the environment, obstacle or semantic maps, are imported into the benchmark. Then, the raw data is preprocessed with downsampling to the user-defined frequency, interpolating the missing detections and trajectory smoothing. Once the dataset is ready, we extract the testing scenarios with the user-specified observation and prediction lengths. The observed histories of all people in the testing scenario, along with environment data, are explicitly interfaced as input to the prediction algorithm. The returned predictions are evaluated against the ground truth using several metrics. Finally, the prediction results can be visualized with plots or animations. Meta-parameters to control the data processing and experiments are stored in a separate yaml file, eliminating the need to modify and re-compile the code.

A. Datasets

The benchmark users can import any dataset in the specific json file format, compatible with TrajNet++ [3], which includes for each detection the time stamp, person id and position. The json dataset format also supports the obstacle and semantic grid maps, and the common goals in the environment, which may insight the possible destinations of people. Our benchmark currently includes the following three datasets:

i) **ETH** [6]: This dataset contains people detections from video data recorded outdoors in the ETH campus in two locations: ETH and HOTEL.

ii) **ATC** [9]: It is recorded in a shopping mall in Japan, representing therefore a large indoor environment with densely crowded scenes.

iii) **THÖR** [11]: This dataset captures human motion in a room with static obstacles.

These three datasets come from different countries, taken in different environments, which increases the diversity of the data, and allows comparing the prediction methods on different social and cultural contexts.

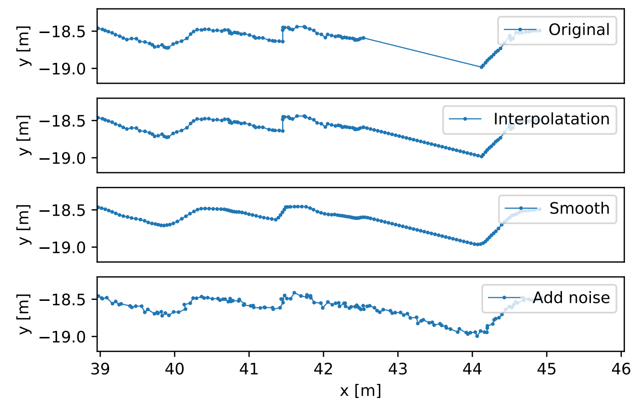

Fig. 2. Example trajectory from the ATC dataset, which shows the noise and missing detections in the raw data (*original* trajectory on the top). Our benchmark offers interpolation and smoothing to fix this, followed by adding a controlled amount of noise to test the robustness of the prediction algorithm.

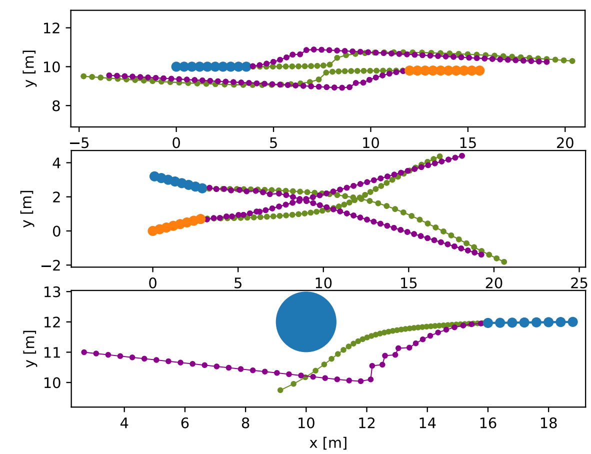

Fig. 3. Synthetic testing scenarios (top to bottom: *opposing*, *crossing* and *avoiding an obstacle*). The **blue** and **orange** dots show the observations of the two people, the **green** dotted lines show the social force predictions [25], and the **purple** dotted lines illustrate the predictive social force result [26].

B. Preprocessing

Raw datasets often include noise and annotation artifacts (e.g. missing detections) [11]. Hence, our benchmark offers interpolating and smoothing options in the preprocessing step. In addition, to check the robustness of the implemented models, Gaussian noise may be added to each detection. Fig. 2 shows the preprocessing steps applied to one trajectory in the ATC dataset. After detecting the missing frames in the original trajectory based on the average annotation frequency, we interpolate the points in the missing part of the trajectory. Then, a moving average filter is used to smooth the noise. Finally, random noise distributed as $\mathcal{N}(0, \sigma^2)$, where σ is inversely proportional to the frame frequency, can be added to each detection.

After the data preprocessing, Atlas generates the testing scenarios with the observation length O and ground truth for the following T frames. As the prediction quality may strongly depend on the observation length (in particular for intention estimation and when the person detection is noisy), it is critical that all people in the testing scenario are observed in each of the O frames. A testing scenario, along with the environment information, is passed to the motion prediction step.

C. Prediction

Our benchmark offers a direct interface to the prediction module, which is called at this step for the given testing scenario. This allows automated evaluation with a systematic variation of parameters, defined at the previous steps. For optimizing the hyperparatemers of the prediction methods, such as [25]–[29], Altas includes an interface to the SMAC3 optimizer [5].

Prior to benchmarking the prediction model on real data, the users can first validate their methods with several synthetic scenarios, which model fundamental interactions between people and the environment, e.g. individuals and groups walking in the opposite directions, crossing paths and navigating around hindrances (see several examples in Fig. 3). For instance, Fig. 3 (top) shows two people walking on a collision course towards each other. Their velocities are $1\ \mathrm{m\,s^{-1}}$ and the initial displacement in the y axis is 0.2 m. The frame frequency is $2.5\,\mathrm{Hz}$ and the observation length is 8 frames.

Our benchmark supports analytical, discrete and particle-based uncertainty representation for the prediction results. Discrete uncertainty is encoded in the grid map of the environment, separately for each person in each time step. Analytical uncertainty is represented with a mixture of Gaussians. Particle-based uncertain predictions are represented with a set of discrete samples. These options allow evaluating most existing prediction algorithms.

D. Evaluation

The Atlas benchmark supports geometric and probabilistic metrics. Geometric metrics include the *Average Displacement Error* (ADE), which describes the error between points of predicted trajectory and the ground truth at the same time step, and the *Final Displacement Error* (FDE), which computes the error at the last prediction step. Probabilistic metrics include the *Negative Log-Probability* (NLP), which computes the average probability of the ground truth position under the predicted distribution for the corresponding frame, and *Top-k ADE and FDE*, which compute the displacements between the ground truth position and the closest of the k samples from the probability distribution.

E. Experiments

On top of the datasets, metrics and pre-processing steps, in our benchmark we propose a set of experiments to study the prediction performance under the influence of various factors. These experiments allow systematic validation of parameters and help the users to gain a deeper insight into the applicability of the methods, in contrast to a limited insight contained in a single benchmark score. Due to the automated nature of our benchmark, the experiments are scripted with all parameters available externally in a yaml file.

1) Prediction accuracy conditioned on parameters: Observation length and prediction horizon are among the main factors, associated with predicting motion. The accuracy naturally degrades for further time instances, while longer observations may improve it overall. In Atlas it is possible to measure the accuracy of prediction conditioned on these two main parameters. We intend to add more conditioned experiments in the future, e.g. based on the number of people in the scenario.

2) Transfer experiment: A crucial part of evaluating a prediction method is testing its applicability in new environments outside the training data. Surprisingly, this experiment is most often overlooked in evaluation sections. In Atlas it is possible to script hyperparameter optimization in one dataset, and evaluate the resulting method in another. In the future we plan to extend this functionality to training.

3) Robustness experiment: For a system working in the real world, perception of the people's positions is often prone to noise, therefore the predictor must be robust to noisy input. One possible way to quantify robustness, implemented in Atlas, is by measuring accuracy on the testing scenarios, artificially adding increasing amounts of Gaussian noise to the initially noise-free data.

IV. Experiments with Local Interaction Models

Social force model [25] is a well-known approach for describing joint motion of people with promising results in the prediction domain [30]. A reasonable and popular choice due to its reliability, performance and simple implementation, the social force model suffers from inherent reactivity: the agents engage in passive collision avoidance only when in close proximity for the social forces to take effect (see Fig. 3). In reality, people adapt their trajectories to avoid collisions in advance. To correct this sort of behavior, the social force theory was extended with explicit collision prediction by a number of authors.

In this section we use the experiments in Atlas to compare the social force (*Sof*) with two popular predictive extensions: the model by Zanlunlgo et al. [26], abbreviated as *Zan* in

	Prediction horizon				
	Methods	1.6 s	3.2 s	4.8 s	8 s
ADE	CVM	0.10 ± 0.05	0.23 ± 0.13	0.40 ± 0.23	0.84 ± 0.57
	LIN	0.17 ± 0.09	0.34 ± 0.19	0.55 ± 0.35	1.02 ± 0.69
	Sof	0.10 ± 0.05	0.23 ± 0.12	0.39 ± 0.20	0.78 ± 0.45
	Zan	0.10 ± 0.05	0.23 ± 0.12	0.38 ± 0.19	0.76 ± 0.44
	Kara	0.11 ± 0.06	0.23 ± 0.11	0.38 ± 0.19	0.75 ± 0.44
FDE	CVM	0.19 ± 0.10	0.50 ± 0.28	0.90 ± 0.54	1.96 ± 1.44
	LIN	0.29 ± 0.16	0.66 ± 0.39	1.13 ± 0.73	2.25 ± 1.62
	Sof	0.19 ± 0.09	0.49 ± 0.26	0.85 ± 0.45	1.72 ± 1.12
	Zan	0.19 ± 0.10	0.49 ± 0.26	0.85 ± 0.44	1.67 ± 1.08
	Kara	0.20 ± 0.10	0.49 ± 0.25	0.85 ± 0.44	1.67 ± 1.07

TABLE I
ADE/FDE IN THE ETH DATASET WITH DIFFERENT PREDICTION HORIZONS

	Prediction horizon				
	Methods	1.6 s	3.2 s	4.8 s	8 s
ADE	CVM	0.15 ± 0.09	0.38 ± 0.24	0.71 ± 0.45	1.51 ± 0.91
	LIN	0.29 ± 0.18	0.60 ± 0.38	0.99 ± 0.63	1.84 ± 1.08
	Sof	0.18 ± 0.10	0.36 ± 0.20	0.60 ± 0.35	1.13 ± 0.67
	Zan	0.15 ± 0.09	0.34 ± 0.20	0.59 ± 0.36	1.16 ± 0.70
	Kara	0.16 ± 0.08	0.35 ± 0.19	0.60 ± 0.36	1.16 ± 0.69
FDE	CVM	0.28 ± 0.18	0.86 ± 0.54	1.64 ± 1.05	3.54 ± 2.11
	LIN	0.49 ± 0.31	1.20 ± 0.75	2.07 ± 1.30	3.97 ± 2.27
	Sof	0.29 ± 0.16	0.72 ± 0.42	1.27 ± 0.79	2.48 ± 1.54
	Zan	0.26 ± 0.16	0.72 ± 0.43	1.31 ± 0.82	2.62 ± 1.61
	Kara	0.28 ± 0.15	0.73 ± 0.42	1.31 ± 0.82	2.59 ± 1.59

TABLE II
ADE IN THE THÖR DATASET ("ONE OBSTACLE" SCENARIO) WITH DIFFERENT PREDICTION HORIZONS

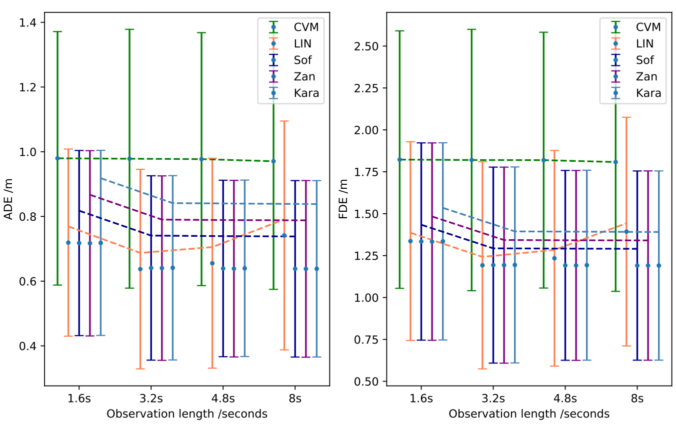

Fig. 4. ADE/FDE in the ATC dataset with different observation lengths using Gaussian filter as initial velocity filter and without smoothing

plots and tables, and the model by Karamouzas et al. [27], abbreviated as *Kara*. As a baseline, we add the linear velocity model (*Lin*), implemented as average velocity in the observed track, and constant velocity model (*CVM*), implemented as forward propagating the last observed motion state.

A. Results and Discussion

Tables I and II show the results of experimenting with different prediction horizons. In general, and not surprisingly, the social force models outperform the linear velocity variants with lower displacement errors, and show more stable performance with lower standard deviations. However, we did not find a substantial difference between Sof, Kara and Zan in any of the datasets and on any of the prediction horizons.

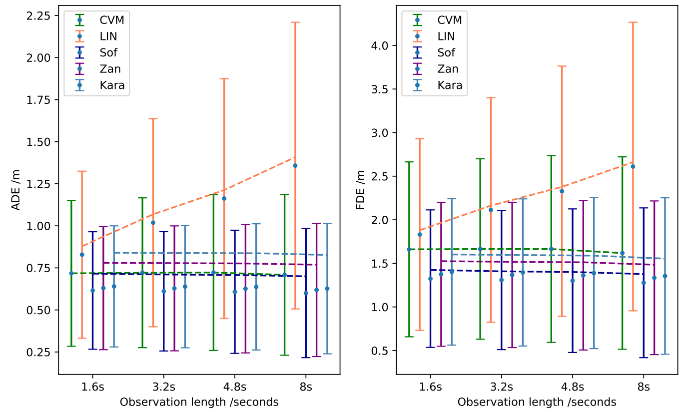

Fig. 5. ADE/FDE in the THÖR dataset ("Three obstacles" scenario) with different observation lengths

	Test				
	Dataset	ETH	HOTEL	ATC	THÖR1
Calibrate	ETH	CVM: 0.40 ± 0.23	0.20 ± 0.16	0.50 ± 0.12	0.83 ± 0.43
		LIN: 0.55 ± 0.35	0.25 ± 0.22	0.56 ± 0.16	1.14 ± 0.59
		Sof: 0.39 ± 0.20	0.21 ± 0.16	0.50 ± 0.12	0.74 ± 0.37
		Zan: 0.38 ± 0.19	0.20 ± 0.16	0.50 ± 0.12	0.71 ± 0.37
		Kara: 0.38 ± 0.19	0.21 ± 0.16	0.50 ± 0.12	0.72 ± 0.38
	HOTEL	Sof: 0.40 ± 0.23	0.20 ± 0.16	0.50 ± 0.12	0.83 ± 0.43
		Zan: 0.40 ± 0.23	0.20 ± 0.16	0.50 ± 0.12	0.82 ± 0.43
		Kara: 0.40 ± 0.23	0.20 ± 0.16	0.50 ± 0.12	0.83 ± 0.43
	ATC	Sof: 0.40 ± 0.23	0.20 ± 0.16	0.50 ± 0.12	0.82 ± 0.43
		Zan: 0.40 ± 0.23	0.20 ± 0.16	0.50 ± 0.12	0.82 ± 0.43
		Kara: 0.40 ± 0.23	0.20 ± 0.16	0.50 ± 0.12	0.82 ± 0.43
	THÖR1	Sof: 0.39 ± 0.20	0.21 ± 0.16	0.50 ± 0.12	0.71 ± 0.36
		Zan: 0.38 ± 0.20	0.20 ± 0.16	0.50 ± 0.12	0.71 ± 0.37
		Kara: 0.38 ± 0.20	0.21 ± 0.16	0.50 ± 0.12	0.71 ± 0.37

TABLE III
ADE MEASURED IN THE TRANSFER EXPERIMENTS ON DIFFERENT DATASETS. THÖR1 ABBREVIATES THE "ONE OBSTACLE" SCENARIO.

Similarly, in experiments with different observation horizons we found no difference between the models. Interestingly, if the observations have low levels of noise, observing additional frames does not improve the performance, see a comparison between the noisy ATC dataset and noise-free THÖR in Fig. 4 and 5 respectively.

Table III summarizes the transfer experiment, where the methods are calibrated on one dataset and tested on another. Also in this case we did not find that one of the three social force models exhibits superior transferability.

Finally, in Fig. 6 and 7 we show the robustness experiment, where we measure performance in presence of noise. While all social force models have excellent robustness, on the level of the very simple and therefore very robust constant velocity model, the predictive variants do not outperform here either.

V. CONCLUSION

In this paper we present the *Atlas* motion prediction benchmark for thorough evaluation in automated repeatable experiments with a systematic variation of the several key prediction parameters. In the future work we plan to release Atlas implementation in Python, implement additional metrics, further baseline algorithms, new scripted experiments and an interface to fine-tune the pattern-based methods in the transfer experiments.

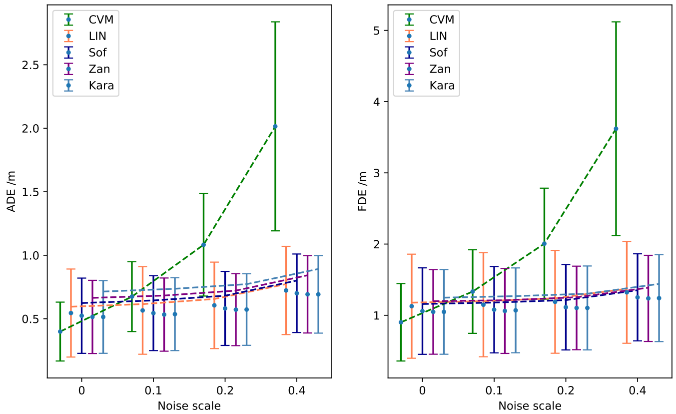

Fig. 6. ADE/FDE in the ETH dataset with added noise and using linear velocity filter

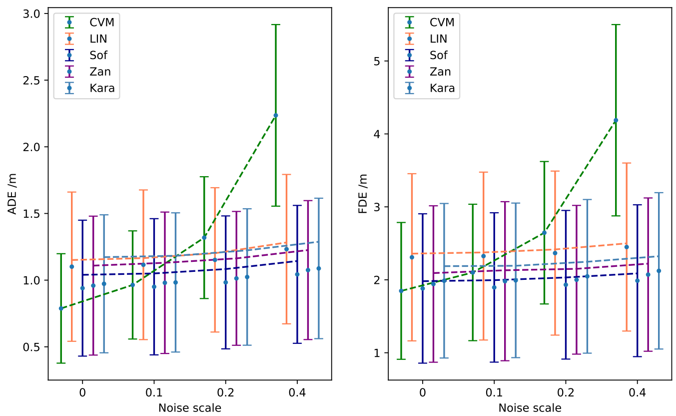

Fig. 7. ADE/FDE in the THÖR dataset ("Three obstacles" scenario) with added noise and using linear velocity filter

REFERENCES

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *Proc. of the IEEE Conf. on Comp. Vis. and Pat. Rec. (CVPR)*, pages 961–971, 2016.

[2] Christoph Schöller, Vincent Aravantinos, Florian Lay, and Alois Knoll. The simpler the better: Constant velocity for pedestrian motion prediction. *arXiv:1903.07933*, 2019.

[3] Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human trajectory forecasting in crowds: A deep learning perspective. *arXiv:2007.03639*, 2020.

[4] Amir Sadeghian, Vineet Kosaraju, Agrim Gupta, Silvio Savarese, and Alexandre Alahi. TrajNet: Towards a benchmark for human trajectory prediction. *arXiv preprint*, 2018.

[5] Marius Lindauer, Katharina Eggensperger, Matthias Feurer, Stefan Falkner, André Biedenkapp, and Frank Hutter. Smac v3: Algorithm configuration in python. https://github.com/automl/SMAC3, 2017.

[6] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, pages 261–268, 2009.

[7] A. Lerner, Y. Chrysanthou, and D. Lischinski. Crowds by example. In *Computer Graphics Forum*, volume 26, pages 655–664. Wiley Online Library, 2007.

[8] B. Majecka. Statistical models of pedestrian behaviour in the forum. *Master's thesis, School of Informatics, University of Edinburgh*, 2009.

[9] D. Brščić, T. Kanda, T. Ikeda, and T. Miyashita. Person tracking in large public spaces using 3-d range sensors. *IEEE Trans. on Human-Machine Systems*, 43(6):522–534, 2013.

[10] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *Proc. of the Europ. Conf. on Comp. Vision (ECCV)*, pages 549–565. Springer, 2016.

[11] A. Rudenko, T. P. Kucner, C. S. Swaminathan, R. T Chadalavada, K. O. Arras, and A. J. Lilienthal. THÖR: Human-robot navigation data collection and accurate motion trajectories dataset. *IEEE Robotics and Automation Letters*, 5(2):676–682, 2020.

[12] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras. Human motion trajectory prediction: A survey. *Int. J. of Robotics Research*, 39(8):895–935, 2020.

[13] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, and Silvio Savarese. SoPhie: An attentive GAN for predicting paths compliant to social and physical constraints. In *Proc. of the IEEE Conf. on Comp. Vis. and Pat. Rec. (CVPR)*, pages 1349–1358, 2019.

[14] Yingfan Huang, HuiKun Bi, Zhaoxin Li, Tianlu Mao, and Zhaoqi Wang. STGAT: Modeling spatial-temporal interactions for human trajectory prediction. In *Proc. of the IEEE Int. Conf. on Computer Vision (ICCV)*, pages 6272–6281, 2019.

[15] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, S Hamid Rezatofighi, and Silvio Savarese. Social-BiGAT: Multimodal trajectory forecasting using Bicycle-GAN and graph attention networks. *arXiv:1907.03395*, 2019.

[16] Nishant Nikhil and Brendan Tran Morris. Convolutional neural network for trajectory prediction. In *Proc. of the Europ. Conf. on Comp. Vision (ECCV)*, pages 0–0, 2018.

[17] Manh Huynh and Gita Alaghband. Trajectory prediction by coupling scene-LSTM with human movement LSTM. In *Int. Symposium on Visual Computing*, pages 244–259. Springer, 2019.

[18] Hao Xue, Du Q Huynh, and Mark Reynolds. SS-LSTM: a hierarchical LSTM model for pedestrian trajectory prediction. In *Proc. of the IEEE Winter Conf. on Applications of Computer Vision (WACV)*, pages 1186–1194. IEEE, 2018.

[19] Tianyang Zhao, Yifei Xu, Mathew Monfort, Wongun Choi, Chris Baker, Yibiao Zhao, Yizhou Wang, and Ying Nian Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *Proc. of the IEEE Conf. on Comp. Vis. and Pat. Rec. (CVPR)*, pages 12126–12134, 2019.

[20] Pu Zhang, Wanli Ouyang, Pengfei Zhang, Jianru Xue, and Nanning Zheng. SR-LSTM: State refinement for LSTM towards pedestrian trajectory prediction. In *Proc. of the IEEE Conf. on Comp. Vis. and Pat. Rec. (CVPR)*, pages 12085–12094, 2019.

[21] Javad Amirian, Jean-Bernard Hayet, and Julien Pettré. Social ways: Learning multi-modal distributions of pedestrian trajectories with GANs. In *Proc. of the IEEE Conf. on Comp. Vis. and Pat. Rec. (CVPR) Workshops*, pages 0–0, 2019.

[22] F. Bartoli, G. Lisanti, L. Ballan, and A. D. Bimbo. Context-aware trajectory prediction. In *Proc. of the IEEE Int. Conf. on Pattern Recognition*, pages 1941–1946, 2018.

[23] Tharindu Fernando, Simon Denman, Sridha Sridharan, and Clinton Fookes. Soft+Hardwired attention: An LSTM framework for human trajectory prediction and abnormal event detection. *Neural networks*, 108:466–478, 2018.

[24] Chaofan Tao, Qinhong Jiang, Lixin Duan, and Ping Luo. Dynamic and static context-aware lstm for multi-agent motion prediction. *arXiv:2008.00777*, 2020.

[25] D. Helbing and P. Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.

[26] F. Zanlungo, T. Ikeda, and T. Kanda. Social force model with explicit collision prediction. *EPL (Europhysics Letters)*, 93(6):68005, 2011.

[27] I. Karamouzas, P. Heil, P. van Beek, and M. H. Overmars. A predictive collision avoidance model for pedestrian simulation. In *Int. Workshop on Motion in Games*, pages 41–52. Springer, 2009.

[28] S. Kim, S. J. Guy, W. Liu, D. Wilkie, R. W. H. Lau, M. C. Lin, and D. Manocha. BRVO: Predicting pedestrian trajectories using velocity-space reasoning. *Int. J. of Robotics Research*, 34(2):201–217, 2015.

[29] F. Farina, D. Fontanelli, A. Garulli, A. Giannitrapani, and D. Prattichizzo. Walking ahead: The headed social force model. *PloS one*, 12(1):e0169734, 2017.

[30] A. Rudenko, L. Palmieri, and K. O. Arras. Joint prediction of human motion using a planning-based social force approach. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 1–7, 2018.