HSFM-ΣNN: Combining a Feedforward Motion Prediction Network and Covariance Prediction

Aleksey Postnikov[1,2] Aleksander Gamayunov[1,2] Gonzalo Ferrer[2]

***Abstract*—In this paper, we propose a new method for motion prediction: HSFM-ΣNN. Our proposed method combines two different approaches: a feedforward network whose layers are model-based transition functions using the HSFM and a Neural Network (NN), on each of these layers, for covariance prediction. We will compare our method with classical methods for covariance estimation showing their limitations. We will also compare with a learning-based approach, social-LSTM, showing that our method is more precise and efficient. We will evaluate our results using the [10], [11] datasets.**

I. Introduction

High accurate prediction of human trajectories in urban environments is a topic that has been actively investigated during the last years and it has a deep impact in related topics, such as, decision making, path planning, surveillance, tracking, etc. The problem of forecasting where pedestrians will be in the near future is, however, ill-posed by nature: Human beings tend to be unpredictable on their decisions and motion is neither exempt of it.

Most modern motion prediction algorithms focus on accurate prediction of agent position errors. Nonetheless, the precision due to this inherent uncertainty is equally important, and this paper is an effort to research on this direction.

Motion prediction algorithms has been classically divided into model and learning-based. A more relevant classification to our paper, is based on the representation of the output:

First-order moments: usually mean is predicted, which is a single vector of state variables. On this category we would include most of the methods. The Social Force Model (SFM) [12] and its Headed variant (HSFM) [1] and [2], Prediction for decision making [8], learning-based approaches with deep neural networks [3], also learning based inverse reinforcement learning models [15], [16] .

Second-order moments: assuming a Gaussian distribution, only two moments are required to completely specify a distribution. Many current Deep Learning (DL) approaches belong to this category, such as Social-LSTM [5] and other DL methods [7], [15].

Non-parametric Any distribution of the prediction variables is possible, for instance an occupancy 2D grid [13], [14].

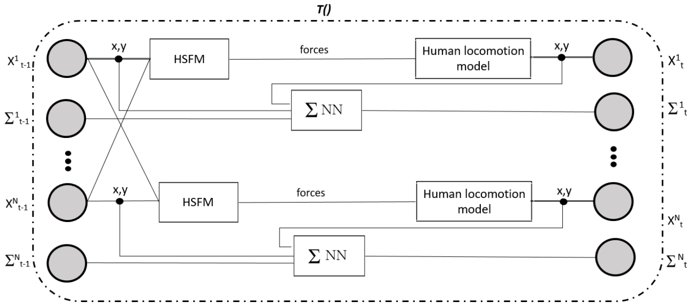

Fig. 1: Diagram of the proposed method, HSFM-ΣNN. At each time-step, the HSFM generates virtual forces, which are then integrated. Our approach combines this with a NN for covariance prediction.

In this paper, we propose a motion prediction network, which propagates the system states variables, i.e. each of the pedestrians positions, over several iterations up to a time horizon. To achieve that, we combine on *each* transition function (or network layer): 1) a model-based prediction (HSFM) and 2) a NN to precisely predict covariance, getting the best of both approaches: efficiency and simplicity from model-based and precision from NN.

II. Motion Prediction Network

In our work, we use the transition function $T()$, shown at Fig. 1, which modifies the state variables of a pedestrian 2D pose at timestamp t to $t+1$, in the following way:

$$x_{t+1} = T(x_t). \tag{1}$$

A motion prediction network is defined as a number of consecutively stacked transition layers $T()$, similar to the feedforward network proposed in [8], using SFM modules. The contribution in this work is the addition of a shallow neural network at *each* transition block in order to predict covariances (Sec. III-C).

III. Uncertainty estimation

A. Linearization and Covariance Forward-Propagation (FP)

The transition function (1) is a non-linear differentiable function (by construction). The simplest method for covariance estimation is using the first-order Taylor expansion:

$$x_{t+1} = T(\mu_t) + G_t(x_t - \mu_t), \tag{2}$$

where μ_t is the current state estimate and G_t is the Jacobian of $T()$. From here, we apply *Covariance Propagation* of a Gaussian random variable ($x_t \sim \mathcal{N}(\mu_t, \Sigma_t)$) over a linear function:

$$x_{t+1} \sim \mathcal{N}\big(T(\mu_t), G_t \cdot \Sigma_t \cdot G_t^\top\big). \tag{3}$$

[1] The authors are with the Sberbank Robotics Laboratory, Moscow, Russia. {postnikov.a.l,gamayunov.a.r}@sberbank.ru.

[2] Skolkovo Institute of Science and Technology, Moscow, Russia. g.ferrer@skoltech.ru.

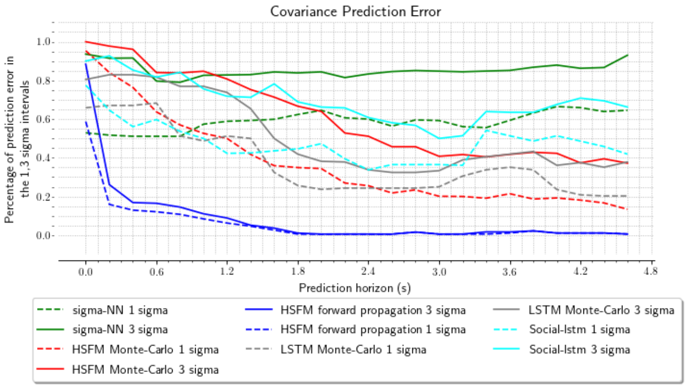

Fig. 2: Evaluation of calculated covariances for Monte-Carlo, forward propagation methods based on HSFM and social-LSTM transition functions.

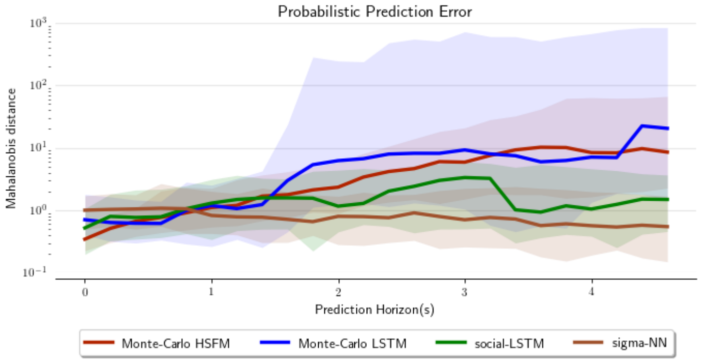

Fig. 3: Mahalanobis error distances for social-LSTM and HSFM transition functions. In solid lines are drawn median, and colored intervals are .25 and .75 percentiles.

B. Monte-Carlo Covariance Estimation

The Monte-Carlo (MC) approach is a commonly used and powerful technique to quantify uncertainty.

The procedure is straightforward: we sample from an initial distribution $x_t^i \sim p(x_t), \quad i = 1, \ldots, N$, propagate each sample $x_{t+1}^i = T(x_t^i)$ and calculate the statistics of this new set, in particular, we calculate sample mean and sample covariance.

C. Neural Network Covariance Prediction

In order to predict covariances, a Neural Network (NN) is trained separately. We assume that ground truth covariances are available (see Sec.IV). The proposed architecture is a fully connected NN, consisting of 2 hidden layers with ReLU activation function. The inputs are the stacked vectors x_t, Σ_{x_t} and x_{pred}, as seen in Fig. 1. Hidden layers input features dimensions are 100 and 50, respectively, and the final layer outputs 2 variables: $\sigma^2_{x_{t+1}}, \sigma^2_{y_{t+1}}$.

IV. Evaluation

In this section, we present experiments on two publicly available human-trajectory datasets: ETH [10] and UCY [11]. The ground truth covariance is unknown and for training purposes we approximate it as follows(which might be subject for future improvements):

$$\bar{\Sigma}_H = ||\tilde{x}_1 + H \cdot v_1 - \tilde{x}_H||_2^2 \cdot I_{2\times 2}$$

where v_1 is the linear velocity at initial time, H is the horizon time, $\tilde{x}_1$ and $\tilde{x}_H$ are obtained from the dataset(DS). This quantity is a measure on how much the future position deviates from a constant linear propagation during H. Covariances are then calculated for a range of prediction horizons up to $4.8s$ and $\Delta t = 0.2s$.

Figure 2 shows the results for the covariance prediction for each of the methods described above. The graphic shows a percentage of number of times that the predicted error $||x_H - \tilde{x}_H||^2_{\Sigma_H}$, considering the estimated covariance, lies inside the $1, 3\sigma$ intervals.

Then, we check the consistency of the covariance prediction by comparing with the theoretical results on 2D Gaussian variables: we should observe around 64% of the predicted poses values being within one standard deviation interval (1σ), and 98% within 3σ.

Method	percent of predicted values inside 1σ (Δ from expected)	percent of predicted values inside 3σ(Δ from expected)
LSTM	47.60 (-16.39)	69.74 (-28.25)
LSTM MC	37.16 (-26.83)	51.33 (-46.66)
HSFM MC	37.11 (-26.88)	60.98 (-37.01)
HSFM FP	6.06 (-57.93)	8.60 (-89.39)
HSFM-ΣNN	**58.79 (-5.20)**	**85.45(-12.54)**

TABLE I: Comparison of calculated covariances

The forward propagation (FP) method collapses and provides poor results (Fig. 2) due to vanishing gradients over multiple FPs. This is a valuable negative result we report in this paper. Stacking several layers on a prediction network makes the FP approach unusable for covariance estimation.

The MC approach is neither providing good results: for short time horizons the predicted covariance is consistent, however for larger horizons, we observe a degradation on both 1 and 3-σ, clearly underestimating the true covariance. The same result is obtained for social-LSTM. On the other hand, our proposed method, HSFM-ΣNN achieves consistent results for any time horizon, both on 1 and 3-σ intervals, which support the initial hypothesis of assuming Gaussian rvs and it justifies the ground truth covariance approximation.

In Fig. 3 is depicted the Mahalanobis distance of the predicted error. In this case, we observe how both social-LSTM and our method (HSFM-ΣNN) perform well and the probabilistically weighted error norm is preserved. An unexpected drop in Mahalanobis error and covariance prediction error after 3s of forecasting for Social LSTM caused by an increase in predicted covariance. MC increases the error with the time horizon.

V. Conclusions

We have proposed a method, HSFM-ΣNN, for trajectory prediction based on a motion prediction network and we have added a covariance prediction NN for each of the transition modules used. We have evaluated that the most precise estimation of covariances is by NN prediction: Linear covariance propagation collapses by vanishing gradients, MC estimation does not capture the error correctly and other learning approaches, such as social-LSTM, are accurate in MH distances but become overconfident on their covariance prediction over longer horizons.

References

[1] Farina, F., Fontanelli, D., Garulli, A., Giannitrapani, A., Prattichizzo, D. (2016, December). *When Helbing meets Laumond: the headed social force model.* In 2016 IEEE 55th Conference on Decision and Control (CDC) (pp. 3548-3553). IEEE.

[2] Mombaur, K., Truong, A., Laumond, J. P. (2010). From human to humanoid locomotion—an inverse optimal control approach. Autonomous robots, 28(3), 369-383.

[3] Yi, S., Li, H., Wang, X. (2016, October). Pedestrian behavior understanding and prediction with deep neural networks. In European Conference on Computer Vision (pp. 263-279). Springer, Cham.

[4] Ellis, D., Sommerlade, E., Reid, I. (2009, September). Modelling pedestrian trajectory patterns with gaussian processes. In 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops (pp. 1229-1234). IEEE.

[5] 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).

[6] Zhang, P., Ouyang, W., Zhang, P., Xue, J., Zheng, N. (2019). Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 12085-12094).

[7] Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A. (2018). Social gan: Socially acceptable trajectories with generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2255-2264).

[8] Mehta, D., Ferrer, G., Olson, E. (2018, May). Backprop-MPDM: Faster risk-aware policy evaluation through efficient gradient optimization. In 2018 IEEE International Conference on Robotics and Automation (ICRA) (pp. 1740-1746). IEEE.

[9] Abdelaziz, A. H., Watanabe, S., Hershey, J. R., Vincent, E., Kolossa, D. (2015, September). Uncertainty propagation through deep neural networks.

[10] Pellegrini, Stefano, et al. ”You'll never walk alone: Modeling social behavior for multi-target tracking.” 2009 IEEE 12th International Conference on Computer Vision. IEEE, 2009.

[11] Lerner, Alon, Yiorgos Chrysanthou, and Dani Lischinski. ”Crowds by example.” Computer graphics forum. Vol. 26. No. 3. Oxford, UK: Blackwell Publishing Ltd, 2007.

[12] Helbing, Dirk, and Peter Molnar. ”Social force model for pedestrian dynamics.” Physical review E 51.5 (1995): 4282.

[13] Sarmady, Siamak, Fazilah Haron, and Abdullah Zawawi Talib. ”Simulation of pedestrian movements using fine grid cellular automata model.” arXiv preprint arXiv:1406.3567 (2014).

[14] Rehder, Eike, and Horst Kloeden. ”Goal-directed pedestrian prediction.” Proceedings of the IEEE International Conference on Computer Vision Workshops. 2015.

[15] Zhang, Yanfu, et al. ”Integrating kinematics and environment context into deep inverse reinforcement learning for predicting off-road vehicle trajectories.” arXiv preprint arXiv:1810.07225 (2018).

[16] Fernando, Tharindu, et al. ”Neighbourhood context embeddings in deep inverse reinforcement learning for predicting pedestrian motion over long time horizons.” Proceedings of the IEEE International Conference on Computer Vision Workshops. 2019.