Dynamic and Static Context-aware LSTM for Multi-agent Motion Prediction

Chaofan Tao[1,2][0000−0002−6093−0854], Qinhong Jiang[3][0000−0002−5509−7247], Lixin Duan[2][0000−0002−0723−4016], and Ping Luo[1][0000−0002−6685−7950]

[1] The University of Hong Kong, Hong Kong, China
[2] University of Electronic Science and Technology of China, China
{tcftrees,lxduan}@gmail.com
{pluo}@cs.hku.hk
[3] SenseTime, China
{jiangqinhong}@sensetime.com

Abstract. Multi-agent motion prediction is challenging because it aims to foresee the future trajectories of multiple agents (*e.g.* pedestrians) simultaneously in a complicated scene. Existing work addressed this challenge by either learning social spatial interactions represented by the positions of a group of pedestrians, while ignoring their temporal coherence (*i.e.* dependencies between different long trajectories), or by understanding the complicated scene layout (*e.g.* scene segmentation) to ensure safe navigation. However, unlike previous work that isolated the spatial interaction, temporal coherence, and scene layout, this paper designs a new mechanism, *i.e.*, Dynamic and Static Context-aware Motion Predictor (DSCMP), to integrates these rich information into the long-short-term-memory (LSTM). It has three appealing benefits. (1) DSCMP models the dynamic interactions between agents by learning both their spatial positions and temporal coherence, as well as understanding the contextual scene layout. (2) Different from previous LSTM models that predict motions by propagating hidden features frame by frame, limiting the capacity to learn correlations between long trajectories, we carefully design a differentiable queue mechanism in DSCMP, which is able to explicitly memorize and learn the correlations between long trajectories. (3) DSCMP captures the context of scene by inferring latent variable, which enables multimodal predictions with meaningful semantic scene layout. Extensive experiments show that DSCMP outperforms state-of-the-art methods by large margins, such as 9.05% and 7.62% relative improvements on the ETH-UCY and SDD datasets respectively.

Keywords: Motion prediction, Trajectory Forecasting, Social model

1 Introduction

Multi-agent motion prediction is an important task for many real-world applications such as self-driving vehicle, traffic surveillance, and autonomous mobile robot. However, it is challenging because it aims at foreseeing the future trajectories of multiple agents such as pedestrians simultaneously in a complicated scene. Existing work $[1, 5, 29, 2, 21, 3, 35, 15]$ that addressed this challenge can be generally partitioned into two categories. In the first category $[1, 5, 29, 2]$, previous work predicted the motions by learning social spatial interactions, which are represented by the positions of pedestrians. However, these approaches typically ignored the the dependency between different long trajectories of pedestrians. In the second category $[21, 35, 15, 3]$, prior arts combined scene understanding to regularize the predicted trajectory, such as visual feature of the complicated scene layout.

Different from existing work that either model agents' interactions or the scene layout, we carefully designed novel mechanisms in LSTM to model dynamic interactions of pedestrians in both spatial and temporal dimensions, as well as modeling the semantic scene layout as latent probabilistic variable to constrain the predictions. These design principles enable our model to predict multiple trajectories for each agent that cohere in time and space with the other agents. We see that the proposed method outperforms its counterparts in many benchmarks as shown in Fig.1(c).

We name the proposed method as Dynamic and Static Context-aware Motion Predictor (DSCMP), which has an encoder-decoder structure of LSTM that has carefully devised mechanisms to tackle multi-agent motion prediction. DSCMP has three appealing benefits that previous work did not have.

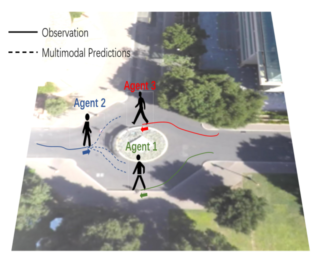
For the first benefit, unlike existing methods $[1, 5, 29, 21, 35]$ that employed recurrent neural networks (RNNs) to learn motion by passing message frame by frame, DSCMP incorporates a queue mechanism in LSTM to explicitly propagate hidden features of multiple frames, enabling to capture long trajectories among pedestrians more explicitly than prior arts.

Specifically, the vanilla LSTM in previous approaches $[4, 22]$ attempts to learn a frame-by-frame predictor for each agent i, denoted as $m^i_{t+1} = p(m^i_t, h^i_{t-1})$, where $p(\cdot)$ is a prediction function of LSTM, m^i_t represents the current motion state (*i.e.* the x, y positions) at the t-th frame and h^i_{t-1} is the hidden feature of the previous single frame. The frame-wise models hinders their capacity to capture the dependency between long trajectories of pedestrians.

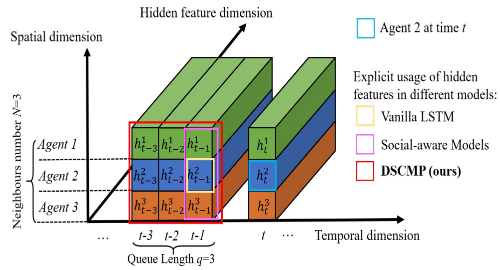
The recent approaches of social-aware LSTM models $[1, 5, 27]$ modified the above vanilla LSTM by using $m^i_{t+1} = p(m^i_t, \bigcup_i^{N(i)} h^i_{t-1})$, where $\bigcup$ denotes combination of a set of hidden features of the spatial neighbors of the i-th pedestrian (denoted by $N(i)$) at the previous $(t-1)$-th frame.

However, the above methods are insufficient to consider the interactions across agents. For example, as shown in Fig.1(a), the agent 2 is heading towards agent 1 and agent 3. To avoid collision, the agent 2 tends to adjust his future trajectory by anticipating the intention of agent 1 and 3 based on their recent movement history, rather than their states at the previous one frame only.

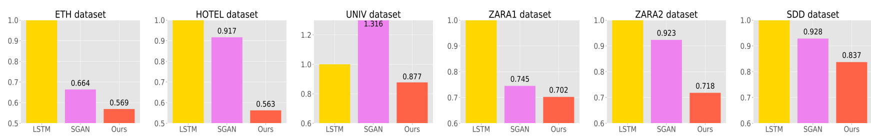
Different from the above existing approaches, a LSTM is carefully designed in DSCMP to learn both spatial dependencies and temporal coherence of moving pedestrians. The LSTM contains two modules, including an Individual Context Module (ICM) and a Social-aware Context Module (SCM). As shown in Fig.1(b), our model fully understands spatial and temporal contexts across agents by



(a) Illustration of multi-agent motion prediction.



(b) Comparisons of different approaches.



(c) Comparisons of the Relative Average Distance Error (ADE) on various datasets.

Fig. 1. (a) illustrates motion prediction in a real-world scenario, where both the dynamic motion context across agents and static context are involved. **(b)** compares various methods including LSTM, social-aware models (*e.g.* SGAN [5]) and DSCMP. The queue mechanism in DSCMP enriches the scope of dynamic context at each frame, enabling DSCMP to effectively learn long motions. **(c)** compares the Average Distance Error (ADE) of LSTM [6], SGAN [5] and DSCMP on ETH (two subsets: ETH, HOTEL) [17], UCY (three subsets: UNIV, ZARA1, ZARA2) [13], SDD [19] by using the performance of LSTM as baseline (*i.e.* unit 1; lower value is better). We see that DSCMP surpasses its counterparts by large margins. Best viewed in color.

learning a predictor denoted as $m^i_{t+1} = p(m^i_t, \bigcup_i^{N(i)} Q^i_t)$, where Q^i_t denotes a set of features not only across agents at a certain frame, but also across multiple successive frames of different agents.

More specific, the ICM of DSCMP passes feature of the current motion state and the corresponding feature queue into LSTM cell. Multiple forget gates control the information flow of the frames in the queue. At each iteration, we update the queues by appending the features of the latest frame, and popping out the earliest features. Furthermore, the SCM of DSCMP refines the updated queues by using the queues of the neighboring agents. Since these queues preserve agent-specific motion cues in the past multiple frames, we are able to learn a long-range spatial-temporal interactions with the aggregation of queues.

For the second benefit, we observe that the future movements of agents in real scenarios have uncertainty, since multiple trajectories are plausible. For instance, an agent would naturally consider his/her surrounding scene layout when deciding his/her possible future paths. In particular, an agent could turn left or right in a crossroad, while he/she has limited choices around a street corner. However, the recent methods either neglected the guidance of scene layout to produce diverse predictions for each agent, or even totally ignored the scene information. In contrast, DSCMP incorporates the scene information into the learning of diverse predictions by using $m_{t+1}^i = p(m_t^i, \bigcup_i^{N(i)} Q_t^i, I)$, where I indicates the semantic scene feature after scene segmentation. In practice, this semantic scene feature is modeled as a latent variable of a probabilistic distribution to predict multiple future trajectories for each agent.

For the third benefit, in order to understand the uniqueness of DSCMP, we propose a new evaluation metric called Temporal Correlation Coefficient (TCC) to fully evaluate the temporal correlation of motion patterns, bridging the gap where the commonly used metric such as Average Distance Error (ADE) and Final Distance Error (FDE) are insufficient to evaluate temporal motion correlations. Extensive experiments on dataset ETH [17]-UCY [13], SDD [19] show that DSCMP surpasses state-of-the-art methods on all the above metrics by large margins, such as 9.05% and 7.62% relative improvements on metric ADE compared to the latest method STGAT [7] method.

To summarize the above benefits, this work has three main **contributions**. **(1)** We present a novel future motion predictor, named DSCMP, which is able to explicitly model both the spatial and temporal interactions between different agents, as well as producing multiple probabilistic predictions of future paths for each agent. **(2)** We carefully design the LSTM modules in DSCMP to achieve the above purposes, where all modules can be trained end-to-end including the Individual Context Module (ICM), the Social-aware Context Module (SCM), and a latent scene information module. **(3)** Extensive experiments on the ETH [17], UCY [13], and SDD [19] datasets demonstrate that DSCMP outperforms its counterparts by large margin in multiple evaluation metrics such as ADE and FDE, as well as a new metric, Temporal Correlation Coefficient (TCC), proposed by us to better examine the temporal correlation of motion prediction.

2 Related Work

Motion Prediction Early approaches for motion prediction [20] like physics-based methods[17, 30, 18] and planning-based methods [31, 25, 12, 9] are usually limited by hand-crafted kinematic equations and reward function respectively. With the development of recurrent neural networks, pattern-based methods [1, 5, 27, 11, 33, 11, 8, 29, 16, 10] have been studied recently. While most of the models consider the agents in world coordinates, some work [16, 10] explore trajectory prediction with egocentric vision.

A pioneering pattern-based work that combine the LSTM and social interactions is introduced in [1]. The authors of [5] proposes an adversarial framework to sequentially generate predictions. A social pooling is employed to learn the spatial dependencies among agents. Spatio-temporal graphs [27, 14, 8] are adopted to model the relations on a complete graph, whereas these methods suffer from implicit modelling of dynamic edges or poor scalability with $O(N^2)$ complexity. STGAT [7] is the most relevant method to our work. It considers the temporal correlations of motion in multiple frames. Unlike STGAT deduces a single correlation representation from the whole observation process, we explicitly keep track of the temporal correlation for each iteration during observation. We also take scene context into consideration.

Contextual Understanding Humans are capable of inferring and reasoning by understanding the context. Rich contextual information is proven to be valuable in sequential data modeling (e.g. video, language, speech). Attention mechanism [26, 32] has shown great success in concentrating on the significant parts of visual or textual inputs at each time steps. Non-local operation [28] works as a generic block to capture the dependencies directly by computing the pair-wise relations in the long-range context. In the field of motion prediction, graph attention [7, 11] assigns different importance to the neighboring pedestrians to involve the social-aware interactions. The authors of [35, 21] encode visual features of scene context in the LSTM to predict physics-feasible trajectories.

Multimodal Predictions Multi-modality is an important characteristic of motion prediction that implies the multiple plausible choices of future trajectories. To model this uncertainty, the model is required to generate diverse predictions. A common approach [5, 35, 21] is to fuse latent variables sampled from a predefined Gaussian distribution $\mathcal{N}(0,1)$ with hidden feature. However, predefined latent variables suffer from the absense of context reasoning. In this paper, the latent variable is learnable from the scene context, which enables our model to generate multimodal predictions with meaningful semantics.

3 Method

Overview. To be specific, during the observation from frame t_1 to t_{obs}, the motion states of all N agents $M_{t_1:t_{obs}} = \{M_t | t \in [t_1, \cdots, t_{obs}]\}$ in a scene and scene information I are given, where $M_t \in R^{N \times d}$. Symbol d is the dimension of input motion state. It refers to the x-y coordinates of agent's location in this paper, thereby $d = 2$. $M_t = \left\{M_t^i | i \in N\right\}$, where $M_t^i = (x_t^i, y_t^t) \in R^2$ denotes the location of agent i at frame t . Our goal is to predict the locations of all the agents $M_{t_{obs+1}:t_{obs+pred}}$ in the future frames. The workflow of our framework DSCMP is illustrated in Fig.2. For each iteration during observation, we send the current motion states with proposed queues to the encoder, and then we update the queues via ICM and then refine them via SCM. The last hidden features in the encoder is concatenated with a scene-guided latent variable. Later, the fused feature is passed to a LSTM decoder to obtain predicted motion.

3.1 The Function of Queues

With the setup of queues, the previous motion context and memory context for current frame t are temporarily stored. Specifically, we construct a hidden feature queue $Q_{h_t}^i = \left[h_{t-q}^i, \cdots, h_{t-1}^i\right] \in R^{1*q*h}$ and cell queue $Q_{c_t}^i = \left[c_{t-q}^i, \cdots, c_{t-1}^i\right] \in R^{1*q*h}$ for each agent i The size of LSTM feature is denoted as h. The queue length q describes a time bucket that the features are explicitly propagated. We initialize the hidden feature queues and cell queues with zero for each agent.

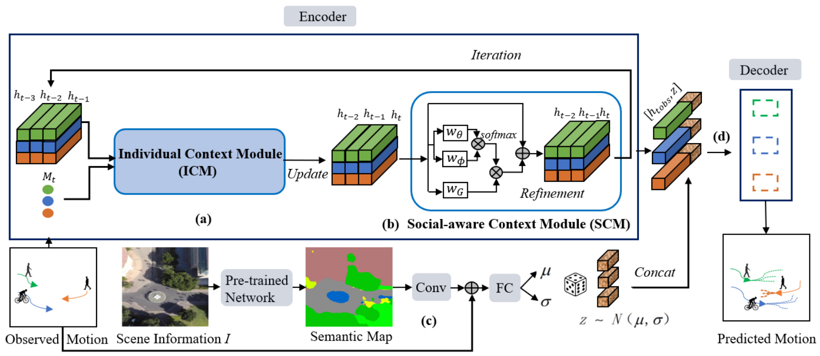


Fig. 2. The overview of our framework (DSCMP). Given a sequence of the observed motions, we construct agent-specific queue to store the LSTM features of previous frames within a queue length. The queue length is set to 3 as example. **(a)** For each iteration during observation, the current motion state and queues are encoded via ICM. **(b)** The queues are updated by appending the latest features and popping the earliest ones. In the SCM, the queues are adaptively refined by considering the pair-wise relations of features in the neighbours' queues. **(c)** The semantic map of static context is incorporated with observed motion to generate a learnable latent variable z. **(d)** We concatenate the last hidden feature in the encoder and the latent variable to predict the motions via LSTM decoder.

3.2 Individual Context Module

Based on the aforementioned queues, we firstly capture the temporal correlations of trajectories from individual level. In order to handle with multiple inputs in one iteration, we employ a tree-like LSTM cell [24]. The historical states in a time bucket $(t-q,\cdots,t-1)$ are viewed as the children of the current state t. After an iteration, the queues are updated by appending features at frame t and popping features at frame $t-q$ We first average the hidden features in the queue to obtain a holistic representation $\tilde{h}_{t-1}^i = \sum_{l=1}^{q} h_{t-l}^i$ from the past frames. As the computational graph of ICM shown in Fig.3(a), the propagation is formulated as follows:

$$\begin{aligned}
g_t^i &= \sigma(W^g M_t^i + U^g \tilde{h}_{t-1}^i + b^g), & f_t^{il} &= \sigma(W^f M_t^i + U^f h_{t-l}^i + b^f),\\
o_t^i &= \sigma(W^o M_t^i + U^o \tilde{h}_{t-1}^i + b^o), & u_t^i &= \tanh(W^u M_t^i + U^u \tilde{h}_{t-1}^i + b^u),\\
c_t^i &= g_t^i \odot u_t^i + \sum_{l=1}^{q} f_t^{il} \odot c_{t-l}^i, & h_t^i &= o_t^i \odot \tanh(c_t^i),
\end{aligned} \tag{1}$$

where σ is sigmoid function and $\odot$ is element-wise multiplication. From the equation 1, we could observe that multiple previous frames pass message to the current cell. The contributions of these frames to the current state are controlled by multiple forget gates $f_t^{il}, l \in [1,q]$. In the case $q=1$, ICM degenerates to a vanilla LSTM cell, which considers the previous single feature only at one iteration.

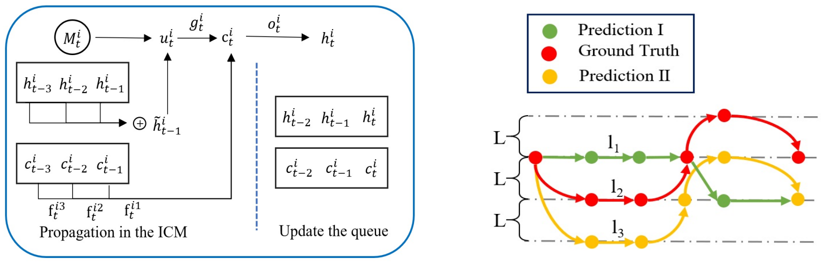


Fig. 3. (a): The computation graph of Individual Context Module (ICM). **(b):** An example to show Euclidean distance-based metric (e.g. ADE and FDE) cannot fully evaluate motion patterns. The prediction I and prediction II score the same ADE (5L/6) and FDE (L), whereas the prediction II captures the temporal correlation of motion pattern much better than the prediction I.

In practice, we assign each agent the queues of the fixed length. We point out that it is inappropriate in some cases. For example, the motion of some agents may be erratic that temporally incoherent with the past states. However, the adaptive forget gates could control the volume of information from the past frames. Hence, irrelevant motions could be filtered during the propagation.

3.3 Social-aware Context Module

Social interactions works as an important part of dynamic context. Since the aggregation of neighbors' queues $\in R^{N(i)*q*h}$ store the surrounding historical information across agents, our model could learn spatio-temporal dependencies in a single operation. Here $N(i)$ denotes the number of neighbors of the agent i (include oneself). In the SCM, we compute the pair-wise relations of elements in the queues from neighbours. The refined queues can be viewed as a weighted sum from the neighbors' queues. Non-local block [28] is chosen for relation inference since it not only captures distant relations but also keeps the shape of input. The refinement of the hidden feature queue $Q_{h_{t+1}}^i = \left[h_{t-q+1}^i, \cdots, h_t^i\right]$ is computed as:

$$h_{t-l}^i = h_{t-l}^i \oplus \frac{1}{\mathcal{Z}_{t-l}^i} \sum_{j=1}^{N(i)} \mathcal{R}(h_{t-l}^i, h_{t-l}^j), \mathcal{G}(h_{t-l}^j),\ l \in \{0, \cdots, q-1\} \tag{2}$$

$$\mathcal{R}(h_{t-l}^i, h_{t-l}^j) = (W_\theta h_{t-l}^i)^{\mathrm{T}}(W_\phi h_{t-l}^j), \quad \mathcal{G}(h_{t-l}^j) = W_{\mathcal{G}} h_{t-l}^j, \tag{3}$$

where $\mathcal{R}(h_{t-l}^i, h_{t-l}^j)$ is a scalar that reflects the relationships between the feature h_{t-l}^i and h_{t-l}^j. And the component $\mathcal{G}(h_{t-l}^j)$ refers to the transformed representation of the neighbor agent j at frame $t-l$. $N(i)$ denotes the neighbors of agent i. $\mathcal{Z}_{t-l}^i = \sum_{j=1}^{N(i)} \mathcal{R}(h_{t-l}^i, h_{t-l}^j)$ is a normalization factor. The parameters of function $\mathcal{R}(\cdot,\cdot)$ and $\mathcal{G}(\cdot)$ are shared among agents. The cell queues stay invariant since we focus on motion interactions rather than memory at this step.

3.4 Semantic Guidance from Scene Context

Scene information is a valuable static context that provides the semantic of layout around the agents. In practice, we extract the semantic maps from resized 256×256 scene images I via pre-trained PSPNet [36, 34] off-line. After that, we send the semantic maps to convolutional layers (Conv) and then combine them with the observed trajectories via a fully-connected (FC) layer. The latent variable z is obtained with reparameterization trick on the mean μ and variance σ as follows:

$$[\mu, \sigma] = \mathrm{FC}(\mathrm{Conv}(I) \oplus \sum_{i}^{N(i)} M^i_{t_1:t_{obs}}), \quad z \sim \mathcal{N}(\mu, \sigma), \tag{4}$$

where $\oplus$ denotes element-wise addition. The latent variable z enables multi-modal predictions by going into the LSTM decoder with the last hidden features h_{obs} during observation. During the forecasting phase, the predicted motions $\hat{M}^i_{t_{obs}:t_{obs+pred}}$ are sequentially generated in the decoder.

3.5 Model Training

In order to encourage the coherence of temporal-neighboring features, we utilize a regularization loss L_c inspired by [23], which is defined as follows:

$$L_c = \begin{cases} 1 - \cos(h^i_{t_1}, h^i_{t_2}), & |t_1 - t_2| < q \\ \max(0,\ \cos(h^i_{t_1}, h^i_{t_2}) - \mathrm{margin}), & otherwise \end{cases} \tag{5}$$

where *cos* is cosine similarity and *margin* is a hyperparameter. The pair-wise features $(h^i_{t_1}, h^i_{t_2})$ are randomly sampled in a batch. L_c maximizes the similarity of features within a queue length (where frames are likely to strongly correlated with each other), while penalizing the similarity of features over a queue length (where frames are likely to belong to different motion patterns). The total loss function combines the regularization term L_c and the variety loss (the second term) followed by [5] as:

$$Loss = \lambda L_c + \min_m \left\| M^i_{t_{obs}:t_{obs+pred}} - \hat{M}^{i(m)}_{t_{obs}:t_{obs+pred}} \right\|_2, \tag{6}$$

where λ is a trade-off parameter. The variety loss computes the $L2$ distance between the best of m predictions and the ground truth, which encourages to cover the space of outputs that conform to the past trajectory.

4 Experiments

4.1 Datasets and Evaluation Metrics

We evaluate our method on three datasets (ETH [17], UCY [13], SDD [19]). ETH contains two subsets named ETH and HOTEL. UCY consists of three subsets, called UNIV, ZARA1 and ZARA2. Totally, there are 1,536 trajectories of pedestrian in the crowd collected in 5 scenes. We observe for 3.2s (8 frames) and the predict the motions the next 4.8s (12 frames) for every pedestrian simultaneously. For the data split and evaluation, we follow the leave-one-out method in [5]. SDD dataset has large volume with complex scenes. It contains 60 bird-eye-view videos with corresponding trajectories that involves multiple kinds of agents (pedestrian, bicyclist, .etc). The observation duration is 3.2s and the prediction duration ranged from 1s to 4s. We divide the dataset into 16,000 video clips and follow a 5-fold cross-validation setup.

Commonly used Euclidean-based metrics like ADE and FDE neglect the temporal correlation of motion patterns. An illustration example is shown in Fig.3(b). In order to supplement this loophole, we introduce a new metric that requires no assumptions about the temporal distribution of trajectories, Temporal Correlation Coefficient (TCC). The TCC is defined as:

$$TCC = \frac{1}{2}(TCC_x + TCC_y). \tag{7}$$

$$TCC_x = \frac{1}{N}\sum_{i}^{N}\frac{Cov(\hat{x^i}, x^i)}{\sqrt{D(\hat{x^i})D(x^i)}}, \qquad TCC_y = \frac{1}{N}\sum_{i}^{N}\frac{Cov(\hat{y^i}, y^i)}{\sqrt{D(\hat{y^i})D(y^i)}}, \tag{8}$$

where the ground truth trajectory for the agent i is $M^i = (x^i, y^i)$. The corresponding predictions are denoted as $\hat{M^i} = (\hat{x^i}, \hat{y^i})$. From the equations above, we can observe that TCC ranges from $[-1, 1]$. A high TCC implies the predictions capture the time-varying motion patterns greatly, whereas a negative TCC denotes a weak capture of temporal correlation.

For the evaluation, the metric ADE (Average Distance Error) denotes the average L2 distance between the predictions and ground truth, and the metric FDE (Final Distance Error) reflects the L2 distance between the predictions and ground truth in the final frame. The TCC (Temporal Correlation Coefficient) is used to evaluation the temporal correlation of motion pattern in predictions.

4.2 Implementation Details

We preprocess the input motion state as the relative position. The size of hidden feature and the dimension of latent variable are set as 32 and 16 respectively. The convolutional part for scene is three-layer with kernel size as 10, 10, 1. The subsequent FC layer is a 16 $\times$ 16 transformation with sigmoid activation. The queue length is set as 4, 2, 3 for the ETH dataset, ZARA datasets and otherwise datasets respectively. For the loss function, the λ and $margin$ for the regularizer L_c are 0.1 and 0.5 respectively. The m in the variety loss is set as 20. The batch size is 64 and the learning rate is 0.001 with Adam optimizer.

4.3 Standard Evaluations

We choose two basic methods linear models and LSTM [6], and several representative state-of-the-art methods for comparison. S-LSTM [1] and SGAN [5]

Table 1. Quantitative comparisons on the ETH and UCY datasets. ADE/FDE are reported as in meters. All the models observe for 3.2 seconds and predict for the next 4.8 seconds.

Methods	ETH	HOTEL	UNIV	ZARA1	ZARA2	Avg.
Linear	0.91/1.97	0.42/0.81	0.70/1.33	0.58/1.23	0.64/1.21	0.65/1.31
LSTM	1.16/2.20	0.48/0.86	0.57/1.20	0.47/0.99	0.39/0.82	0.61/1.21
S-LSTM	0.84/1.85	0.45/0.86	0.55/1.14	0.35/0.76	0.36/0.77	0.51/1.08
SGAN	0.77/1.41	0.44/0.88	0.75/1.50	0.35/0.69	0.36/0.73	0.53/1.04
Sophie	0.70/1.43	0.76/1.67	0.54/1.24	0.30/0.63	0.38/0.78	0.54/1.15
MATF	1.01/1.75	0.43/0.80	0.44/0.91	0.26/0.45	0.26/0.57	0.48/0.90
STGAT	0.76/1.61	0.32/0.56	0.52/1.10	0.34/0.74	0.31/0.71	0.45/0.94
Ours_{IC}	0.70/1.35	0.30/0.52	0.52/1.09	0.37/0.76	0.30/0.63	0.44/0.87
$\text{Ours}_{\text{IC-SC}}$	0.70/1.36	0.27/0.48	0.51/1.08	0.35/0.74	0.28/0.61	0.42/0.85
Ours (full)	**0.66/1.21**	**0.27/0.46**	**0.50/1.07**	**0.33/0.68**	**0.28/0.60**	**0.41/0.80**
Rel. gain	+13.15%	+15.63%	+3.85%	+2.94%	+9.68%	+9.05%

are the famous deterministic method and stochastic method that combine deep learning with spatial-only interaction respectively. The most recent approaches like Sophie [21], MATF [35] and STGAT [7] incorporate information from either static scenes or spatio-temporal dependencies. To verify the effectiveness of Individual Context Module and Social-aware Context Module, we adopt two variant of our methods, Ours_{IC} and $\text{Ours}_{\text{IC-SC}}$. In accord with [5, 21, 35, 7], the latent variables employed in the variants Ours_{IC} and $\text{Ours}_{\text{IC-SC}}$ are sampled from $\mathcal{N}(0, 1)$, and the results are reported by sampling 20 times to choose the best prediction. "Rel. gain" shows the relative ADE gain of our full model (marked in red) compared with the latest method STGAT (marked in blue).

Table 2. Quantitative comparisons on the SDD dataset. ADE/FDE are reported in pixel coordinates at 1/5 resolution followed [12]. All the models observe the agents 3.2 seconds and then make predictions in the next 1∼4 seconds.

Methods	1.0 sec	2.0 sec	3.0 sec	4.0 sec	Avg.
Linear	1.52/2.90	2.10/4.32	3.01/6.23	3.10/6.33	2.43/4.95
LSTM	1.38/2.05	2.04/3.48	3.02/5.91	3.07/5.96	2.38/4.35
S-LSTM	1.33/2.02	2.00/3.46	3.03/5.86	3.05/5.91	2.35/4.31
SGAN	1.37/2.26	2.50/4.95	2.82/5.54	2.85/5.78	2.39/4.63
STGAT	1.19/1.68	1.69/2.90	2.70/5.22	2.83/5.37	2.10/3.79
Ours_{IC}	1.10/1.66	1.70/2.90	2.55/5.02	2.65/5.13	2.00/3.68
$\text{Ours}_{\text{IC-SC}}$	1.09/1.65	1.68/2.87	2.52/4.96	2.61/5.08	1.98/3.64
Ours (full)	**1.08/1.63**	**1.64/2.83**	**2.48/4.91**	**2.57/5.02**	**1.94/3.60**
Rel. gain	+10.19%	+2.96%	+8.14%	+9.19%	+7.62%

As presented in the Table 1 and 2, linear method and LSTM suffer from bad performance since they are too shallow to consider the surrounding context.

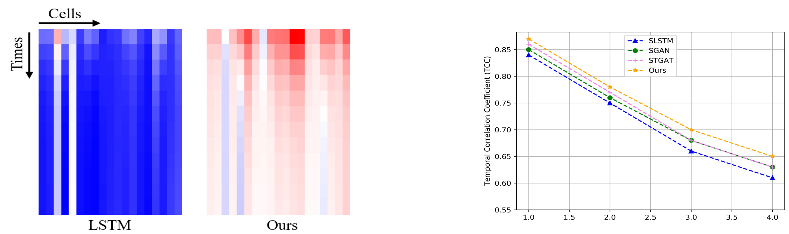


Fig. 4. (a): Memory cell visualization for LSTM and our method. Although the memory capacity decreases over time for both models (light blue → dark blue for LSTM, dark red → light red for ours), most of the cells in ours remain positive, which implies that they keep track of the evolving motion patterns from the historical context. **(b):** Comparison of TCC among state-of-the-art methods and ours. Our method enjoy high TCC, which indicates an effective capture of temporal correlation.

Compared with the variant Ours_{IC} with the state-of-the-methods, Ours_{IC} has already shown advantages across different datasets. It indicates us that the explicit temporal dependencies extraction among multiple frames is valuable to enhance the performance. $\text{Ours}_{\text{IC-SC}}$ makes some improvement with social-aware features refinement. The performance gap between $\text{Ours}_{\text{IC-SC}}$ and our full model empirically shows that the guidance of static scene context is useful for multimodal predictions.

5 Discussion

5.1 Memory Cell Visualization

As illustrated in Fig.4(a), we compare the memory capacity of LSTM and our method via cell activation. Red denotes a positive cell state, and blue denotes a negative one. The most of cells in vanilla LSTM are negative. In contrast, our model keeps track of the context throughout the prediction. Although the memory capacity of our model recedes over time (from dark red to light red), it still stays active. These results inspire us that the frame-by-frame observation used by LSTM is prone to get short sight. Instead, explicit modeling on the dependencies in multiple frames improves the captures of long-term motion.

5.2 The Capture of Motion Pattern

Fig.4(b) summarizes the quantitative results of the TCC for different methods in the SDD dataset. By learning the spatio-temporal context of dynamic agents, Our method outperforms the state-of-the-art methods (SLSTM, SGAN, STGAT), especially on the long-term predictions(4s). TCC declines consistently for different methods as the prediction duration goes on. It is reasonable since the temporal correlation of longer trajectory is harder to be learned.

Table 3. Comparisons of non-linear trajectories on the ETH and UCY datasets. ADE/FDE are reported as in meters. "Ours" denotes the proposed full model.

Methods	ETH	HOTEL	UNIV	ZARA1	ZARA2	Avg.
Linear	1.01/2.21	0.50/0.96	0.80/1.56	0.62/1.33	0.99/1.95	0.78/1.60
LSTM	1.08/2.28	0.57/1.02	0.65/1.37	0.49/1.04	0.60/1.30	0.68/1.40
S-LSTM	0.92/2.01	0.48/0.92	0.61/1.29	0.38/0.79	0.50/1.06	0.58/1.21
SGAN	0.86/1.54	0.52/1.03	0.81/1.61	0.39/0.76	0.49/0.97	0.61/1.18
STGAT	0.86/1.67	0.39/0.71	0.60/1.33	0.39/0.85	0.50/1.14	0.55/1.14
Ours (q=2)	0.73/1.37	0.34/0.63	0.61/1.29	**0.36/0.76**	**0.48/1.04**	0.50/1.02
Ours (q=3)	0.71/1.35	**0.33/0.58**	**0.60/1.28**	0.38/0.81	0.49/1.09	**0.50/1.01**
Ours (q=4)	**0.70/1.29**	0.34/0.60	0.60/1.31	0.39/0.83	0.49/1.09	0.50/1.02

5.3 Exploration on the Queue Length

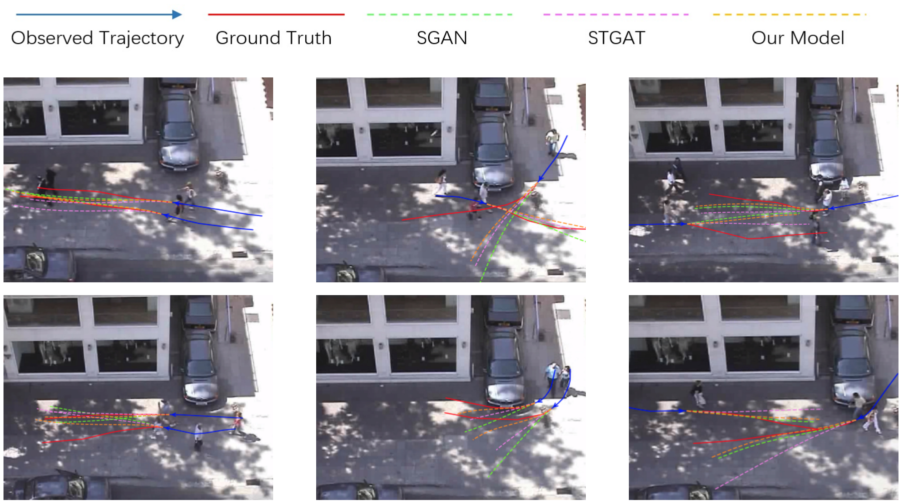
An important setup in our model is the proposed queue that understand the long-term motion. Hence, we study the effect of queue length on non-linear cases which are usually treated as hard cases. As shown in the Table 3, linear model and LSTM suffer from the unsatisfactory performance. Compared with the methods S-LSTM, SGAN and STGAT, our model enjoys relatively lower error. With the variation of queue length, our performance is robust and competitive against state-of-the-art methods. The model with long queue length ($q = 4$) is the most suitable for the ETH dataset, where most of the trajectories are highly non-linear. Short queue length ($q = 2$) works better in the ZARA1 and ZARA2 datasets, where the trajectories are faintly non-linear. Compared with linear trajectories, We speculate that the motion states of non-linear trajectory are temporally correlated within a relatively long range, where long queues capture long-term motions better than short queues.

5.4 Social Behaviors Understanding

In the scenario of real-world applications, it is imperative to handle the social interactions in the multi-agent system. Therefore, we verify whether our method well perceive the social behaviors in the crowd. As shown in the Fig.5, we select three common scenarios involve social behaviors, "Walk in parallel", "Turing" and "Face-to-Face". From the comparison between the ground truth and various predictions, we could observe that the trajectories predicted by our method are close with ground truth. Moreover, our predictions are reliable that no collisions or large deviations appear during the whole forecasting period. It indicates that our model predict multiple trajectories for each agent that cohere in time and space with the other agents.

5.5 Analysis of Multimodal Predictions

In order to evaluate the quality of multimodal predictions, we visualize the diverse trajectories predicted by our model. The top row in the Fig.6 reports the

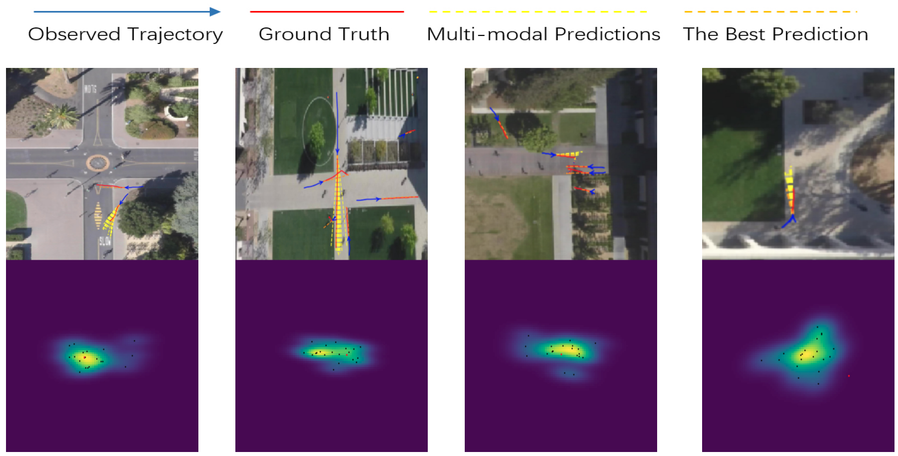


(a) Walk in parallel (b) Turning (c) Face-to-face

Fig. 5. Comparison among our model, SGAN and STGAT in different scenarios. These visualizations show that 1): Our model is capable of generating convincing trajectories that are closer to the ground truth than other state-of-the-art methods. 2): The social interactions across agents are well captured by our model. Our predictions avoid collision during the whole forecasting period.

multimodal predictions for one agent of interest. Rather than using a predefined Gaussian noise, the learnable latent variable z benefits from the semantic cues of the static scene context. Our predictions (yellow lines) suggest plausible trajectories that are close to the ground truth (red line), instead of producing a wide spread of candidates randomly. For instance, in the scenario of "Crossroad" and "Intersection", it is possible for the target agent to turn left or right at the endpoint of observation. Our model provides predictions that in line with common sense. In the scenario of "Sideway" and "Corner", the target agent has limited choices for the future trajectories due to the constraint of scene layout. In these kinds of scenarios, all our predictions are moving towards reasonable directions. Hence, our model has good interpretability with the incorporation of scene information.

In the bottom row, we investigate the uncertainty of predictions with distribution heatmap. Here we estimate the distribution of the predicted destination (black point) via kernel density estimation, and then apply the true destination (red point) to this distribution. The brighter the location, the more possibility that the point belongs to the distribution. Our visualization shows that the true destination usually appear in the bright locations. It indicates our predictions enjoy low uncertainty.



(a) Crossroad (b) Intersection (c) Sideway (d) Corner

Fig. 6. The visualization of the multimodal predictions in four scenes. **Top row:** we plot multiple possible future trajectories for one agent of interest. **Bottom row:** we visualize the distribution heatmap of destinations (location at the final frame) via kernel density estimation. The predicted destinations and ground truth are shown as black points and red point respectively. The distribution heatmap shows that our model not only provides semantically meaningful predictions, but also enjoys low uncertainty.

6 Conclusions

In this paper, we proposed a novel method DSCMP to highlight the three core elements of contextual understanding, *i.e.* spatial interaction, temporal coherence and scene layout, for multi-agent motion prediction. We designed a differentiable queue mechanism embedded on LSTM to capture the spatial interactions across agents and temporal coherence in long-term motion. And a learnable latent variable was introduced to learn the semantics of scene layout. In order to understand the uniqueness of DSCMP, we also proposed a metric Temporal Correlation Coefficient (TCC) to evaluate the temporal correlation of predicted motion. Extensive experiments on three benchmark datasets demonstrate the effectiveness of our proposed method. For the future research on autonomous applications, this work sheds a light on the modelling of spatio-temporal dependencies in multiple frames and the semantic cues from scene layout.

7 Acknowledgments

This work is partially supported by the SenseTime Donation for Research, HKU Seed Fund for Basic Research, Startup Fund and General Research Fund No.27208720. This work is also partially supported by the Major Project for New Generation of AI under Grant No. 2018AAA0100400 and by the National Natural Science Foundation of China under Grant No. 61772118.

References

1. Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., Savarese, S.: Social lstm: Human trajectory prediction in crowded spaces. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 961–971 (2016)
2. Bisagno, N., Zhang, B., Conci, N.: Group lstm: Group trajectory prediction in crowded scenarios. In: Proceedings of the European conference on computer vision (ECCV). pp. 0–0 (2018)
3. Choi, C., Dariush, B.: Looking to relations for future trajectory forecast. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 921–930 (2019)
4. Fragkiadaki, K., Levine, S., Felsen, P., Malik, J.: Recurrent network models for human dynamics. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4346–4354 (2015)
5. Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A.: Social gan: Socially acceptable trajectories with generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2255–2264 (2018)
6. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
7. Huang, Y., Bi, H., Li, Z., Mao, T., Wang, Z.: Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6272–6281 (2019)
8. Ivanovic, B., Pavone, M.: The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2375–2384 (2019)
9. Karasev, V., Ayvaci, A., Heisele, B., Soatto, S.: Intent-aware long-term prediction of pedestrian motion. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 2543–2549. IEEE (2016)
10. Kim, B., Kang, C.M., Kim, J., Lee, S.H., Chung, C.C., Choi, J.W.: Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC). pp. 399–404. IEEE (2017)
11. Kosaraju, V., Sadeghian, A., Martín-Martín, R., Reid, I., Rezatofighi, H., Savarese, S.: Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. In: Advances in Neural Information Processing Systems. pp. 137–146 (2019)
12. Lee, N., Choi, W., Vernaza, P., Choy, C.B., Torr, P.H., Chandraker, M.: Desire: Distant future prediction in dynamic scenes with interacting agents. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 336–345 (2017)
13. Lerner, A., Chrysanthou, Y., Lischinski, D.: Crowds by example. In: Computer graphics forum. vol. 26, pp. 655–664. Wiley Online Library (2007)
14. Ma, Y., Zhu, X., Zhang, S., Yang, R., Wang, W., Manocha, D.: Trafficpredict: Trajectory prediction for heterogeneous traffic-agents. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 6120–6127 (2019)
15. Manh, H., Alaghband, G.: Scene-lstm: A model for human trajectory prediction. arXiv preprint arXiv:1808.04018 (2018)
16. Park, S.H., Kim, B., Kang, C.M., Chung, C.C., Choi, J.W.: Sequence-to-sequence prediction of vehicle trajectory via lstm encoder-decoder architecture. In: 2018 IEEE Intelligent Vehicles Symposium (IV). pp. 1672–1678. IEEE (2018)

17. Pellegrini, S., Ess, A., Schindler, K., Van Gool, L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 261–268. IEEE (2009)
18. Petrich, D., Dang, T., Kasper, D., Breuel, G., Stiller, C.: Map-based long term motion prediction for vehicles in traffic environments. In: 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013). pp. 2166–2172. IEEE (2013)
19. Robicquet, A., Sadeghian, A., Alahi, A., Savarese, S.: Learning social etiquette: Human trajectory understanding in crowded scenes. In: European conference on computer vision. pp. 549–565. Springer (2016)
20. Rudenko, A., Palmieri, L., Herman, M., Kitani, K.M., Gavrila, D.M., Arras, K.O.: Human motion trajectory prediction: A survey. arXiv preprint arXiv:1905.06113 (2019)
21. Sadeghian, A., Kosaraju, V., Sadeghian, A., Hirose, N., Rezatofighi, H., Savarese, S.: Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1349–1358 (2019)
22. Shah, R., Romijnders, R.: Applying deep learning to basketball trajectories. arXiv preprint arXiv:1608.03793 (2016)
23. Synnaeve, G., Dupoux, E.: A temporal coherence loss function for learning unsupervised acoustic embeddings. In: SLTU. pp. 95–100 (2016)
24. Tai, K.S., Socher, R., Manning, C.D.: Improved semantic representations from tree-structured long short-term memory networks. arXiv preprint arXiv:1503.00075 (2015)
25. Vasquez, D.: Novel planning-based algorithms for human motion prediction. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 3317–3322. IEEE (2016)
26. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
27. Vemula, A., Muelling, K., Oh, J.: Social attention: Modeling attention in human crowds. In: 2018 IEEE international Conference on Robotics and Automation (ICRA). pp. 1–7. IEEE (2018)
28. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7794–7803 (2018)
29. Xu, Y., Piao, Z., Gao, S.: Encoding crowd interaction with deep neural network for pedestrian trajectory prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5275–5284 (2018)
30. Yamaguchi, K., Berg, A.C., Ortiz, L.E., Berg, T.L.: Who are you with and where are you going? In: CVPR 2011. pp. 1345–1352. IEEE (2011)
31. Yi, S., Li, H., Wang, X.: Pedestrian behavior modeling from stationary crowds with applications to intelligent surveillance. IEEE transactions on image processing **25**(9), 4354–4368 (2016)
32. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318 (2018)
33. Zhang, P., Ouyang, W., Zhang, P., Xue, J., Zheng, N.: 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 (2019)

34. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2881–2890 (2017)
35. Zhao, T., Xu, Y., Monfort, M., Choi, W., Baker, C., Zhao, Y., Wang, Y., Wu, Y.N.: Multi-agent tensor fusion for contextual trajectory prediction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12126–12134 (2019)
36. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)