BGM: Building a Dynamic Guidance Map without Visual Images for Trajectory Prediction

Beihao Xia*, Conghao Wong*, Heng Li, Shiming Chen, Qinmu Peng, Xinge You

School of Electronic Information and Communications, Huazhong University of Science and Technology, Hubei, China
{xbh_hust, conghao_wong, liheng, shimingchen, pengqinmu, youxg}@hust.edu.cn

Abstract
Visual images usually contain the informative context of the environment, thereby helping to predict agents’ behaviors. However, they hardly impose the dynamic effects on agents’ actual behaviors due to the respectively fixed semantics. To solve this problem, we propose a deterministic model named BGM to construct a guidance map to represent the dynamic semantics, which circumvents to use visual images for each agent to reflect the difference of activities in different periods. We first record all agents’ activities in the scene within a period close to the current to construct a guidance map and then feed it to a Context CNN to obtain their context features. We adopt a Historical Trajectory Encoder to extract the trajectory features and then combine them with the context feature as the input of the social energy based trajectory decoder, thus obtaining the prediction that meets the social rules. Experiments demonstrate that BGM achieves state-of-the-art prediction accuracy on the two widely used ETH and UCY datasets and handles more complex scenarios.

Introduction
Trajectory prediction is to forecast agents’ locations in the future based on their past positions. It is popular and widely applied in self-driving (Lee et al. 2017), robotic navigation (Trautman and Krause 2010), tracking (Robicquet et al. 2016) and more other areas. Many researchers are committed to this topic and made their contribution. Early works like (Alahi et al. 2016) focus on predicting agents’ future trajectories by just modeling social interaction. In addition to being affected by others, agents always consider planning their future routes under environmental constraints. Works like (Xue, Huynh, and Reynolds 2018) extract visual features of the scenes by CNNs to help predict agents’ future trajectories, which shows their effectiveness. The CNN features represent the scene environment around agents so that models could learn agents’ activities in different scene characteristics, to give better predictions in other scenes with similar looks. More additional inputs like scene segmented images and person behaviors are also adopted in (Liang et al. 2019) to describe the agents’ past actions and their corresponding context comprehensively to achieve better results.

Most existing methods like (Manh and Alaghband 2018, Sadeghian et al. 2019, Liang et al. 2019, Kosaraju et al. 2019) tackle agents’ movement and scene context separately. They do not consider the impact of real agents’ activities and the dynamic changes of their preferences in the scene when modeling the agent-scene interaction. An easily overlooked fact is that the scene interaction is not static as time goes by. Several researchers treat the scene interaction as the influence of static background objects (such as trees, roads, and buildings) on agents. How to describe the interaction between a standing pedestrian and another moving one? Previous methods usually consider it as the agent-agent interaction. However, it is incredibly similar to the impact of trees on other agents since they both hinder others’ movements. It means that the interaction among different agents and the scene are connected rather than separate. Thus, it is not comprehensive that only use the visual images or static segmented images to model the generally talked scene interaction. We regard the factors that may affect the scene’s interactive behaviors as a whole and summarize it as the scene context. The scene context contains both dynamic factors and interaction producers, which have no categories limitations. For instance, standing pedestrians and parking cars are part of the scene context, similar to trees and stores’ doors.

As shown in Fig. 1, we offer different maps that gather the scene context. However, (i) Static visual images cannot describe the dynamics of the scene, since the flow of agents and their interaction behaviors could change over time; (ii) The segmented map only shows the rough scene annotations.

*indicates equal contribution.
However, it does not point out the interactive components like Zara’s entrance as the actual map does. The agent’s historical trajectory is the reflection of his past activities (Hong et al. 2020). Considering the disadvantages of these maps, we propose a novel dynamic guidance map to model the changeable scene context. We select a proper time window to record available trajectories in the scene and then predict agents’ future trajectories with the help of the distribution of observed trajectories near the agent, thus reflecting the scene context’s dynamics. The guidance map contains the dynamic change of walking areas and interaction preferences, as well as interactive components like the entrance of shops, which static visual images and segmented images do not contain.

To better model social interaction, most current methods judge each agent’s contribution based on their relative distance among the participants. However, they cannot well model the interaction situation when the distances are the same. For example, when a pedestrian is running, and another is walking towards another same pedestrian at the same distance. These models cannot better describe this case since the two people have different contributions. Besides, previous methods also limit the number of agents participating in the interaction due to their network structures. Inspired by the work (Helbing and Molnar 1995; Yamaguchi et al. 2011), we build a social energy field and regard that the agents will always pass through lower social energy areas. We design a social module to make the model’s outputs socially acceptable based on the social energy field, which needs not limit the number of agents participating in the interaction. Besides, the universal form of energy function makes the model more robust to various social situations.

The main contributions of this work are listed as follows:

- A guidance map with more semantics is constructed better to predict future trajectories in response to the dynamic context.
- A social module is designed to make the predictions in line with social etiquette.
- Experiments show that our BGM has state-of-the-art prediction performance on ETH and UCY datasets compared with other models.

The rest of the paper is organized as follows: We give a brief overview of related work in Section 2. In Section 3, we will give a detailed description of our model. We show the performance of BGM in Section 4.

**Related Work**

**Trajectory Clustering Analysis**

Finding a suitable representation of historical trajectories is a significant step for trajectory prediction. It is essential to study the task of trajectory data mining. Agglomerative clustering model (Zheng et al. 2009) is explored to mine the interesting locations to catch trajectories. Considering different lengths of trajectories, the work (Brox and Malik 2010) construct the affinity matrix. Some spectral clustering models like (Zhang et al. 2016) are proposed to capture causal relationships between time series.

**Description of Social Interaction**

The earliest work Social Force Model (Helbing and Molnar 1995) describes the social interaction with speed, acceleration, direction, and other social force items. In addition to these factors, later works like (Kitani et al. 2012; Pellegrini et al. 2009) add different aspects to model interaction among agents. With the development of data-driven methods, Social LSTM (Alahi et al. 2016), which utilizes LSTM (Hochreiter and Schmidhuber 1997) and an additional social pooling layer to extract social features, is proposed to learn to predict human behaviors in crowded scenes. Besides, works like (Fernando et al. 2018; Vemula, Muelling, and Oh 2018) take advantage of the attention mechanism to judge each agent’s contribution to the interaction considering the different influences between varied agents and the target one. Due to graph neural networks’ success, some methods based on graph attention networks like (Huang et al. 2019; Kosaraju et al. 2019; Sun, Jiang, and Lu 2020) also try to model social interaction with the graph network dynamically. However, these methods have limitations on the number of nodes in the graph, limiting the number of agents participating in the interaction. Works like (Xu, Piao, and Gao 2018; Sadeghian et al. 2019) define a maximum number of agents participating in the interaction. Others like (Gupta et al. 2018) take a spatial window to obtain the neighbors of the current agent and model interactions among them, which may ignore the further agent with a more outstanding contribution. To deal with all possible social interaction, the approach (Zhao et al. 2019) construct a new tensor, which encodes trajectories of all agents in the crowded scene. Unlike other works, (Chandra et al. 2019) and (Ma et al. 2019) also considered the heterogeneity of agents caused by the different categories of them.

**Description of Scene Interaction**

The scene context would also change agents’ future movements. Work (Robicquet et al. 2016) annotates scenes with static semantic segmentation results since they regard that people always walk under the rules of common sense. Some works (Sadeghian et al. 2019; Kosaraju et al. 2019) use Convolutional Neural Networks to extract the visual features. With the development of semantic segmentation, more detailed extra information are available in recent works like (Liang et al. 2019; Manh and Alaghband 2018; Xue, Huynh, and Reynolds 2018). What is more, agents’ physical activities are used in (Liang et al. 2019) to make a better prediction. However, it should be noted that these methods that obtain scene information with the help of additional tools make it lose the connection between the scene and agents, resulting in the inability to make full use of the auxiliary information when making predictions.

**Model**

**Problem Definition**

Given a sequence of observed historical trajectory of $i$-th agent from time step $T_0$ to $T_0 + T_{obs} - 1$, the prediction task is to find a possible future trajectory from time step $T_0 + T_{obs}$ to $T_0 + T_{obs} + T_{pred} - 1$. $T_{obs}$ and $T_{pred}$ represent the length of observations and predictions respectively. To describe more
clearly, we define \( T_1 = T_0 + T_{\text{obs}} \) and \( T_2 = T_1 + T_{\text{pred}} \). Use \( p_t^i = (x_t^i, y_t^i) \) to represent the coordinate of \( i \)-th agent at time step \( t \). Its observed trajectory is denoted by \( X_{T_0:T_1} = \{p_{T_0+t}^i \in \mathbb{R}^2 | t = 0, \ldots, T_{\text{obs}}-1 \} \) and ground truth trajectory by \( Y_{T_1:T_2} = \{p_{T_1+t}^i \in \mathbb{R}^2 | t = 0, \ldots, T_{\text{pred}}-1 \} \). The prediction problem is to find a model \( F(\cdot) \) to predict the agent’s reasonable future trajectory \( \hat{Y}_{T_1:T_2} = F(X_{T_0:T_1}) \).

Overview

As shown in Fig. 2, the BGM contains three main parts: a Historical Trajectory Encoder (HTE), a Dynamic Context Encoder (DCE), and a Social Energy based Trajectory Decoder, which includes a Trajectory Decoder and a Social Module.

Historical Trajectory Encoder

We build a historical trajectory encoder to focus on the essential behaviors that may affect the future, which occurred in the distant past during the observation period. Previous methods like (Alahi et al., 2016; Gupta et al., 2018) pay more attention to the state of trajectory at the last moments closer to the current due to the structure of basic LSTM. We select several time windows with different lengths to obtain a subset of historical trajectories and then feed them to LSTMs (weights are shared) to get the trajectory status at all other observation moments. For agent \( i \), we divide his historical trajectory \( X_{T_0:T_1} \) into \( T_{\text{obs}} \) different scales as \( \{X_{T_0:T_0+t}^i | t = 1, \ldots, T_{\text{obs}} \} \). Then we input all of them to the LSTMs to obtain a set of multi-level historical trajectory features \( F_{T_0:T_1}^i = \{\text{LSTM}[\phi(X_{T_0:T_0+t}^i)] | t = 1, \ldots, T_{\text{obs}} \} \), where \( \phi \) denotes the position embedding function. To make better use of this set of features, we introduce a set of trainable weight matrices \( W_{T_0:T_1} = \{w_1, w_2, \ldots, w_{T_{\text{obs}}} \} \) and a bias vector \( b \) to weight each of \( F_{T_0:T_1}^i \), to obtain the final sequence feature \( f_{0:T_1}^i \)

\[
f_{S,T_0:T_1}^i = \sum_{t=1}^{T_{\text{obs}}} w_t \text{LSTM}[\phi(X_{T_0:T_0+t}^i)] + b. \tag{1}
\]

Dynamic Context Encoder

This encoder computes a dynamic context map called Guidance Map, which displays other agents’ trajectory distribution near the target agent during observation. We use a new update strategy to maintain the time efficiency of the guidance map. We first define the time period this encoder used to create the guidance map as the Record Period \( T_R = \{t_1, t_2, \ldots, t_m \} \). We shift this time window \( T_R \) forward through training or testing to capture specific interactive cases to reflect the scene context’s dynamic changes. For
agents appeared in \( T_R \), we collect all of their observed positions as the Record Positions \( X(T_R) = \{ p_i = (x_i, y_i) \} \).

We adopt an adaptive selection strategy to make the guidance map more robust by limiting the number of records and this window’s length. We show the details in Algorithm 1.

We use a 2-dimension grid \( M(T_R) \in \mathbb{R}^{H \times W} \) to represent the distribution of these recorded trajectories in the scene. Denote the mapping function from actual position \( p \) to grid position \( p_g \) by \( p_g = m(p) \). The complete guidance map in a specific \( T_R \) is computed by:

\[
M(T_R) = \sum_{p \in X(T_R)} \delta(p; m(p_0)),
\]

where \( \delta(p) \in \mathbb{R}^{H \times W} \) and \( \delta(p; x_0) = \begin{cases} 1, & \text{if } p = x_0, \\ 0, & \text{otherwise}. \end{cases} \) \( (2) \)

For target agent \( i \), we take a part of the complete map above surrounding him as his Local Guidance Map \( M^i_L(T_R) = M(T_R)[m(p) - (l, l) : m(p) + (l, l)] \), where \( p \) represents his last observed position, and the side length of the local guidance map is \( 2l \). Then we use a Context CNN (C-CNN) (detailed structure can be seen in Fig 5) to obtain his final context feature. Formally,

\[
f^C_i(T_R) = \text{C-CNN}[M^i_L(T_R)]. \quad (3)
\]

For the context feature \( f^C_i(T_R) \), we can see that it will become different when \( i \) and \( T_R \) changes. This well-designed context feature could adapt to different periods and different agents, reflecting the dynamic adaptability of our BGM.

### Social Energy based Trajectory Decoder

A joint feature \( f^i \) is concatenated by the sequence feature \( f^i_L \) and context feature \( f^C_i(T_R) \) above to show both historical activities and future intention of the target agent \( i \) according to his local context. (Subscript \( t_1 : T_R \) is omitted for a clear expression when there are no conflicts.)

We use an MLP (denoted by MLP\(_D\)) to process this joint feature to obtain the preliminary prediction, formally \( Y^i_d = \text{MLP}_D(f^i) \). Note that the joint feature \( f^i \) contains the record of all historical behaviors of agent \( i \), which are reflected by his observed trajectory. Therefore, the output dimension of MLP\(_D\) is set to \( 2T_{pred} \) to obtain the entire prediction just by one step of the calculation, which reduces the cumulative effect of errors due to the circular calculations and the information omissions. Besides, we employ an Interaction Predictor (marked with \( P \)) and a Social Energy based Discriminant Function (marked with \( D \)) to make sure that the prediction is with social etiquette. We call the \( P \) and the \( D \) together as the Social Module.

### Social Energy based Discriminant Function

We use a widely applicable and reasonable energy function method to describe agents’ social-behavioral preferences. Note that we build this function for each agent, which means that each agent in the scene has his discriminant function different from others. Define the static social energy exhibited by some other agents at position \( p_o \) to the target agent as:

\[
f(p; p_o, r, a) = \begin{cases} a - \frac{a}{r} \| p - p_o \|, & \| p - p_o \| \leq r, \\ 0, & \| p - p_o \| > r. \end{cases} \quad (4)
\]

Where \( p \in \mathbb{R}^2 \) indicates the location variable for current agent’s energy function, \( r \) and \( a \) represent the range and amplitude of energy respectively. We classify items that could affect agents’ future trajectories and activities into three parts: destination, interplay, and social etiquette. Use \( \Phi = \{\lambda_d, \lambda_1, \lambda_s, r_d, r_i, r_s\} \) to denote a set of social parameters. The social energy function for agent \( i \) is defined as:

\[
E^i(p; \Phi) = \lambda_d E^i_D(p; r_d) + \lambda_1 E^i_I(p; r_i) + \lambda_s E^i_S(p; r_s)
\]

\[
= \lambda_d \sum_{p_o \in Y^i_d} f(p; p_o, r_d, -1) + \lambda_1 \sum_{j \in J^i} \sum_{p_j \in Y^j} d(i, j) v(i, j) f(p; p_j, r_i, -1) + \lambda_s \sum_{r_j \in J^i} \sum_{p_j \in Y^j} f(p; p_j, r_s, 1). \quad (5)
\]

(i) Destination \( E^i_D(p; r_d) \). The destination refers to the desired future trend under one’s own will. We use the preliminary prediction, \( Y^i_d \), to represent the original future trends of agent \( i \).

(ii) Interplay \( E^i_I(p; r_i) \). Pedestrians always follow their companions and avoid others coming oncoming to prevent probable collisions. Agents’ avoidance behaviors have a relationship with other directions and velocities. \( J^i = \{j_1, ..., j_n\} \) represents all other agents appeared during the
prediction period except agent i. We use d(i, j) to describe the overall direction difference for the observed trajectories of between agent i and j, and v(i, j) to describe the relative velocity. Formally:

\[
d(i, j) = \left(\hat{p}_{T_0}^i - p_{T_1}^j\right)^T \left(\hat{p}_{T_0}^i - p_{T_1}^j\right)
\]

\[
v(i, j) = \left\|\hat{p}_{T_0}^i - p_{T_1}^j\right\| \left\|\hat{p}_{T_0}^i - p_{T_1}^j\right\|,
\]

(iii) Social Etiquette. When planning future journeys, pedestrians always avoid getting too close to others to arouse their disgust. It is also called a safe social distance.

We use an Interaction Predictor to generate a little local variation to the preliminary prediction as:

\[
D(\hat{Y}_0^i) = \sum_{p \in \hat{Y}_0^i} E^i(p).
\]

**Interaction Predictor** We use an Interaction Predictor to generate a little local variation to the preliminary prediction according to the social energy function to align with the social rules. For preliminary prediction of agent i at time t, define the k order (k \geq 1) interaction predictor as:

\[
P^k(p_t^i) = \left\{
\begin{array}{ll}
    p_t^i - \theta \frac{dE^i(p_t^i)}{dp} & , k = 1, \\
    P^1(P^{k-1}(p_t^i)) & , k \geq 2.
\end{array}
\right.
\]

θ represents the update coefficient. To make sure that the prediction is with social etiquette, we gradually increase the order k of the interaction predictor until D gives a steady value. Formally, the final prediction is obtained by:

\[
\hat{Y}_{T_1:2}^i = P^k(\hat{Y}_0^i) = \{P^k(p_t^i) | t = T_1, ..., T_2 - 1\},
\]

where |D[P^k(\hat{Y}_0^i)] - D[P^{k-1}(\hat{Y}_0^i)]| \leq \epsilon.

**Loss Functions and Implementation Details**

For the historical trajectory encoder, context CNN and MLP_D, we treat all these components as a whole end-to-end network and use the L2 loss function to train it. For each point in the preliminary prediction of i-th agent p_t^i \in \hat{Y}_0^i and their ground truth p_t^{real} \in Y^i, the loss function can be written as:

\[
L = \sum_{t} \sum_{i} \left\|p_t^i - p_t^{real}\right\|.
\]

We train the network with the Adam optimizer with a learning rate of 0.01 in 500 epochs. For the original T_{obs} \times 2 historical trajectories, we embed them into T_{obs} \times 64 vectors. The dimensions of the hidden state of LSTM is 64. We also use one layer of MLP with a 256-dimension’s output and ReLU activation to process the sequence feature. We set the resolution of the grid guidance map to 0.25 meters per grid. For the current task of predicting future 4.8 seconds, we use a square window with side length 2l = 8 meters to prepare local guidance maps for each agent. We also guide two MLPs to increase the dimension of both the sequence and the context feature into 256. We concatenate the sequence feature and the context feature into a 512-dimension feature vector, and then feed it to the MLP_D (512 \rightarrow T_{pred} \times 64 \rightarrow T_{pred} \times 2) to get the preliminary prediction. ReLU activations are applied in MLP_D except for its last output layer.

We build each agent’s energy function with a grid format like the social module’s guidance map. Its resolution is set to 0.1 meters per grid. Due to a safe social distance is about 0.1 meters (Sadeghian et al. 2019), we set social parameters as \( \Phi = \{1.0, 1.0, 0.2, 2.0, 1.5, 1.5\} \). The update coefficient θ is set to 0.001. In experiments, we use a 10-order interaction predictor to make the final prediction socially acceptable.

**Experiments**

**Datasets** We have evaluated BGM on two public human trajectory datasets: ETH (Yamaguchi et al. 2011) and UCY (Leal-Taixé et al. 2014). These two datasets contain five scenarios: eth, hotel, zara1, zara2, and univ. They are annotated trajectories of pedestrians in real-world scenes.

**Evaluation Metrics** Similar to previous works like (Alahi et al. 2016, Gupta et al. 2018, Kosaraju et al. 2019), we use the Average Displacement Error (ADE) and the Final Displacement Error (FDE) to describe the performance. Their specific calculation method can refer to Eq.7 in (Xu, Piao, and Gao 2018). In particular, for models that generate multiple predictions, we use the minADE (Mohamed et al. 2020) among K trajectories to evaluate their performance. Our experiments show the prediction performance of agents in the future 4.8 seconds (12 frames) using their trajectories during the past 3.2 seconds (8 frames).

**Baselines** We choose deterministic models, like CIDNN (Xu, Piao, and Gao 2018), SR-LSTM (Zhang et al. 2019), and generative models, like Social GAN (Gupta et al. 2018), STGAT (Huang et al. 2019), SoPhie (Sadeghian et al. 2019), Social-BiGAT (Kosaraju et al. 2019), and Social-STGCNN (Mohamed et al. 2020) as baselines. We reproduce these models except SoPhie and Social-BiGAT in the same environment as ours by their publicly available codes.

**Quantitative Evaluation**

**Compared with Other Methods** As shown in Table 1, SR-LSTM is a deterministic model with the best performance at present. BGM shows more competitive performance than it: Compared with SR-LSTM, ADE, and FDE of BGM reduce 13.0% and 14.4%, respectively. Authors of (Kosaraju et al. 2019) provide the results of SoPhie and Social-BiGAT when generating K = 1 trajectory. In this case, these models can be considered as deterministic models. For these two models, ADE of BGM has improved 43.7% and 34.4%, and FDE improved 43.8% and 37.6%, which means that BGM has a relatively stable prediction performance.

The ADE of our deterministic BGM is improved by 11.1% compared with the best generative model Social-STGCNN that generates 20 predictions. It should be noted that minADE of most generative models in Table 1 on UCY-zara1 and UCY-zara2 datasets have achieved a better level
than deterministic models, reflecting the difference in the application purpose between these models.

Different from ADE, FDE is a significant metric that many researchers ignore. The relative relationship between FDE and ADE reflects models’ performance in predicting agents’ future trends. Visualized results (Fig.4) shows our performance on predicting agents’ future trends. Compared with all methods listed in Table1, the FDE of BGM is only worse than Social-STGCNN’s. Especially, our FDE makes the best among the deterministic methods.

Ablation Study To verify the effectiveness of each component in our model, we perform ablation experiments. Results can be seen in the last lines in Table1.

- Dynamic Context Encoder. With the context feature’s help, our model could predict the agent’s future trajectory, considering both his historical trajectory and the contextual information. Therefore, even similar trajectories will have a different prediction due to the difference in their guidance maps, reflecting the other available interaction cases between agents and the environment. Compared with BGM w/o Social Module, the ADE and FDE of BGM improve 9.0% and 6.7%.

- Social Module. Social interaction is always inevitable in actual scenarios. We use this module to make sure that the prediction meets the social rules. From the result of BGM w/o Social Module, its ADE and FDE have deteriorated about 2.5% and 1.2% compared with BGM. Although ADE and FDE become worse than BGM w/o Social Module, we have ensured that each prediction is sufficient to meet a safe social distance, thus avoiding possible collisions to the greatest extent and satisfying the attraction situation in the group. Details can be seen in Fig4.

Qualitative Evaluation

Visualized Results We select several prediction scenarios to show the comparison between BGM and other existing models. Details can be seen in Fig4. We choose two kinds of ordinary prediction cases, meeting and following, to evaluate these models. In case (a), BGM’s prediction has shown a firm intention on social interaction that the target pedestrian wants to avoid others from being collided, consistent with her future behavior. Among other methods, only SR-LSTM’s prediction shows a similar trend. Results of both Social GAN-P and Social-STGCNN seems to lack the consideration of this interaction situation. Similar to (a), (b, c, d) show the results with specific interaction intentions. Compared with other baselines, BGM shows a better ability to describe interactions while keeping walking styles, reflecting our strong ability to adapt to different situations.

The Dynamics of Guidance Map We build a guidance map for each agent in the scene to better describe both their past and future behaviors by attaching the scene context to their historical trajectories, respectively. The guidance map’s dynamic update strategy makes it available to reflect the latest trend of agents’ activities in the scene. For instance, there is a metro entrance close to a shopping mall. Pedestrians always choose to enter the metro rather than the shopping mall in the morning when most people go to work. However, the visual scene images or segmented images can not reflect these dynamic changes. We visualized the complete guidance map in the UCY-univ scene in different periods in Fig5. According to the update strategy shown in Algorithm1, we build three different guidance maps $M(T_a)$, $M(T_b)$ and $M(T_c)$ in three different record periods $T_a$, $T_b$ and $T_c$. Fig5(d) shows the total guidance map build by all available trajectories in the whole dataset. Fig5(a) (b) and (c) show different walking preferences of pedestrians in these three periods. For example, pedestrians in $T_b$ prefer walking from right-bottom to left-top in the scene. In contrast, most pedestrians in $T_c$ prefer to move in the horizontal direction in the scene, which shows the dynamic change of agents’ flow.

We perform further comparative experiments to verify the effect of our dynamic guidance map. As shown in Table2, the result shows that the BGM performs the best when the
Figure 4: Visualized results of different models. (a), (b), (c), and (d) are four different prediction conditions, which include meeting (a and c) and following (b and d) two kinds of everyday prediction situations. Arrows indicate their walking directions.

Figure 5: The dynamics of guidance map. (a) (b) and (c) are the complete dynamic guidance maps \(M(T_a)\), \(M(T_b)\) and \(M(T_c)\) built through three different record periods \(T_a\), \(T_b\) and \(T_c\). (d) shows the map generated by all available trajectories in the UCY-univ dataset.

Table 2: Quantitative evaluation of the dynamics of guidance map. We select 3 different sets of test trajectories \(X_a\), \(X_b\) and \(X_c\) to run our prediction with their 3 corresponding complete guidance maps \(M(T_a)\), \(M(T_b)\) and \(M(T_c)\) on UCY-univ dataset. Results are shown in ADE/FDE.

| Test Set | \(M(T_a)\) | \(M(T_b)\) | \(M(T_c)\) |
|----------|-------------|-------------|-------------|
| \(X_a\)  | 0.495/1.073 | 0.496/1.075 | 0.496/1.075 |
| \(X_b\)  | 0.437/0.937 | 0.430/0.921 | 0.435/0.930 |
| \(X_c\)  | 0.458/0.991 | 0.463/1.004 | 0.456/0.987 |

test set \(X_x (x \in \{a, b, c\})\) takes the local guidance map from its corresponding complete map \(M(T_x)\) as the input. Results on the diagonal in Table 2 means that our dynamic guidance map helps the prediction, reflecting the dynamic preferences changes in the scene.

Social Module The social module fine-tunes the preliminary prediction by minimizing each agent’s social energy discriminant function to make their predictions socially acceptable. This module needs not to limit the number of agents in the scene, like most current graph-based methods. It uses a form of inferable expression, which can adapt to different situations without extra training steps. One typical interaction case is shown in Fig.6 Two groups of pedestrians walking towards each other. They will collide if they continue to walk as their historical styles. The right figure shows the value of the pedestrian’s social energy field in white clothes at each scene’s position. According to the lower energy principle, the social module suggests fine-tuning her original prediction by shifting down a little to the bottom of the scene to prevent future collisions. The social module optimizes the preliminary predictions adaptively to a new trajectory (represented by green circles) through the lowest energy optimization, making the final prediction socially acceptable. More interaction cases can be seen in Fig 4.

Conclude

This paper presents BGM for human trajectory prediction, which outperforms state-of-the-art methods on two publicly available datasets. We construct a guidance map for each agent to explain why he behaved in the past and to guide his future movement. It updates with the prediction period changes to adapt to the dynamic scene content. We also propose a social module to make the predictions in line with social rules. As shown in visualized results, BGM can give reasonable and socially acceptable predictions in complex scenarios. Despite our contributions, there are still issues that need to be addressed. For example, to get a better guidance map, we need to collect more available observations. Besides, we will try considering agents’ pose and intention to improve future works’ prediction performance.
References

Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; and Savarese, S. 2016. Social lstm: Human trajectory prediction in crowded spaces. In Proceedings of the IEEE conference on computer vision and pattern recognition, 961–971.

Brox, T.; and Malik, J. 2010. Object segmentation by long term analysis of point trajectories. In European conference on computer vision, 282–295. Springer.

Chandra, R.; Bhattacharya, U.; Bera, A.; and Manocha, D. 2019. TraPtic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 8483–8492.

Deo, N.; Rangesh, A.; and Trivedi, M. M. 2018. How would surround vehicles move? a unified framework for maneuver classification and motion prediction. IEEE Transactions on Intelligent Vehicles, 3(2): 129–140.

Fernando, T.; Denman, S.; Sridharan, S.; and Fookes, C. 2018. Soft+ hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection. Neural networks 108: 466–478.

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

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

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. Neural computation 9(8): 1735–1780.

Hong, H.; Lin, Y.; Yang, X.; Li, Z.; Fu, K.; Wang, Z.; Qie, X.; and Ye, J. 2020. HetETA: Heterogeneous Information Network Embedding for Estimating Time of Arrival. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2444–2454.

Huang, Y.; Bi, H.; Li, Z.; Mao, T.; and Wang, Z. 2019. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In Proceedings of the IEEE International Conference on Computer Vision, 6272–6281.

Kitani, K. M.; Ziebart, B. D.; Bagnell, J. A.; and Hebert, M. 2012. Activity forecasting. In European Conference on Computer Vision, 201–214. Springer.

Kosaraju, V.; Sadeghian, A.; Martín-Martín, R.; Reid, I.; Rezatofighi, H.; and Savarese, S. 2019. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. In Advances in Neural Information Processing Systems, 137–146.

Lee, N.; Choi, W.; Vernaza, P.; Choy, C. B.; Torr, P. H.; and Chandraker, M. 2017. Desire: Distant future prediction in dynamic scenes with interacting agents. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 336–345.

Liang, J.; Jiang, L.; Niebles, J. C.; Hauptmann, A. G.; and Fei-Fei, L. 2019. Peeking into the future: Predicting future person activities and locations in videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5725–5734.

Ma, Y.; Zhu, X.; Zhang, S.; Yang, R.; Wang, W.; and Manocha, D. 2019. Trafficpredict: Trajectory prediction for heterogeneous traffic-agents. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, 6120–6127.

Manh, H.; and Alaghband, G. 2018. Scene-lstm: A model for human trajectory prediction. arXiv preprint arXiv:1808.04018 .

Mohamed, A.; Qian, K.; Elhoseiny, M.; and Claudel, C. 2020. Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 14424–14432.

Pellegrini, S.; Ess, A.; Schindler, K.; and Van Gool, L. 2009. You’ll never walk alone: Modeling social behavior for multitarget tracking. In 2009 IEEE 12th International Conference on Computer Vision, 261–268. IEEE.

Rhinehart, N.; McAllister, R.; Kitani, K.; and Levine, S. 2019. Precog: Prediction conditioned on goals in visual multi-agent settings. In Proceedings of the IEEE International Conference on Computer Vision, 2821–2830.

Robicquet, A.; Sadeghian, A.; Alahi, A.; and Savarese, S. 2016. Learning social etiquette: Human trajectory understanding in crowded scenes. In European conference on computer vision, 549–565. Springer.

Sadeghian, A.; Kosaraju, V.; Sadeghian, A.; Hirose, N.; Rezatofighi, H.; and Savarese, S. 2019. 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, 1349–1358.

Sun, J.; Jiang, Q.; and Lu, C. 2020. Recursive Social Behavior Graph for Trajectory Prediction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 660–669.

Trautman, P.; and Krause, A. 2010. Unfreezing the robot: Navigation in dense, interacting crowds. In 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 797–803. IEEE.

Vemula, A.; Muelling, K.; and Oh, J. 2018. Social attention: Modeling attention in human crowds. In 2018 IEEE international Conference on Robotics and Automation (ICRA), 1–7. IEEE.

Xu, Y.; Piao, Z.; and Gao, S. 2018. Encoding crowd interaction with deep neural network for pedestrian trajectory
prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5275–5284.

Xue, H.; Huynh, D. Q.; and Reynolds, M. 2018. SS-LSTM: A hierarchical LSTM model for pedestrian trajectory prediction. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 1186–1194. IEEE.

Yamaguchi, K.; Berg, A. C.; Ortiz, L. E.; and Berg, T. L. 2011. Who are you with and where are you going? In CVPR 2011, 1345–1352. IEEE.

Zhang, P.; Ouyang, W.; Zhang, P.; Xue, J.; and 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, 12085–12094.

Zhang, Z.; Huang, K.; Tan, T.; Yang, P.; and Li, J. 2016. Red-sfa: Relation discovery based slow feature analysis for trajectory clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 752–760.

Zhao, T.; Xu, Y.; Monfort, M.; Choi, W.; Baker, C.; Zhao, Y.; Wang, Y.; and Wu, Y. N. 2019. Multi-agent tensor fusion for contextual trajectory prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 12126–12134.

Zheng, Y.; Zhang, L.; Xie, X.; and Ma, W.-Y. 2009. Mining interesting locations and travel sequences from GPS trajectories. In Proceedings of the 18th international conference on World wide web, 791–800.