Computational Visual Media
https://doi.org/10.1007/s41095-021-0236-6 Vol. 8, No. 2, June 2022, 213–224

Research Article

Trajectory distributions: A new description of movement for trajectory prediction

Pei Lv[1], Hui Wei[1], Tianxin Gu[1], Yuzhen Zhang[1], Xiaoheng Jiang[1], Bing Zhou[1], and Mingliang Xu[1] (✉)

Abstract Trajectory prediction is a fundamental and challenging task for numerous applications, such as autonomous driving and intelligent robots. Current works typically treat pedestrian trajectories as a series of 2D point coordinates. However, in real scenarios, the trajectory often exhibits randomness, and has its own probability distribution. Inspired by this observation and other movement characteristics of pedestrians, we propose a simple and intuitive movement description called a trajectory distribution, which maps the coordinates of the pedestrian trajectory to a 2D Gaussian distribution in space. Based on this novel description, we develop a new trajectory prediction method, which we call the *social probability method*. The method combines trajectory distributions and powerful convolutional recurrent neural networks. Both the input and output of our method are trajectory distributions, which provide the recurrent neural network with sufficient spatial and random information about moving pedestrians. Furthermore, the social probability method extracts spatio-temporal features directly from the new movement description to generate robust and accurate predictions. Experiments on public benchmark datasets show the effectiveness of the proposed method.

Keywords trajectory prediction; convolutional LSTM; trajectory distributions; social probability method

1 Introduction

A pedestrian's trajectory is multimodal, and closely depends on the person's hearing, vision, touch, thoughts, and personality, and is also affected by other factors such as the static environment, dynamic human–human interactions, and planned destinations. Nevertheless, pedestrians still can intuitively predict the future trajectories of others and adjust themselves in advance. For example, when people walk in shopping malls, streets, and stations, they quickly predict the trajectories of others so as to choose their own route at the next moment and avoid collisions. The purpose of trajectory prediction is to enable machines, such as robots, self-driving cars, and intelligent tracking systems, to have the ability to predict future trajectories based on historical trajectories. This is a fundamental but extremely challenging task.

In previous works, researchers mainly focused on the following problems in trajectory prediction: interaction between pedestrians [1–6], interaction between pedestrians and their environment [7–9], and multi-modality [10, 11]. Recently, more and more effort has been made to predict multi-future trajectories [12–16], due to the uncertainty in predicted trajectories. In the real world, a trajectory appears as a probability distribution. When the historical trajectory is known and fixed, a person may have many different future trajectories according to dynamic influences. For example, if the same person walks twice from the same starting location to the same destination, the two trajectories usually differ somewhat. Although the methods above can predict multi-future trajectories, their inputs are still fixed points, and take a trajectory as a two-dimensional

1 School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China. E-mail: P. Lv, ielvpei@zzu.edu.cn; H. Wei, weihui@gs.zzu.edu.cn; T. Gu, txgu@gs.zzu.edu.cn; Y. Zhang, zyzzhang@gs.zzu.edu.cn; X. Jiang, iexhjiang@zzu.edu.cn; B. Zhou, iebzhou@zzu.edu.cn; M. Xu, iexumingliang@zzu.edu.cn (✉).

Manuscript received: 2021-01-21; accepted: 2021-04-20

sequence of coordinates (x_t, y_t). Obviously, each coordinate is fixed, and these separate points fail to represent randomness of the trajectory. Consequently, these methods cannot demonstrate uncertainty in the trajectory caused by inherent randomness.

Other earlier works [1–3, 17, 18] made great progress in modeling the impact of human–human interactions. However, great challenges still exist since most of this work achieves the purpose of modeling pedestrian interactions by combining hidden states. Because the input is a one-dimensional vector, these hidden states are also one-dimensional, so carry little spatial information. The lack of spatial information makes the problem of modeling interactions complicated and difficult.

In order to solve the above problems, we propose the concept of a *trajectory distribution*, which is an intuitive and effective motion description. The trajectory is no longer described by a series of fixed coordinates, but by a probability distribution (see Fig. 1). Specifically, we use a probability density function to map the pedestrian coordinates (x_t, y_t) to two-dimensional Gaussian distributions $G(x_t, y_t)$. Unlike fixed point coordinates, the new description can represent randomness. Moreover, we can conveniently map all pedestrian's trajectories at time t into a single two-dimensional space, with unique advantages in modeling human–human interactions.

Based on the proposed trajectory distribution, we further develop a new method called the *social probability method* for predicting robust and accurate pedestrian trajectories. Firstly, the inputs to our method are trajectory distributions, which enable our forecasting model to fully consider the randomness of the trajectory. Secondly, by adding convolution layers to a recurrent neural network, our forecasting model can learn spatio-temporal features efficiently. We extract location information for pedestrians from the two-dimensional probability space through the convolutional neural network; the two-dimensional probability space contains the locations of all pedestrians. By performing convolution operations on the space, we can extract all pedestrians' location information and easily capture changes in relative locations. These two factors are indispensable for modeling interaction.

Fig. 1 Two pedestrians approaching each other. The current positions of the pedestrians are no longer represented by fixed 2D coordinates, but by Gaussian probability distributions, one for each trajectory.

In summary, the main contributions of this paper are as follows:

1. Trajectory distributions for representing pedestrian trajectories. They capture the inherent randomness of trajectories, facilitating subsequent modeling of their indeterminism.
2. The social probability method, based on trajectory distributions and recurrent neural networks. Convolution operations on trajectory distributions allow better modeling of human–human interactions.
3. Experimental verification of our ideas on ADE and FDE public pedestrian datasets, showing that our approach is competitive with state-of-the-art methods.

The rest of the paper is organized as follows. In Section 2, we first review related work on trajectory prediction. Then we introduce the social probability method in detail in Section 3. In Section 4, we further present and analyse experimental results. Finally, we conclude and discuss future directions in Section 5.

2 Related work

2.1 Multi-outcome trajectory forecasting

In recent years, some researchers have tried to model randomness in trajectory prediction. Gupta et al. [10] solved the trajectory prediction problem using generative adversarial networks (GANs) [19] and considered the fact that pedestrian trajectories may have multiple plausible predictions. SoPhie [8] combined semantic scene segmentation with GANs to model trajectories. Multiverse [11] was a joint model to generate multiple plausible future trajectories,

using multi-scale location encodings and convolutional RNNs over graphs. Simultaneously, Refs. [14, 15, 20] also proposed probability networks to incorporate randomness into vehicle trajectory prediction. However, these works all treat position information as two-dimensional point coordinates for input into the prediction model; doing so cannot completely describe the random behavior of pedestrians. Unlike these works, we take a trajectory distribution transformed from the moving trajectory as input, and generate multiple future trajectories.

2.2 Human–human interaction modeling in trajectory forecasting

In social interaction, researchers have utilized multiple methods of modeling interactions between pedestrians, such as *social force* [2], *social pooling* [1], and *attention* [3]. Methods [21, 22] based on social force use the principle that attractive forces are used to guide people toward their destinations, and repulsive forces are used to avoid collisions, both human–human and human–obstacle. Most social force-based models try to learn the parameters of the social force functions from real-world crowd datasets. However, Alahi et al. [1] showed that attraction and repulsion alone cannot simulate complex crowd interactions. Other approaches [1, 10, 23, 24] used a social pooling layer to allow LSTMs to share hidden states. This novel design can model human interaction efficiently, but complexity increases with crowd density. Thus, methods based on the attention mechanism have emerged [3, 8]. Pedestrians can automatically perceive the importance of certain targets that affect their location in following time steps. Further methods [25–27] simultaneously learn spatial and temporal interaction patterns to capture spatio-temporal correlations efficiently and comprehensively, making their interactive models more suitable for real scenarios. RSBG (recursive social behavior graph) [18] established a group-based social interaction model to explore relationships that are not affected by spatial distance, and a graph convolutional neural network [28] has also been applied to trajectory prediction. In this paper, trajectory distributions are introduced, and the influence of spatial interactions is automatically perceived through convolution operations, which avoids the need to design complex interaction modules. Experimental results show that this method has better interaction performance.

2.3 Sequence prediction model

Sequence prediction uses past sequences to predict future sequences, so it is time series data modeling problem. Convolutional neural networks are very useful in the field of computer vision, but it is difficult to learn the characteristics of time series data using them. Recurrent neural networks are specially suitable for dealing with sequence-related data such as audio, video, and text. Recurrent neural networks and their derivatives LSTMs [29] and GRUs (gated recurrent units) [30] have proved their effectiveness in many fields, such as machine translation [31], text generation [32, 33], speech recognition [34–36], and traffic flow prediction [37]. Some researchers have combined convolutional neural networks with recurrent neural networks for novel applications, such as image captioning [33, 38, 39] and video understanding [40, 41]. In order to learn spatio-temporal features simultaneously, Shi et al. [42] added convolution layers to a recurrent neural network. Their ConvLSTM model not only learns temporal relationships, but also extracts spatial features using the convolution layer. We take advantage of ConvLSTM to obtain spatio-temporal features and directly model interaction between pedestrians.

3 Approach

In this section, we first present our new pedestrian motion description, the trajectory distribution, which solves the problem of modeling multiple trajectories from the data description level, and then we propose a prediction model based on trajectory distributions to describe human–human interactions conveniently.

3.1 Problem definition

Our goal is to predict the future trajectories of the pedestrians in a scene. The input is the historical location information for each pedestrian in the scene and the output is trajectory information for all pedestrians in the future. We define the historical trajectory distribution of the pedestrian as $\boldsymbol{X} = \boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$. The predicted future trajectory distribution is denoted $\widehat{\boldsymbol{Y}} = \widehat{\boldsymbol{Y}}_1, \ldots, \widehat{\boldsymbol{Y}}_n$, where n is the number of pedestrians. The input trajectory of pedestrian i is defined as $X_i \sim N(x_t^i, y_t^i)$ for time

steps $t = 1, \ldots, t_{\text{obs}}$ and the future trajectory can be defined similarly as $Y_i \sim N(x_t^i, y_t^i)$ for time steps $t = t_{\text{obs}+1}, t_{\text{obs}+2}, \ldots, t_{\text{pred}}$, where N represents a Gaussian distribution. The prediction is denoted $\widehat{Y_i}$ and the ground truth is denoted Y_i.

3.2 Trajectory distribution

3.2.1 Mathematical definition

Supposing the feasible area for the pedestrians is Ω, we represent the location of a single pedestrian at time t as a probability distribution on Ω. We use a two-dimensional Gaussian distribution, which can well characterize the location of a trajectory. The location distribution at time t has the highest probability density at the center position (x_t, y_t). It means that the location does not have to be at this fixed position, but also has a probability of being located in some other area; the further away from the central location, the smaller the probability density becomes. We suppose that (x_t, y_t) follows a two-dimensional Gaussian distribution with parameters $(\mu_1, \mu_2, \sigma_1, \sigma_2, \rho)$:

$$f(x_t, y_t) = \left(2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}\right)^{-1} \exp\left[-\frac{1}{2(1-\rho^2)} \cdot \left(\frac{(x_t-\mu_1)^2}{{\sigma_1}^2} - \frac{2\rho(x_t-\mu_1)(y_t-\mu_2)}{\sigma_1\sigma_2}\right) + \frac{(y_t-\mu_2)^2}{{\sigma_2}^2}\right] \quad (1)$$

where μ_1 and μ_2 are the mean values of (x_t, y_t) respectively, σ_1 and σ_2 are the variances of (x_t, y_t), and ρ is the correlation coefficient of x_t and y_t. μ_1 is set to x_t and μ_2 is set to y_t. σ_1 and σ_2 are set to 0.3 according to experience, and ρ is 0.

Using this data structure to represent trajectories, we may successfully retain the randomness of trajectories. In two-dimensional space, a pedestrian trajectory is no longer a single point at time t, but a probability distribution, as shown in Fig. 2(a).

3.2.2 Integrating neighbor information

At time t, we denote the trajectory distribution of pedestrian i by p_t^i. However, the scene at time t contains multiple pedestrians. Neighboring pedestrians have great influence on the movement decisions of each subject pedestrian. In order to enable the model to predict future trajectories based on the locations of surrounding pedestrians, we need to integrate the trajectory distributions of all pedestrians at time t into the same two-dimensional probability space. The trajectory distribution at time t is denoted p_t. In two-dimensional space, we integrate p_t^i into p_t using the $\max(\cdot)$ function. Specifically, for the corresponding position in the trajectory distribution, we take the larger value as the consolidated value, as follows.

$$p_t = \max(p_t, p_t^i), \quad i = 1, \ldots, n \quad (2)$$

where n is the number of pedestrians at time t. In order to distinguish the current predicted pedestrian from the other surrounding pedestrians, we set their σ values to 0.1 and 0.3 respectively. A comparison using different σ is shown in Fig. 2(b).

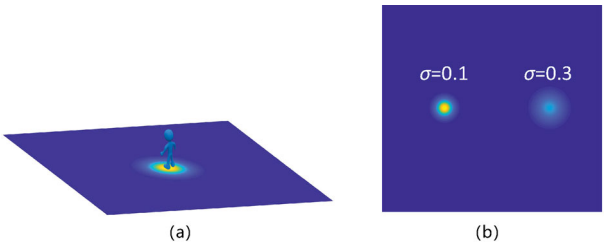

Fig. 2 A trajectory distribution describes the location of a pedestrian as a probability distribution. (a) Distribution for a certain trajectory. (b) Areas of influence for different values of σ of the probability distribution.

3.3 Convolutional LSTM

Due to its unique structure, the long and short-term memory network (LSTM) has great advantages in processing time sequence data. Moreover, Shi et al. [42] proposed a variant of LSTM, which added a convolutional layer to the LSTM module, calling ConvLSTM, and demonstrated that this model can learn spatio-temporal information through experiments. Specifically, the main operations are as follows:

$$i_t = \sigma(W_{xi}X_t + W_{hi}h_{t-1} + W_{ci} \circ c_{t-1} + b_i) \quad (3)$$

$$f_t = \sigma(W_{xf}X_t + W_{hf}h_{t-1} + W_{cf} \circ c_{t-1} + b_f) \quad (4)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tanh(W_{xc}X_t + W_{hc}h_{t-1} + b_c) \quad (5)$$

$$o_t = \sigma(W_{xo}X_t + W_{ho}h_{t-1} + W_{co} \circ c_t + b_o) \quad (6)$$

$$h_t = o_t \circ \tanh(c_t) \quad (7)$$

where X_t is the input at time t, and h_t and c_t are hidden state and cell state, respectively. i_t, f_t, o_t are the gates of the ConvLSTM. They are all three-dimensional tensors whose last two dimensions are spatial dimensions (width, height). W is a weight matrix. "$\circ$" denotes the Hadamard product. At time

t, X_t provides input to the module for calculation only when the input gate is activated. Similarly, the past cell state c_{t-1} is forgotten when the forget gate f_t is activated and the current cell state c_t is transfered when the output gate o_t is open.

ConvLSTM uses the current input and past states to determine future states; the current input includes not only temporal features, but also spatial features. The temporal features can be learned through the gate structure mentioned above, and the spatial features can be extracted through the convolutional layer embedded in the module. Essentially, trajectory prediction can be regarded as a spatio-temporal sequence generation problem. Therefore, applying ConvLSTM to solve it, we can model the temporal characteristics of the trajectory while also considering spatial interactions between different trajectories.

3.4 Social probability

As we show in Fig. 3, the social probability method is a trajectory prediction method based on trajectory distributions. Firstly, we map the position information of all pedestrians at time t into trajectory distributions. Then, the ConvLSTM module takes two-dimensional trajectory distributions as input and outputs predictive trajectory distributions at future time $t+1$. The coordinates of trajectory points can be obtained by sampling the outputs.

3.4.1 Probability-based prediction

The input to the ConvLSTM needs to be two-dimensional tensors. As discussed in Section 3.2, our trajectory distribution is a probability distribution in two-dimensional space. Therefore, it is suitable for input to the ConvLSTM model. Moreover, trajectory distributions are essentially probability density distributions. The value of the trajectory distribution indicates the level of probability density. Modeling trajectory distributions directly makes our method a probability-based forecasting method. Our method not only predicts multiple future trajectories, but the input historical trajectory is also multimodal, unlike previous methods. The problem of modeling multimodal features is solved at the data level.

3.4.2 Human–human interaction modeling

The input to our model comprises the trajectory distributions of all pedestrians at time t, integrated into one two-dimensional space, so modeling human–human interactions is direct and expedient. As illustrated in Fig. 4, after trajectory distributions are input into the model, the convolutional layer extracts features in the two-dimensional trajectory distribution to obtain the hidden state, which is the feature vector in the RNN-based model. Since the convolution kernel slides across the entire two-dimensional space like a sliding window, hidden states contain location information for each pedestrian. Thus, due to the convolution operation, the model not only considers the density value at the current position, but also the density value at surrounding positions when predicting the probability density value in the future. Therefore, our model considers the locations of all pedestrians at time t, considering human–human interactions without complex interaction modules.

3.4.3 Loss function

We empirically choose the loss function to train our model by following previous works [43, 44]. Since

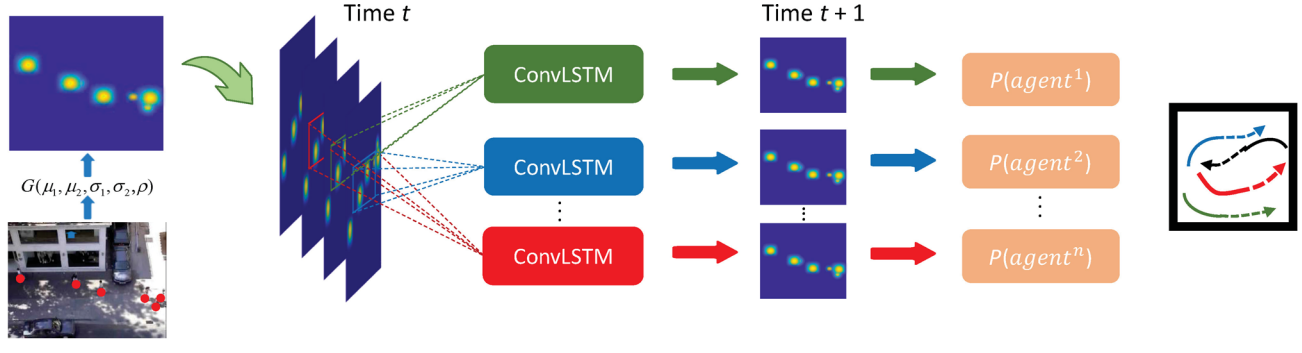

Fig. 3 Overview of the social probability method. We use a separate ConvLSTM network for each trajectory in the scene. Inputs and outputs of the model are both trajectory distributions. The trajectory distribution is mapped from fixed point coordinates; a single trajectory distribution at time t contains all pedestrians' trajectory information. The ConvLSTM network consists of a convolutional layer and gate modules, and has the ability to learn spatio-temporal features. In the prediction stage, the output of our model is again in the form of trajectory distributions, and trajectory coordinates can be obtained by sampling them.

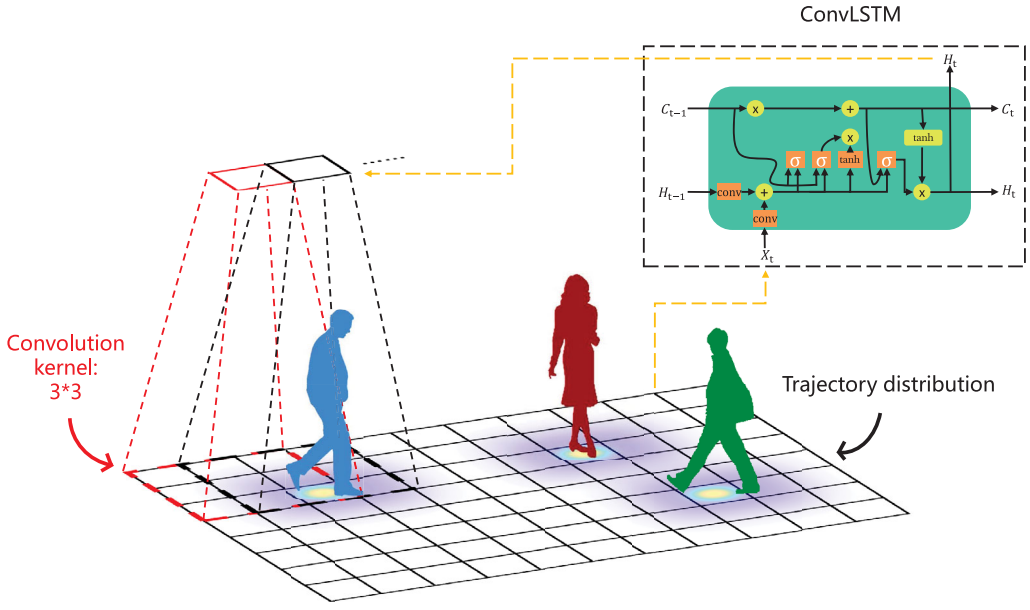

Fig. 4 The convolution layer added to LSTM gives the model the ability to extract spatial features. Convolution operations on the trajectory distribution enable our method to model human–human interactions more intuitively and efficiently.

our model focuses on the specific probability density value, rather than some high-dimensional features, such as style, graphics, or objects, we use an $L2$ loss function to encourage our model to generate accurate probability density distributions:

$$\mathcal{L}_{L2}(\hat{Y}_t^i, Y_t^i) = ||\hat{Y}_t^i - Y_t^i||^2 \tag{8}$$

where $\hat{Y}_t^i$ and Y_t^i are the predicted and ground truth trajectory distributions for person i at time t respectively.

4 Experiments

In this section, we present experimental results using five public datasets, and compare our method to state-of-the-art methods, as well as analyzing the performance of our method.

4.1 Datasets

We validated the proposed model on the public ETH [22] and UCY [45] datasets, which are the widely used benchmarks in the field of trajectory prediction. Most state-of-the-art methods have been evaluated on these datasets. They contain a total of 1536 labeled pedestrians in 4 different scenes. These datasets are based on binocular vision for the research of pedestrian trajectory tracking and prediction. There are altogether 5 sub datasets: ETH contains ETH and HOTEL subsets, while UCY has three subsets: ZARA1, ZARA2, and UNIV. Following previous works, we observe the historical trajectory for the past 8 time steps (3.2 s) and predict the future trajectory for the next 12 time steps (4.8 s).

4.2 Evaluation metrics and methods

Following previous works [1], we use two evaluation metrics:

- *Average displacement error (ADE):* The average Euclidean distance between the predicted trajectories and the true trajectories at each prediction time step:

$$\text{ADE} = \frac{\sum_{i\in\mathbb{Z}} \sum_{t=T_{\text{obs}}+1}^{T_{\text{pre}}} \sqrt{((\hat{x}_t^i, \hat{y}_t^i) - (x_t^i, y_t^i))^2}}{\mathbb{Z} * T_{\text{pre}}}$$

- *Final displacement error (FDE):* The Euclidean distance between the predicted destination and the ground truth destination at the last prediction time step:

$$\text{FDE} = \frac{\sum_{i\in\mathbb{Z}} \sqrt{((\hat{x}_{T_{\text{pre}}}^i, \hat{y}_{T_{\text{pre}}}^i) - (x_{T_{\text{pre}}}^i, y_{T_{\text{pre}}}^i))^2}}{\mathbb{Z}}$$

In the above $(\hat{x}_t^i, \hat{y}_t^i)$ and (x_t^i, y_t^i) are the predicted and ground truth coordinates for pedestrian i at time t respectively, and $\mathbb{Z}$ is the total number of pedestrians in the test set.

We use a leave-one-out approach to evaluate the performance of the model. Four sets are used as the training set and verification set, and the remaining one is used as the test set.

4.3 Implementation details

Five layers are used in the ConvLSTM model and the hidden state channel size in each layer is 128, 64, 64, 32, 32, respectively. The kernel size of the convolutional layer is 3×3 and the padding is 1. We train our model using Adam [46] with an initial learning rate of 0.001. The sizes of the trajectory distribution and the hidden state of our model are both 100×100. In the prediction stage, the variance of the current pedestrian to be predicted is set to 0.1, and the other pedestrians are set to 0.3. In the testing stage, we sample 20 times from the trajectory distribution predicted by the model, and select the best prediction in terms of Euclidean distance for quantitative estimation.

4.4 Comparison with other methods

As shown in Table 1, we choose the following methods for comparison:

1. Linear: A linear regression model which predicts the trajectory by minimizing the least square error.
2. Plain-LSTM: Use the LSTM model to predict the future trajectory. This method only considers its own historical trajectory and does not consider any other factors.
3. Social-LSTM [1]: A social-pooling layer is added to the LSTM, so that the model has the ability to model human–human interactions.
4. Social-GAN [10]: A trajectory prediction model trained with GAN architecture is designed to improve existing models in terms of rationality, diversity, and prediction speed. The model pays attention to the feasibility of generating trajectory predictions using social rules.
5. Social-GAN-P [10]: As Social-GAN, but without the pooling mechanism.
6. SoPhie [8]: An interpretable framework based on GAN for trajectory prediction. It uses two information sources, the historical trajectories of all pedestrians in a scene and the scene contextual information from the scene image.
7. RSBG [18]: A group-based social interaction model to explore pedestrian relationships that are not affected by spatial distance. A graph convolutional neural network is applied to trajectory prediction in this model.
8. NEXT [7]: An end-to-end multi-task learning system that uses pedestrian behavior information and its surrounding scene environment to predict trajectory. This method uses behavior information for the first time to improve the accuracy of trajectory prediction.

4.5 Quantitative analysis

Table 1 presents average displacement error and final displacement error for our method and existing methods, given the task of predicting 12 future time steps from 8 historical time steps. We follow comparative works in choosing the best prediction among multiple samples for quantitative analysis. It can be seen that the linear model usually performs worst, because it is only suitable for predicting straight trajectories, and is insensitive to pedestrian interaction. Social-LSTM and Social-GAN perform better than the linear method since they can handle interactions between pedestrians via the corresponding interaction module. We can see that our method outperforms all others in terms of FDE for the ETH and UNIV datasets, avoiding more potential future collisions. Although the performance of our method is not the best on other datasets, it is still very competitive and significantly superior to

Table 1 Results using different methods on the ETH, HOTEL (from ETH), UNIV, ZARA1, and ZARA2 (from UCY) datasets. Error metrics used are ADE and FDE for the task of predicting 12 future time steps

Method	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
Linear	1.33 / 2.94	0.39 / 0.72	0.82 / 1.59	0.62 / 1.21	0.77 / 1.48	0.79 / 1.59
Plain-LSTM	1.09 / 2.14	0.86 / 1.91	0.61 / 1.31	0.41 / 0.88	0.52 / 1.11	0.70 / 1.52
Social-LSTM [1]	1.09 / 2.35	0.79 / 1.76	0.67 / 1.40	0.47 / 1.00	0.56 / 1.17	0.72 / 1.54
Social-GAN [10]	0.81 / 1.52	0.72 / 1.61	0.60 / 1.26	0.34 / 0.69	0.42 / 0.84	0.58 / 1.18
Social-GAN-P [10]	0.87 / 1.62	0.67 / 1.37	0.76 / 1.52	0.35 / 0.68	0.42 / 0.84	0.61 / 1.21
SoPhie [8]	**0.70** / 1.43	0.76 / 1.67	**0.54** / 1.24	**0.30 / 0.63**	0.38 / 0.78	0.54 / 1.15
RSBG [18]	0.80 / 1.53	0.33 / 0.64	0.59 / 1.25	0.40 / 0.86	**0.30 / 0.65**	0.48 / 0.99
NEXT [7]	0.73 / 1.65	**0.30 / 0.59**	0.60 / 1.27	0.38 / 0.81	0.31 / 0.68	**0.46** / 1.00
Ours	0.74 / **1.22**	0.49 / 0.85	0.63 / **1.23**	0.42 / 0.78	0.38 / 0.70	0.53 / **0.95**

the linear model except in the case of the HOTEL dataset with few pedestrians. This demonstrates that our method implicitly models interaction without complex interaction modules. In addition, our method performs better in terms of FDE than ADE, especially for the ETH dataset. This reflects that our method has more advantages in terms of predicting destinations.

4.6 Qualitative analysis

Figure 5 shows successful and unsuccessful examples for each dataset. Blue trajectories are the predicted trajectories for 12 future time steps given the green observed trajectories over the past 8 time steps; red trajectories are the ground truth trajectories. These examples show that our model is able to correctly predict future paths and has the ability to model human–human interaction. The successful samples show that the model can avoid obstacles in advance when interacting with others. Furthermore, our method is also suitable for crowded scenes, when multiple people are walking forward to the same or in different directions, avoiding each other by following others or passing between them.

The last row shows some unsuccessful examples. These examples have large gaps in predictions, or go in the wrong direction. By analyzing the source videos, we found that such cases were generated when pedestrians stopped walking or turned suddenly, the main reason being the unpredictability of pedestrian intent. Another reason is that when pedestrians interact with their physical surroundings, the model does not handle scene information. Integrating information about the scene is a direction for our future research.

4.7 Ablation study

4.7.1 Attention mechanism

In our experiments, we tried using a spatial attention mechanism [47] to improve the prediction accuracy of our model. In the two-dimensional trajectory distribution space, an attention module was applied to capture which locations have more influence. However, we found that the attention mechanism did not improve our results as expected—see Table 2. The reason may be that the trajectory distribution has already played a role in providing attention. The probability density of each spatial position represents the importance of the location, namely the weight value in the attention mechanism.

Table 2 Results with and without an attention mechanism, using the ETH dataset without data augmentation

Method	ADE	FDE
Our method with attention	0.99	1.73
Our method without attention	0.91	1.61

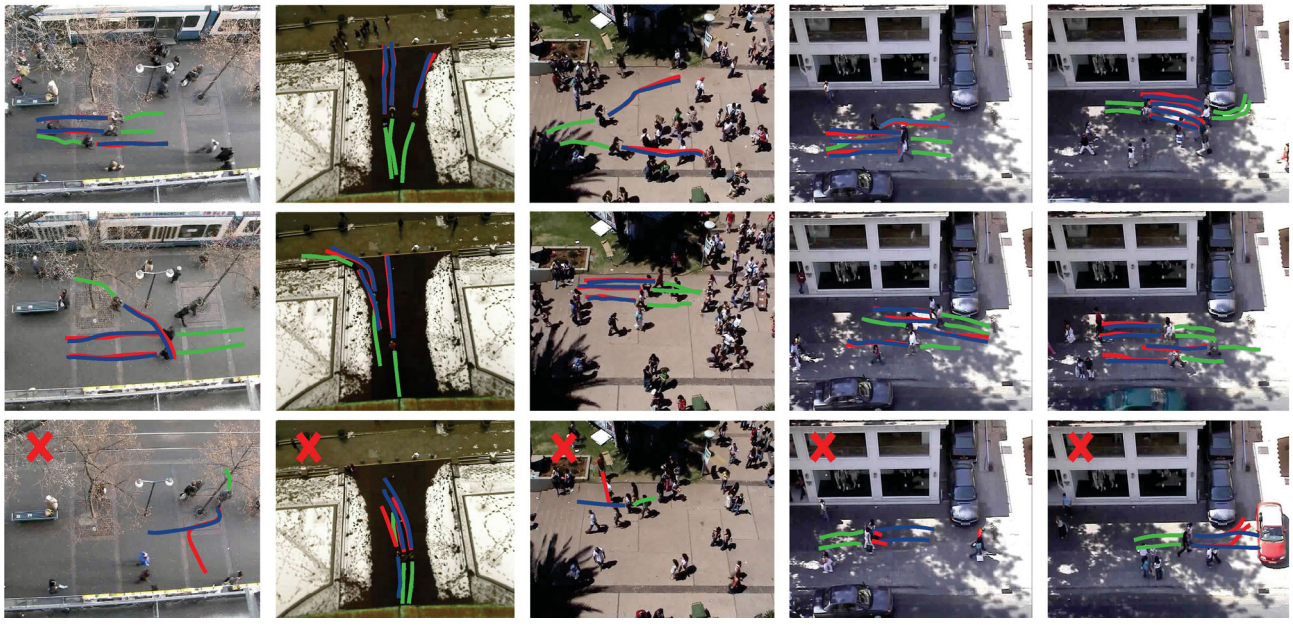

Fig. 5 Example results of our method. Given observed trajectories for the past 8 time steps (green), our method predicts the trajectories for 12 future time steps (blue), and compares them with the ground truth trajectories (red). Left to right: results for ETH, HOTEL, UNIV, ZARA1, and ZARA2. Above, center: representative examples where our method successfully predicts trajectories with small errors. Below: failure cases.

4.7.2 Integration of trajectory distributions

As explained in Section 3.2, we integrate the trajectory distributions of all pedestrians at time t into a single two-dimensional probability space. Here, we omit the other trajectory distributions to verify the ability of our model to capture interaction. Thus, when predicting the trajectory of person i, the trajectory distribution only contains that person's own trajectory information, and trajectory information of people around is omitted. We conducted experiments using the ETH dataset—see Table 3. The method with integration has clearly better ADE and FDE, showing that integrating trajectory distributions provides the ability to model human–human interaction.

4.7.3 Size of trajectory distribution

The trajectory distribution is two-dimensional, so a suitable size must be determined. We set up a comparison, using sizes of 80×80, 100×100, 120×120, 150×150, 170×170, and 200×200. Results are shown in Fig. 6. The predicted result is best when the size is 100×100. Too large or too small a value decreases prediction accuracy. We sampled from the ground truth and found that as the size increases, the sampling error also increases. Sampling error may be the reason for the decrease in prediction accuracy, while as the size decreases, the trajectory distribution is incapable of modeling large enough amounts of data.

Table 3 Results with and without integration of neighbor's trajectory distributions, using the ETH dataset

Method	ADE	FDE
Full algorithm	0.74	1.22
Our method without integration	0.86	1.64

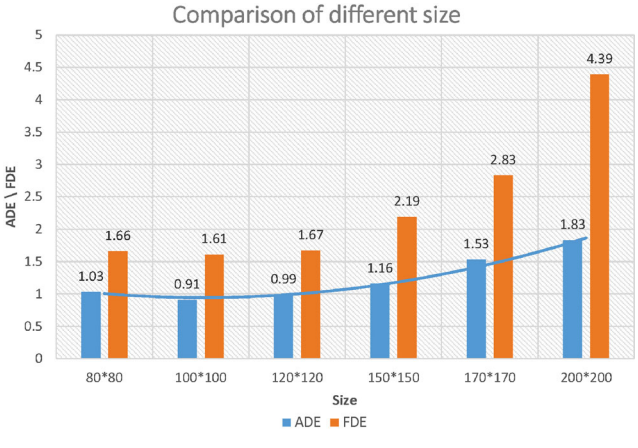

Fig. 6 Size of trajectory distribution. The blue line shows the trend of ADE. Experiments were conducted on the ETH dataset without data augmentation.

5 Conclusions

In this paper, we have proposed the concept of trajectory distributions, with advantages in representing the randomness of trajectories, and explored a new trajectory prediction method based on it. To encode social interaction features, we introduced ConvLSTM, a sequence to sequence prediction model with the ability to model spatio-temporal information. Experiments on public datasets show the effectiveness of our method. Although it is not best on all datasets, our method is simple and has great potential. Our current work does not incorporate the physical environment, but it is obvious that adding such information to our model is straightforward and convenient, and is the direction of our future work.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61772474, 61802351, 61822701, and 61872324, and in part by the Program for Science and Technology Innovation Talents in Universities of Henan Province under Grant No. 20HASTIT021. We also thank the anonymous reviewers for their valuable comments and suggestions that helped improve the quality of this manuscript

References

[1] Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Li, F. F.; Savarese, S. Social LSTM: Human trajectory prediction in crowded spaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 961–971, 2016.

[2] Helbing, D.; Molnár, P. Social force model for pedestrian dynamics. *Physical Review E* Vol. 51, No. 5, 4282, 1995.

[3] Vemula, A.; Muelling, K.; Oh, J. Social attention: Modeling attention in human crowds. In: Proceedings of the IEEE International Conference on Robotics and Automation, 4601–4607, 2018.

[4] Yi, S.; Li, H. S.; Wang, X. G. Pedestrian behavior understanding and prediction with deep neural

networks. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9905*. Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 263–279, 2016.

[5] Zhang, P.; Ouyang, W. L.; Zhang, P. F.; Xue, J. R.; Zheng, N. N. SR-LSTM: State refinement for LSTM towards pedestrian trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 12077–12086, 2019.

[6] Xu, M. L.; Li, C. X.; Lv, P.; Lin, N.; Hou, R.; Zhou, B. An efficient method of crowd aggregation computation in public areas. *IEEE Transactions on Circuits and Systems for Video Technology* Vol. 28, No. 10, 2814–2825, 2018.

[7] Liang, J. W.; Jiang, L.; Niebles, J. C.; Hauptmann, A.; Fei-Fei, L. Peeking into the future: Predicting future person activities and locations in videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2960–2963, 2019.

[8] 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/CVF Conference on Computer Vision and Pattern Recognition, 1349–1358, 2019.

[9] Wong, S. K.; Wang, Y. S.; Tang, P. K.; Tsai, T. Y. Optimized evacuation route based on crowd simulation. *Computational Visual Media* Vol. 3, No. 3, 243–261, 2017.

[10] Gupta, A.; Johnson, J.; Li, F. F.; Savarese, S.; Alahi, A. Social GAN: Socially acceptable trajectories with generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2255–2264, 2018.

[11] Liang, J. W.; Jiang, L.; Murphy, K.; Yu, T.; Hauptmann, A. The garden of forking paths: Towards multi-future trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10505–10515, 2020.

[12] Chai, Y. N.; Sapp, B.; Bansal, M.; Anguelov, D. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. *arXiv preprint* arXiv:1910.05449, 2019.

[13] Li, Y. K. Which way are you going? Imitative decision learning for path forecasting in dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 294–303, 2019.

[14] Makansi, O.; Ilg, E.; Çiçek, Ö.; Brox, T. Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7137–7146, 2019.

[15] Tang, Y. C.; Salakhutdinov, R. Multiple futures prediction. In: Proceedings of the 33rd Conference on Neural Information Processing Systems, 15424–15434, 2019.

[16] Xue, H.; Huynh, D. Q.; Reynolds, M. SS-LSTM: A hierarchical LSTM model for pedestrian trajectory prediction. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 1186–1194, 2018.

[17] Huang, Y. F.; Bi, H. K.; Li, Z. X.; Mao, T. L.; Wang, Z. Q. STGAT: Modeling spatial-temporal interactions for human trajectory prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 6271–6280, 2019.

[18] Sun, J. H.; Jiang, Q. H.; Lu, C. W. Recursive social behavior graph for trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 657–666, 2020.

[19] Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In: Proceedings of the 27th Conference on Neural Information Processing Systems, 2672–2680, 2014.

[20] Thiede, L.; Brahma, P. Analyzing the variety loss in the context of probabilistic trajectory prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 9953–9962, 2019.

[21] Mehran, R.; Oyama, A.; Shah, M. Abnormal crowd behavior detection using social force model. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 935–942, 2009.

[22] Pellegrini, S.; Ess, A.; Schindler, K.; van Gool, L. You'll never walk alone: Modeling social behavior for multi-target tracking. In: Proceedings of the IEEE 12th International Conference on Computer Vision, 261–268, 2009.

[23] Su, H.; Zhu, J.; Dong, Y.; Zhang, B. Forecast the plausible paths in crowd scenes. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence, 2772–2778, 2017.

[24] Xu, Y. Y.; Piao, Z. X.; Gao, S. H. Encoding crowd interaction with deep neural network for

pedestrian trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5275–5284, 2018.

[25] Tang, Q. C.; Yang, M. N.; Yang, Y. ST-LSTM: A deep learning approach combined spatio-temporal features for short-term forecast in rail transit. *Journal of Advanced Transportation* Vol. 2019, Article ID 8392592, 2019.

[26] Zheng, C. P.; Fan, X. L.; Wang, C.; Qi, J. Z. GMAN: A graph multi-attention network for traffic prediction. *Proceedings of the AAAI Conference on Artificial Intelligence* Vol. 34, No. 1, 1234–1241, 2020.

[27] Yu, B.; Yin, H. T.; Zhu, Z. X. Spatiotemporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint* arXiv:1709.04875, 2017.

[28] Henaff, M.; Bruna, J.; LeCun, Y. Deep convolutional networks on graph-structured data. *arXiv preprint* arXiv:1506.05163, 2015.

[29] Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Computation* Vol. 9, No. 8, 1735–1780, 1997.

[30] Cho, K.; van Merrienboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. *arXiv preprint* arXiv:1406.1078, 2014.

[31] Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv preprint* arXiv:1409.0473, 2014.

[32] Karpathy, A.; Joulin, A.; Fei-Fei, L. Deep fragment embeddings for bidirectional image sentence mapping. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, 1889–1897, 2014.

[33] Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. Show and tell: A neural image caption generator. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3156–3164, 2015.

[34] Chorowski, J.; Bahdanau, D.; Cho, K.; Bengio, Y. End-to-end continuous speech recognition using attention-based recurrent NN: First results. *arXiv preprint* arXiv:1412.1602, 2014.

[35] Chung, J.; Kastner, K.; Dinh, L.; Goel, K.; Courville, A.; Bengio, Y. A recurrent latent variable model for sequential data. *arXiv preprint* arXiv:1506.02216, 2015.

[36] Graves, A.; Jaitly, N. Towards end-to-end speech recognition with recurrent neural networks. In: Proceedings of the 31st International Conference on Machine Learning, 1764–1772, 2014.

[37] Yang, B. L.; Sun, S. L.; Li, J. Y.; Lin, X. X.; Tian, Y. Traffic flow prediction using LSTM with feature enhancement. *Neurocomputing* Vol. 332, 320–327, 2019.

[38] Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In: Proceedings of the 32nd International Conference on Machine Learning, 2048–2057, 2015.

[39] You, Q. Z.; Jin, H. L.; Wang, Z. W.; Fang, C.; Luo, J. B. Image captioning with semantic attention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4651–4659, 2016.

[40] Donahue, J.; Hendricks, L. A.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Darrell, T.; Saenko, K. Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2625–2634, 2015.

[41] Srivastava, N.; Mansimov, E.; Salakhudinov, R. Unsupervised learning of video representations using LSTMs. In: Proceedings of the 32nd International Conference on Machine Learning, 843–852, 2015.

[42] Shi, X.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-K.; Woo, W.-C. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, 802–810, 2015.

[43] Feng, Z. H.; Kittler, J.; Awais, M.; Huber, P.; Wu, X. J. Wing loss for robust facial landmark localisation with convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2235–2245, 2018.

[44] Zhao, H.; Gallo, O.; Frosio, I.; Kautz, J. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging* Vol. 3, No. 1, 47–57, 2017.

[45] Lerner, A.; Chrysanthou, Y.; Lischinski, D. Crowds by example. *Computer Graphics Forum* Vol. 26, No. 3, 655–664, 2007.

[46] Kingma, D. P.; Ba, J. L. ADAM: A method for stochastic optimization. *arXiv preprint* arXiv:1412.6980, 2014.

[47] Woo, S.; Park, J.; Lee, J. Y.; Kweon, I. S. CBAM: Convolutional block attention module. In: *Computer Vision–ECCV 2018. Lecture Notes in Computer Science, Vol. 11211*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 3–19, 2018.

Pei Lv received his Ph.D. degree from the State Key Laboratory of CAD&CG, Zhejiang University, China, in 2013. He is an associate professor with the School of Information Engineering, Zhengzhou University, China. His research interests include computer vision and computer graphics. He has authored more than 30 journals and conference papers in the above areas.

Hui Wei received his B.S. degree from the Software Engineering Department, Henan Agricultural University, China, in 2018. He is currently a master student in the School of Information Engineering of Zhengzhou University. His research interests include computer vision and trajectory prediction.

Tianxin Gu received her B.S. degree from the Network Engineering Department, Henan Polytechnic University, China, in 2018. She is currently a master student in the School of Information Engineering of Zhengzhou University. His research interests include computer vision and trajectory prediction.

Yuzhen Zhang received her M.S. degree from the Software Engineering Department, Henan Polytechnic University, China, in 2019. She is currently a doctoral student in the School of Information Engineering of Zhengzhou University. Her research interests include computer vision and trajectory prediction.

Xiaoheng Jiang received his B.S., M.S., and Ph.D. degrees in electronic information engineering from Tianjin University, China, in 2010, 2013, and 2017, respectively. He is currently a lecturer with the School of Information Engineering, Zhengzhou University. His research interests include computer vision and deep learning.

Bing Zhou received his B.S. and M.S. degrees in computer science from Xi'an Jiao Tong University, China, in 1986 and 1989, respectively, and his Ph.D. degree in computer science from Beihang University, China, in 2003. He is currently a professor with the School of Information Engineering, Zhengzhou University, China. His research interests include video processing and understanding, surveillance, computer vision, and multimedia applications.

Mingliang Xu received his Ph.D. degree in computer science and technology from the State Key Laboratory of CAD&CG, Zhejiang University in 2012. He is a full professor with the School of Information Engineering, Zhengzhou University, where he is currently the director of the Center for Interdisciplinary Information Science Research and the vice general secretary of ACM SIGAI China. His research interests include computer graphics, multimedia, and artificial intelligence. He has authored more than 60 journals and conference papers in the above areas.