electronics

MDPI

Article

A Pedestrian Trajectory Prediction Method for Generative Adversarial Networks Based on Scene Constraints

Zhongli Ma [1], Ruojin An [1], Jiajia Liu [1,*], Yuyong Cui [2], Jun Qi [3], Yunlong Teng [4], Zhijun Sun [5] and Juguang Li [6] and Guoliang Zhang [1]

[1] College of Automation, Chengdu University of Information Technology, Chengdu 610103, China; mazl@cuit.edu.cn (Z.M.); an1587601@163.com (R.A.); zhgl@cuit.edu.cn (G.Z.)
[2] Southwest Institute of Technical Physics, Chengdu 610041, China; charleycui@gmail.com
[3] College of Communication Engineering, Chengdu University of Information Technology, Chengdu 610225, China; qijun@cuit.edu.cn
[4] College of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; ylteng@uestc.edu.cn
[5] Nuclear Power Institute of China, Chengdu 610005, China; sunzhijun999@gmail.com
[6] Chengdu Emfuture Technology Co., Ltd., Chengdu 611731, China; Robert.Lee@emfuture.com
* Correspondence: liujj@cuit.edu.cn

Abstract: Pedestrian trajectory prediction is one of the most important topics to be researched for unmanned driving and intelligent mobile robots to perform perceptual interaction with the environment. To solve the problem of the SGAN (social generative adversarial networks) model lacking an understanding of pedestrian interaction and scene constraints, this paper proposes a trajectory prediction method based on a scenario-constrained generative adversarial network. Firstly, a self-attention mechanism is added, which can integrate information at every moment. Secondly, mutual information is introduced to enhance the influence of latent code on the predicted trajectory. Finally, a new social pool is introduced into the original trajectory prediction model, and a scene edge extraction module is added to ensure the final output path of the model is within the passable area in line with the physical scene, which greatly improves the accuracy of trajectory prediction. Based on the CARLA (CAR Learning to Act) simulation platform, the improved model was tested on the public dataset and the self-built dataset. The experimental results showed that the average moving deviation was reduced by 26.4% and the final offset was reduced by 23.8%, which proved that the improved model could better solve the uncertainty of pedestrian turning decisions. The accuracy and stability of pedestrian trajectory prediction are improved while maintaining multiple modes.

Keywords: scene constraint; pedestrian trajectory prediction; generative adversarial networks; self-attention mechanism; CARLA simulation

Citation: Ma, Z.; An, R.; Liu, J.; Cui, Y.; Qi, J.; Teng, Y.; Sun, Z.; Li, J.; Zhang, G. A Pedestrian Trajectory Prediction Method for Generative Adversarial Networks Based on Scene Constraints. *Electronics* **2024**, *13*, 628. https://doi.org/10.3390/electronics13030628

Academic Editor: Felipe Jiménez

Received: 28 December 2023
Revised: 22 January 2024
Accepted: 30 January 2024
Published: 2 February 2024

1. Introduction

Pedestrian trajectory prediction uses the trajectory information of pedestrians in the past to predict the movement trajectory that pedestrians may choose in the future, which is an important part of the environment perception module of unmanned driving technology and intelligent mobile robots [1]. In the field of computer vision, pedestrian trajectory prediction based on visual information has become a research hotspot. Traditional pedestrian trajectory prediction methods usually focus on the establishment of mathematical–statistical models, such as trajectory prediction models based on Social Force (SF) [2] and Social Aware (SA) [3]. The traditional methods above rely on the interaction rules of manually specified pedestrians, resulting in poor adaptability to different scenarios, and simple kinematic models are not suitable for long-term prediction. The pedestrian path prediction method based on deep learning has been proposed more recently, but it is suitable for long-term prediction and has been widely used by researchers. Sumpter et al. [4] used image-based

trajectory sequences as input to predict pedestrian movement trajectory through neural networks. Alahi et al. [5] proposed an LSTM (Long Short-Term Memory) network to model trajectory prediction and achieved good results. Bartoli et al. [6] conducted a more in-depth study on the above methods and proposed a Social-LSTM model, which added static obstacle information in the environment to increase the model's understanding of scene constraints and improve the accuracy of prediction. Varshneya et al. [7] proposed an end-to-end prediction model in which the pooled structure could extract the influence features of the surrounding pedestrians on the target pedestrians. Raipuria et al. [8] applied the above method to the highway scene and achieved good results. Gupta et al. [9] used generative adversarial networks (GAN) for pedestrian trajectory prediction for the first time and proposed Social-GAN (SGAN), which achieved higher accuracy in pedestrian trajectory prediction and made the generated predicted trajectory no longer single. Based on the above methods, Amirian et al. [10] added an attention mechanism to the network to screen the miscellaneous pedestrian interaction information, reducing the computational load of the network and improving the prediction efficiency. PEI Zhao et al. [11] proposed a transformer generative adversarial network (GAN) algorithm, which combines dynamic scene information with pedestrian social interaction information. The convolution neural network model of the dynamic scene extraction module is utilized to extract the dynamic scene information features of the target pedestrian, which improves the average error and the final displacement error. Liming Lao et al. [12] proposed a novel prediction model termed the social and spatial attentive generative adversarial network (SSA-GAN). The SSA-GAN framework utilizes a generative approach, where the generator employs social attention mechanisms to accurately model social interactions among pedestrians. At the same time, the model uses comprehensive motion characteristics as query vectors, which significantly enhances the prediction performance. Li et al. [13] proposed a neural network model with a memory function for the pedestrian and environmental information obtained by the driverless sensing system. Brebisson et al. [14] proposed a neural network based on bidirectional recursion, which converts the data obtained by the sensor into a sequence and completes the position prediction of the target [15]. Kuchar J.K et al. [16] elaborated the basic functional framework of CDR for classifying models, and commented on the current system design process. Migliaccio G et al. [17] used a moving ellipsoid to represent the inviolable space area of unmanned aerial vehicles to detect and avoid potential conflicts. Simulations show that the proposed algorithm is able to detect and avoid situations of potential conflict in the three-dimensional space and in real time, even without the assistance of a human operator. Schouwenaars T et al. [18] proposed a new approach to fuel-optimal path planning of multiple vehicles using a combination of linear and integer programming. A key benefit of this approach is that the path optimization can be readily solved using the CPLEX 9.0 [19] optimization software with an AMPL/Matlab interface.

The pedestrian prediction method based on the GAN model has become the mainstream pedestrian prediction method based on deep learning because of its outstanding ability to deal with future uncertainty. The SGAN model consists of a generator containing an autoencoder, a social pooling module and a decoder, and a decoder based on a long and short time series network, as shown in Figure 1. By constructing a social pool module, the model pools the relative movement and hidden state of pedestrians to obtain the interaction vector of pedestrians and then produces a track distribution closer to the actual track.

SGAN has made many improvements to the pedestrian interaction problem and achieved good results, but there are still some problems:

1. The interaction information obtained in the social pool is numerous and miscellaneous, and it is impossible to identify the information that is useful for predicting the future trajectory of pedestrians to be tested.
2. The function of the hidden code is ignored, so the generated trajectory is not accurate enough.
3. Without considering the scene constraints, the prediction of pedestrian trajectory not only takes into account the interaction between pedestrians but also needs to avoid some static buildings and other obstacles in the traffic scene.

4. The use of the L2 loss function leads to the risk of network collapse and limits the multi-modularity of the trajectory.

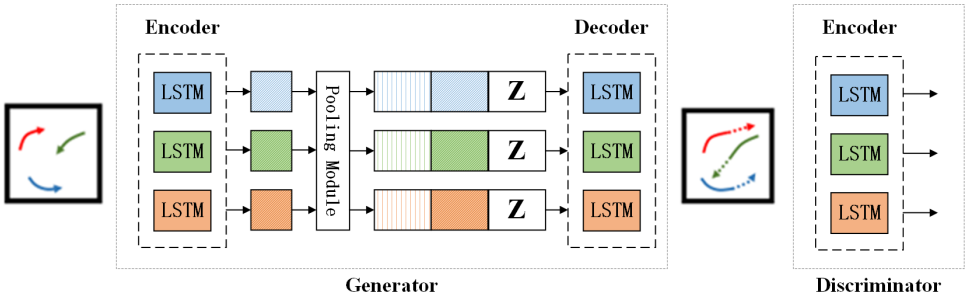


Figure 1. SGAN algorithm framework.

Aimed at solving problems such as the incomprehension of pedestrian interaction problems and scene constraint in the SGAN model, this paper proposes a scene context-based social information generative adversarial network (SC-SIGAN) pedestrian trajectory prediction method based on scene constraints. Compared with SGAN, the improved SC-SIGAN increases the attention mechanism and can fuse information at every moment. Secondly, by introducing mutual information, the correlation between the hidden code and the generator is strengthened to create the influence of the hidden code on the generated trajectory. In addition, the improved model introduces a new social pool and adds a scene edge extraction module, so that the model not only considers the location between the adjacent pedestrians and the target pedestrians in the scene but also considers the speed information of the pedestrians. This ensures that the final output path of the model is within the passable area in line with the physical scene and greatly improves the accuracy of trajectory prediction. The experimental results on the open dataset and the self-built dataset show that the improved model can better solve the uncertainty of pedestrian turning decisions, and improve the accuracy and stability of pedestrian trajectory prediction while maintaining multiple modes.

The overall arrangement of this paper is as follows: In the second part, firstly, the definition of pedestrian trajectory prediction based on deep learning is expounded, and then, the generation countermeasure network model based on scene constraints is described in depth, with emphasis on the improvement of the generator and discriminator. The third part is the experimental results and analysis. Using two public datasets, ETH [20] and UCY [21], and a self-made dataset on the CARLA simulation platform, and taking ADE and FDE as evaluation indicators, the Kalman filter (KF) algorithm [22], SLSTM algorithm, SGAN algorithm, ASGAN algorithm [23], and SC-SIGAN algorithm proposed in this paper are compared to verify the universality and accuracy of the pedestrian trajectory prediction model in this paper. The fourth part summarizes the novelty of the pedestrian trajectory prediction method in this paper, as well as the future development trends and challenges.

2. Generative Adversarial Network Model Based on Scene Constraints

2.1. Definition of Pedestrian Trajectory Prediction Problem Based on Deep Learning

Trajectory prediction means to understand pedestrian movement patterns by observing pedestrian time series data. In the pedestrian trajectory prediction network model, the future running state information of each pedestrian is usually predicted by observing the past running state information and scene information of all pedestrians in the scene. The input of the pedestrian trajectory prediction network model based on deep learning contains two pieces of information, one is the pedestrian trajectory information, and the other is the obstacle limitation information on the scene.

Let the pedestrian's past track information be defined as $X_t^u = (x_t^u, y_t^u)$, and predict that the output of the generator to $\hat{Y}_t^u = (\hat{x}_{t+t_{obs}}^u, \hat{y}_{t+t_{obs}}^u)$ represents the predicted future trajectory, then the true future trajectory is $Y_t^u = (x_t^u, y_t^u)$. Then there is:

$$X_t^u = X_1^u, X_2^u, X_3^u \ldots X_{obs}^u (u = 1 \ldots n, t \in [1, t_{obs}]) \tag{1}$$

$$\hat{Y}_t^u = \hat{Y}_1^u, \hat{Y}_2^u, \hat{Y}_3^u \ldots \hat{Y}_{pred}^u (t \in [t_{obs} + 1, t_{obs} + t_{pred}]) \tag{2}$$

In these formulas, u is the number of pedestrians to be measured, n is the total number of pedestrians to be measured, t_{obs} is the number of frames observed, and t_{pred} is the number of predicted frames.

Then, the speed of a pedestrian u at time t is:

$$V_t^u = \left(x_t^u - x_{t-1}^u, y_t^u - y_{t-1}^u\right) \tag{3}$$

Information about obstacles in the scene is entered into the network in the top view or side view.

2.2. SC-SIGAN Network Model

2.2.1. Overall Framework of Model

The structure of the SC-SIGAN model proposed in this paper is shown in Figure 2. Its framework also adopts the basic structure of generating adversarial networks, which is composed of a generator (G) and discriminator (D).

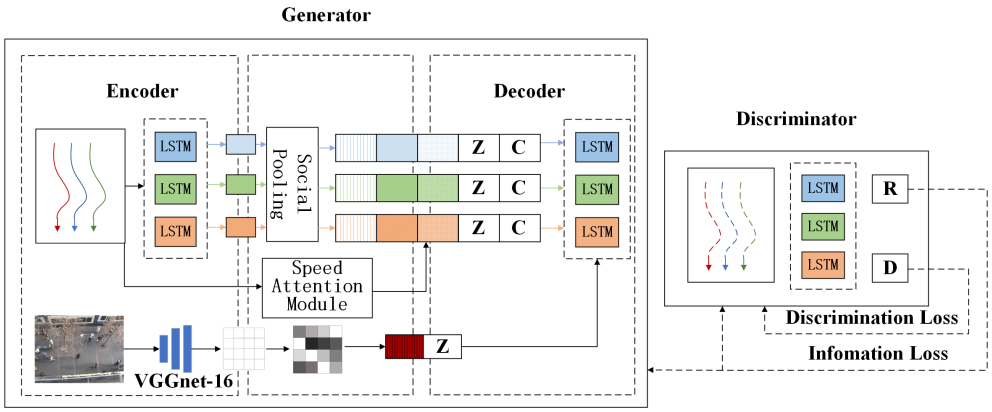


Figure 2. Structure of SC-SIGAN model generator and encoder.

The generator includes three parts: an encoder, a social pool, and a decoder. The encoder extracts the features from the original track and image through the LSTM network and the Visual Geometry Group (VGG) network and encodes them. They are then transferred to the social pool (location and velocity attention pooling (LVAP)) for screening, important weighted feature information, noise, and an initialized latent code are inputted into the decoder for decoding, the updated latent code is obtained, and the generation of the predicted trajectory is controlled. The discriminator improves the performance of the generator model by forcing the generator to generate prediction samples that are closer to the real trajectory.

2.2.2. Generator

1. Encoder

   The function of the encoder is to upgrade the 2D trajectory sequence of pedestrians into a high-dimensional vector on the one hand and to realize the feature extraction of the scene on the other hand.

First, through the connection layer network $\phi(\cdot)$, the trajectory sequence X_t^u of each selected pedestrian is raised from two-dimensional coordinates to a higher dimensional vector e_t^u. The coding formula is as follows:

$$e_t^u = \phi(X_t^u, W_{\phi 1}) \tag{4}$$

In the formula, $W_{\phi 1}$ is the weight parameter of $\phi(\cdot)$ in the fully connected network in the encoder.

Then, after the e_t^u is embedded by an embedding function γ with ReLU nonlinear activation, the previous state feature H_{t-1}^{eu} is inputted to the encoder LSTM module for encoding. All information is encoded until the end of the observation sequence, and the current motion state features H_t^{eu} of the pedestrian u are updated.

$$H_t^{eu} = LSMT(H_{t-1}^{eu}, \gamma(e_t^u, W_\gamma); W_{ec}) \tag{5}$$

In the formula, W_γ is the weight parameter of the function γ, and W_{ec} is the weight parameter of the encoder, initialized by pre-training fine-tuning.

The feature extraction of the encoder scene is completed by VGGnet-16. The VGG network [24] maps the feature image generated by the convolutional layer into a fixed-length feature vector, and the resulting classification still belongs to the image-level classification. In order to complete the class semantic segmentation and extraction of scene features, the last full connection layer of VGG is changed to the full convolution layer so that the output layer outputs the softmax loss calculated on a pixel-to-pixel basis, and finally, the pixel-to-level classification is obtained. The changed network structure is shown in Figure 3.

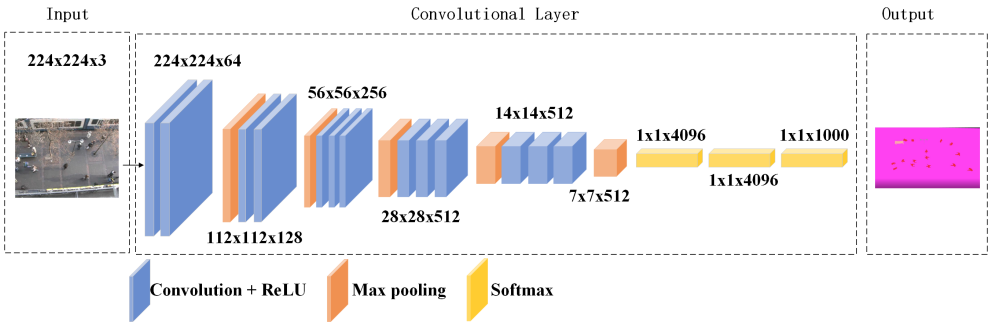


Figure 3. Changed VGG network structure.

Through the full convolutional VGG network structure, the scene features obtained from the image Image$_t$ are:

$$S_t = FCN(\text{Image}_t, W_{fcn}) \tag{6}$$

In the formula, W_{fcn} is the weight of the full convolutional network.

2. Information screening

The feature screening is divided into two parts. The first part screens the pedestrian motion state features in the encoder and collects the feature information useful for determining the future direction of the pedestrian u. The second part enables the model to understand the interaction between the scene and the pedestrian by applying soft attention.

The first part is composed of the social pool layer structure (as shown in Figure 4) and the self-attention module. The former is concerned with the relative displacement change in pedestrian movement, and the latter is concerned with the relative speed change in pedestrian movement. The second part adopts the "soft" deterministic attention mechanism $ATT(\cdot)$ proposed by Xu et al. [25] through the standard backpropagation method.

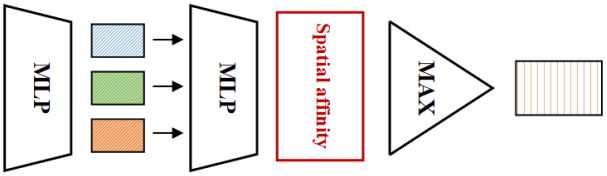


Figure 4. Social pooling layer structure.

(a) Calculate the relative displacement change information p_t^{um} of pedestrian u and its neighboring pedestrians.
The relative influence of pedestrians can generally be analyzed by spatial affinity. Let $\xi_t^{um} \in O^3$ represent the spatial affinity between the pedestrian u and the close pedestrian m around him, which includes three parts: Euclidean distance, azimuth angle, and nearest approach distance between pedestrian u and pedestrian m. Then, the relative position information between pedestrian u and pedestrian m can be calculated from o_t^{um}:

$$o_t^{um} = \{\xi_t^{um} | t = 1, \ldots, t_{obs}\} \in O^3, u \neq m \tag{7}$$

Then, o_t^{um} is mapped to p_t^{um} through the fully connected network $\phi(\cdot)$, and the relative displacement change information p_t^{um} between pedestrian u and the closely interacting pedestrian m around him is obtained:

$$p_t^{um} = \phi(o_t^{um}, W_{\phi 2}),\ u \neq m \tag{8}$$

In the formula, $W_{\phi 2}$ is the weight parameter for this fully connected layer.

(b) The attention weight b_t^{um} of the relative displacement change between the pedestrian u and its neighbors is calculated.
The relative displacement change p_t^{um} between pedestrians u and m is transformed into a high-dimensional vector, which is embedded into H_t^{em} (motion feature information of adjacent pedestrians) by a fully connected layer to obtain $\varrho(p_t^{um}, H_t^{em})$.

$$\varrho(p_t^{um}, H_t^{em}) = \frac{N-1}{\sqrt{d_\varrho}} < p_t^{um}, W_\varrho H_t^{em} >,\ u \neq m \tag{9}$$

In the formula, N is the total number of pedestrians, d_ϱ is p_t^{um} and the common row of linear map weights applied to the motion feature information, and W_ϱ is the weight parameter of the fully connected layer.
Finally, the attention weight b_t^{um} is obtained by scalar product and softmax by using p_t^{um} and H_t^{em} to obtain the relative displacement change of pedestrian u and each adjacent pedestrian m:

$$b_t^{um} = \frac{exp\left(o(p_t^{um}, H_t^{um})\right)}{\sum_{n \neq u} exp(o(p_t^{um}, H_t^{um}))}, u \neq m \tag{10}$$

$$b_t^{um} \triangleq [b_t^{u1},\ b_t^{u2},\ b_t^{u3} \ldots b_t^{un}]^T, m \in [1, n],\ u \neq m \tag{11}$$

(c) Calculate the relative speed change information C_t^u of pedestrians u and their neighbors.
Since the spatial affinity in the social pool can only pay attention to the distance information of the displacement between pedestrians, to better analyze the interaction between pedestrians, it is also necessary to pay attention to the influence of the speed change between pedestrians. Here, the self-attention mechanism model shown in Figure 5 is adopted, focusing on the speed information of each pedestrian.

Let the input of the model be the speed of each pedestrian in the scene at time t $V_t^i (i = 1, 2, \ldots, n)$, then output the attention information of relative speed.

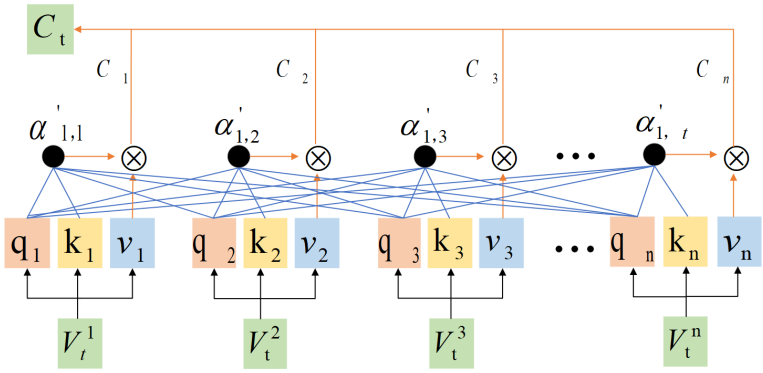


Figure 5. Self-attention module.

In Figure 5, q represents the query, k represents the key, v represents the value, α represents attention distribution, α' represents attention distribution after normalization, and C_t represents output attention information.
The formulas for calculating the q_t^u and k_t^u of the pedestrian u to be measured are:

$$q_t^u = W_q V_t^u \tag{12}$$

$$k_t^u = W_k V_t^u \tag{13}$$

In the formula, W_q and W_k are weight matrices.
The correlation degree α of the pedestrian u to be measured with the speed of other pedestrians is calculated as follows:

$$\alpha_{\mathrm{u},1} = q_t^{\mathrm{u}} k_t^1, \alpha_{\mathrm{u},2} = q_t^{\mathrm{u}} k_t^2, \ldots, \alpha_{\mathrm{u},n} = q_t^{\mathrm{u}} k_t^n \tag{14}$$

Through the calculation of softmax, all the correlation degrees are normalized, the attention distribution α' is obtained, and the speed of adjacent pedestrians at the same time is related to the speed of the pedestrian to be measured.

$$\alpha'_{\mathrm{u},n} = \exp(\alpha_{\mathrm{u},n}) / \sum_n \exp(\alpha_{\mathrm{u},n}) \tag{15}$$

Finally, the relative velocity information C_t^u is extracted according to the attention distribution.

$$C_t^u = \sum_1^n v_t^u \alpha'_{u,n} \tag{16}$$

In the formula, v_t^u is the key value of the pedestrian to be measured:

$$v_t^u = W_v V_t^u \tag{17}$$

In the formula, W_v is the weight matrix.

(d) Calculate the interaction information A_t^u between the scene and the pedestrian. The "soft" deterministic attention mechanism $ATT(\cdot)$ can make the model pay attention to the edge of static obstacles in the scene so that the final output path of the whole model is within the passable area that conforms to the physical scene. Interactive information is represented as:

$$A_t^u = ATT(S_t, H_t^{eu}, W_{ATT}) \tag{18}$$

In the formula, S_t represents the scene feature, H_t^{eu} is the motion feature information of the pedestrian u, and W_{ATT} is the weight of the attention mechanism module.

3. Decoder

According to the weight of the important information obtained after the above screening, the decoder can combine the motion state H_t^{eu} of pedestrian u and the motion state H_t^{em} of the adjacent pedestrian m to obtain the useful hidden feature σ_{t-1}^u of pedestrian movement:

$$\sigma_{t-1}^u = [(H_{t-1}^{eu})^T, (\sum_{u \neq m} b^{um} H_{t-1}^{em})^T, (C_{t-1}^u)^T, (A_{t-1}^u)^T, (Z)^T]^T \tag{19}$$

In the formula, H_{t-1}^{em} is the motion state information of adjacent pedestrian m at the last moment, and $\sum_{u \neq m} b^{um} H_{t-1}^{em}$ is the influence of the relative displacement change of the surrounding pedestrian m at the last moment on the future trajectory of the pedestrian u. C_{t-1}^u is the influence of the relative speed change of the surrounding pedestrian m on the future trajectory of pedestrian u, $A(t-1)^u$ is the influence of static obstacles in the scene of the previous moment on the future trajectory of the pedestrian u, and Z is noise.

The pedestrian trajectory $\widehat{Y}_t^u$ is predicted according to the motion hidden feature σ_{t-1}^u and the current motion state of the pedestrian H_t^{du}. The initial current motion state information of the pedestrian U received by the long and short time series network in the decoder is H_t^{du}, which is obtained by the state $H_t^e u$ cascaded high-level noise Z of the encoder $t = t_{obs}$:

$$H_t^{du} = [H_t^{eu}, Z] \tag{20}$$

After updating H_t^{du}, it is necessary to combine the motion state information H_{t-1}^{du} of the last moment and the useful hidden features σ_{t-1}^u of the attention mechanism module of the last moment into the long and short time series network.

$$H_t^{du} = \lambda^d(H_{t-1}^{du}, \sigma_{t-1}^u; W_{\lambda^d}) \tag{21}$$

In the formula, λ^d is the decoding unit function of the long and short time series network, and W_{λ^d} is the weight of the long and short time series network in the decoder.

Then, the updated current motion state H_t^{du} is converted into the coordinate space by gamma function γ, and the predicted future trajectory $\widehat{Y}_t^u$ is obtained:

$$\widehat{Y}_t^u = \gamma(H_t^{du}, W_\gamma) \tag{22}$$

In the formula, W_γ is the weight of the function γ.

2.2.3. Discriminator

1. Code enhancement

Based on the original SGAN network, mutual information is used as an optimization target to enhance the role of latent code in predicting trajectory generation. Through model training, the difference between mutual information lower bound and mutual information distribution becomes smaller, so that the correlation between the latent code and the predicted trajectory becomes larger, and the generated predicted trajectory is closer to the real trajectory. The designed SC-SIGAN network is also composed of generator G, discriminator D, and subnetwork R. During training, the discriminator has nothing to do with mutual information, and the parameters of the generator are fixed, so the change in mutual information is only determined by the subnetwork.

According to the definition of mutual information, obtain hidden code C and generator-generated forecast track X mutual information $I(C; X) = I(C; G(Z, C))$ as follows:

$$I(C; X) = \sum_{c \in C} \sum_{x \in X} p(c|x) \log\left(\frac{p(c|x)}{p(c)p(x)}\right) = H(C) - H(C|X) \tag{23}$$

In the formula, $P(x)$ is the distribution probability of x, $P(c)$ is the distribution probability of c, $p(c|x)$ is the probability of c occurring under the condition that x occurs, $H(C)$ represents the information entropy of C, and $H(C|X)$ represents the uncertainty of C given X.

The posterior distribution $p(C|X)$ in Equation (23) can be estimated by defining an auxiliary distribution $R(C|X)$:

$$\begin{aligned} I(C;X) &= H(C) - H(C|X) \\ &= E_{x\sim G(z,c)}\Big[E_{c'\sim P(z|c)}[logP(c'|x)]\Big] + H(c) \\ &= E_{x\sim G(z,c)}\Big[D_{kl}\big(P(\cdot|x) \parallel R(\cdot|x)\big) + E_{c'\sim P(z|c)}[logR(c'|x)]\Big] + H(c) \\ &\geq E_{x\sim G(z,c)}\Big[E_{c'\sim P(z|c)}[logR(c'|x)]\Big] + H(c) \end{aligned} \tag{24}$$

In the formula, $H(C)$ is a constant. $D_{kl}\big(P(\cdot|x) \parallel R(\cdot|x)\big)$ is the divergence, a measure of the difference between $P(C|X)$ and $R(C|X)$.

$E_{x\sim G(z,c)}\Big[E_{c'\sim P(z|c)}[logR(c'|x)]\Big] + H(c)$ in Equation (24) is the lower bound $I_L(C;Q)$ of $I(C;X)$.

$$\begin{aligned} I_{\mathrm{L}}(C;Q) &= E_{x\sim G(z,c)}\Big[E_{c'\sim P(z|c)}[\log R(c' \mid x)]\Big] + H(c) \\ &= E_{c\sim p(c),x\sim G(z,c)}[\log Q(c \mid x)] + H(c) \end{aligned} \tag{25}$$

After adding the loss function generated by the adversarial network structure generated by the entire model itself, the overall optimization objective is:

$$\min_{G,Q}\max_{D} V(R,G,D) = V(D,G) - \lambda I_L(C;Q) \tag{26}$$

In the formula, G is the generator and D is the discriminator.

2. Loss function

Similar to SGAN, LSTM is used to encode the input of the discriminator, and the accuracy of the predicted trajectory is judged using the fully connected layer.

(a) Discriminator D total loss function d_{-loss}

$$d_{-loss} = -E_{x\sim pdata}logD(x) - E_{z,c}log\Big(1 - D\big(G(z,c)\big) - \lambda I\big(c,G(z,c)\big)\Big) \tag{27}$$

In the formula, λ is constant.

(b) Loss function $R_{info-loss}$ generated by network R.

$$R_{info-loss} = L_1(G,Q) = E_{c\sim p(c),x\sim G(z,c)}[logQ(c|x)] + H(c) \tag{28}$$

(c) Generator G total loss function g_{-loss}

$$g_{-loss} = -E_{z,c}log\Big(D\big(G(z,c)\big) - \lambda I\big(c,G(z,c)\big)\Big) \tag{29}$$

The above brings the generated predicted trajectory closer to the characteristics pointed out by the latent code C. For example, if the character of the hidden code C is that the trajectory of the person in the line is shifting to the right, then the generated predicted trajectory will continue to the right until it approaches the direction of the shift of the person in the line.

In Figure 6, the pseudocode of the SC-SIGAN network model is as follows:

pseudocode 1: A pedestrian trajectory prediction method for generative adversarial networks based on scene constraints

Input: Pedestrian historical trajectory; Images;

1: **Generator:**
2: - Encoder:
3: Through LSTM network and convolutional neural network.
4: - Location and velocity attention pooling
5: - Decoder
6: - Output the predicted trajectory of pedestrians.
7: **Discriminator:**
8: - The predicted trajectory and real trajectory of pedestrians pass through LSTM network.
9: - Mutual information function Q.
10: - Loss function (e.g., Discriminator total loss function).
11: Feedback to discriminator and generator

Figure 6. The pseudocode of SC-SIGAN network model.

3. Experimental Results and Analysis

3.1. Experimental Environment and Dataset

3.1.1. Experimental Environment

The model was performed using Python 3.6 on PyTorch 0.8, using the Adam optimizer for iterative training to optimize the parameters of SC-SIGAN.

All internal fully connected layers of the trace generator and discriminator are associated with the LeakyReLU activation function with a slope of 0.1. In each dataset, the SC-SIGAN network is trained using the following parameter settings:

Minimum batch size 64, generator learning rate 0.001, discriminator learning rate 0.0001, momentum 0.9, and training 2000 rounds. The parameter optimization process is as follows:

1. Initial attenuation rate vector input l_r, $\beta_1 = 0.9$, $\beta_2 = 0.999$ can learn parameters θ_0, $\in = 10^{-8}$.
2. Set the initial cumulative gradient: $m_0 = 0$, the square of the initial cumulative gradient $v_0 = 0$, and the initial training number $t = 0$.
3. Training times are updated: $t = t + 1$.
4. Cumulative gradient: $\mathrm{m_t} = \beta_1 * \mathrm{m_{t-1}} + (1 - \beta_1) * g_t$, g_t is the gradient of each parameter itself.
5. Cumulative gradient squared: $v_t = \beta_2 * v_{t-1} + (1 - \beta_2) * (g_t)^2$.
6. Deviation correction: $\widehat{\mathrm{m_t}} = \frac{\mathrm{m_t}}{1-(\beta_1)^2}$, $\widehat{v_t} = \frac{v_t}{1-(\beta_2)^2}$.
7. Update parameters: $\theta_t = \theta_{t-1} - \frac{m_t}{\sqrt{\widehat{v_t}+\epsilon}} lr$.

3.1.2. Dataset Selection

In this paper, two public datasets, ETH [20] and UCY [21], and a self-made dataset on the CARLA [26] simulation platform are used to verify the generalization and accuracy of the pedestrian trajectory prediction model.

1. Public datasets
   The training set accounts for 70% of the total dataset, and the test set accounts for 30% of the total dataset [27]. The cross-validation method is adopted to train the model, and four other subdatasets are taken as training data. Each subdataset is trained, from which the model with the best performance on the verification set is selected.

2. Self-built dataset
   The real trajectory observed in existing publicly available datasets for trajectory prediction evaluation is only one of many possible future trajectories that conform to social norms. Liang [28] et al. proposed a simulation map based on the real traffic environment, which can provide richer semantic information. CARLA 0.9.6 and Unreal Engine 4 were used to build a simulation platform for the real traffic environment, reconstruct the static scene and its dynamic elements, and obtain the simulated traffic scene as shown in Figure 7. Then, the dataset was manually labeled by controlling the direction of the target pedestrian to be measured.

Figure 7. Simulated traffic scene.

3.2. Evaluation Target

Average Differential Error (ADE) and Final Differential Error (FDE) are used as evaluation indexes for trajectory prediction.

$$ADE = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{t_{pred}}\sum_{t=t_{obs}+1}^{t_{obs}+t_{pred}}\sqrt{\left(x_i^t-\hat{x}_i^t\right)^2+\left(y_i^t-\hat{y}_i^t\right)^2} \tag{30}$$

$$FDE = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(x_i^{t_{ped}}-\hat{x}_i^{t_{ped}}\right)^2+\left(y_i^{t_{pred}}-\hat{y}_i^{t_{ped}}\right)^2} \tag{31}$$

ADE represents the accuracy of the predicted trajectory at every time t average, and FDE represents the accuracy of the predicted trajectory at the last moment.

3.3. Open Dataset Experimental Results and Analysis

In order to evaluate the effect of the model, SC-SIGAN, in this paper, is compared with several common trajectory prediction methods, including the Kalman filter (KF) algorithm [22], SLSTM algorithm, SGAN algorithm, and ASGAN algorithm [23].

3.3.1. Data Comparison and Analysis

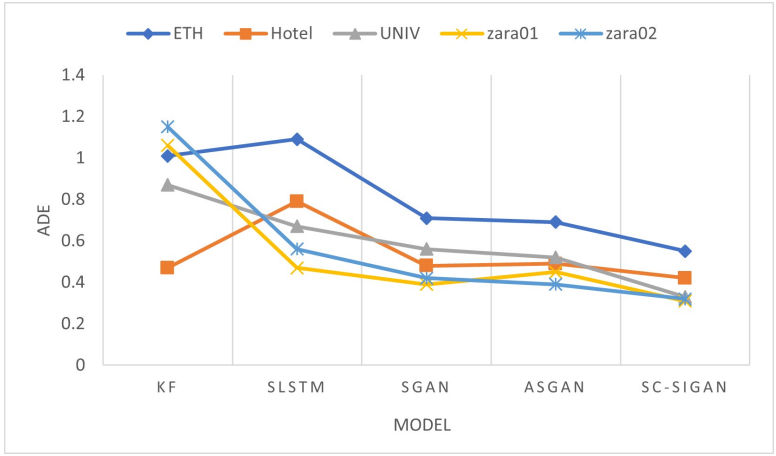
Table 1 shows the prediction results of the above methods on the public dataset when $t_{obs} = 8$ and $t_{pred} = 12$, and the error measure is len12. In the table, ETH (E) indicates the off-campus scene, Hotel (H) indicates the hotel scene, Univ (U) indicates the campus scene, and Z1 and Z2 indicate the shopping scene..

As can be seen from Table 1, minimum error values are indicated in bold, the ADE and FDE of the SC-SIGAN model in this paper are the best on all datasets, except for the NKF method used on the Hotel dataset. This is because the scene in the Hotel dataset is not crowded, pedestrians generally have no interaction, and people usually keep the same rhythm as their previous movements. This is more consistent with the regularity of pedestrian movement.

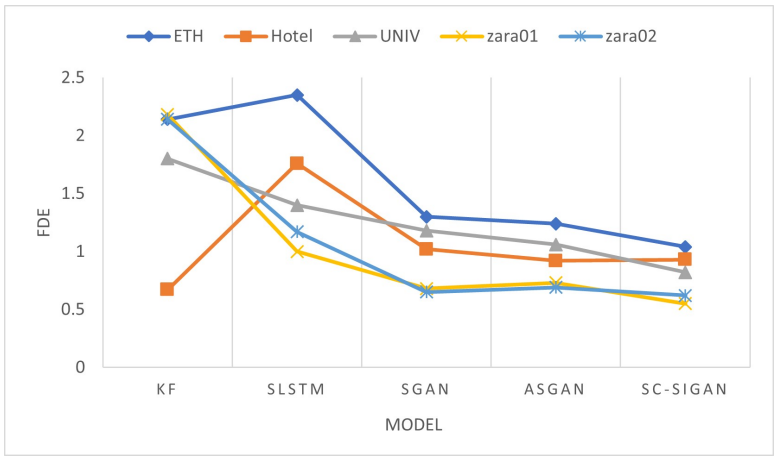
Table 1. Open dataset testing.

Indices	Model/Dataset	E	H	U	Z1	Z2	Average
ADE	KF	1.01	0.47	0.87	1.06	1.15	0.91
	NKF	0.91	**0.39**	0.58	0.75	0.53	0.63
	SLSTM	1.09	0.79	0.67	0.47	0.56	0.72
	SGAN	0.71	0.48	0.56	0.39	0.42	0.51
	ASGAN	0.69	0.49	0.52	0.45	0.39	0.45
	OURS	**0.55**	**0.42**	**0.33**	**0.31**	**0.32**	**0.38**
FDE	KF	2.14	0.67	1.8	2.18	2.14	1.79
	NKF	1.87	**0.62**	1.23	0.92	1.02	1.13
	SLSTM	2.35	1.76	1.4	1	1.17	1.54
	SGAN	1.3	1.02	1.18	0.68	0.65	0.96
	ASGAN	1.24	0.92	1.06	0.73	0.69	0.92
	OURS	**1.04**	**0.93**	**0.82**	**0.55**	**0.62**	**0.79**

In other datasets, SC-SIGAN showed a 25.4% decrease in average ADE and a 17.7% decrease in average FDE compared to pre-modified SGAN. The ADE index reflects the prediction errors at different moments in the prediction process. The fusion of pedestrian motion features in the method proposed in this paper is based on each moment. The decline in ADE indicates that the method effectively reduces the prediction errors at different time points, making the pedestrian trajectory features obtained more effective and improving the prediction accuracy. The Table 1 test comparison curve is shown in Figure 8.



(**a**)



(**b**)

Figure 8. Open dataset test metrics comparison curve. (**a**) *ADE* indices. (**b**) *FDE* indices.

The proposed algorithm is compared with other algorithms to predict the speed on the same server, and the 12-step prediction time of a single pedestrian is shown in Table 2.

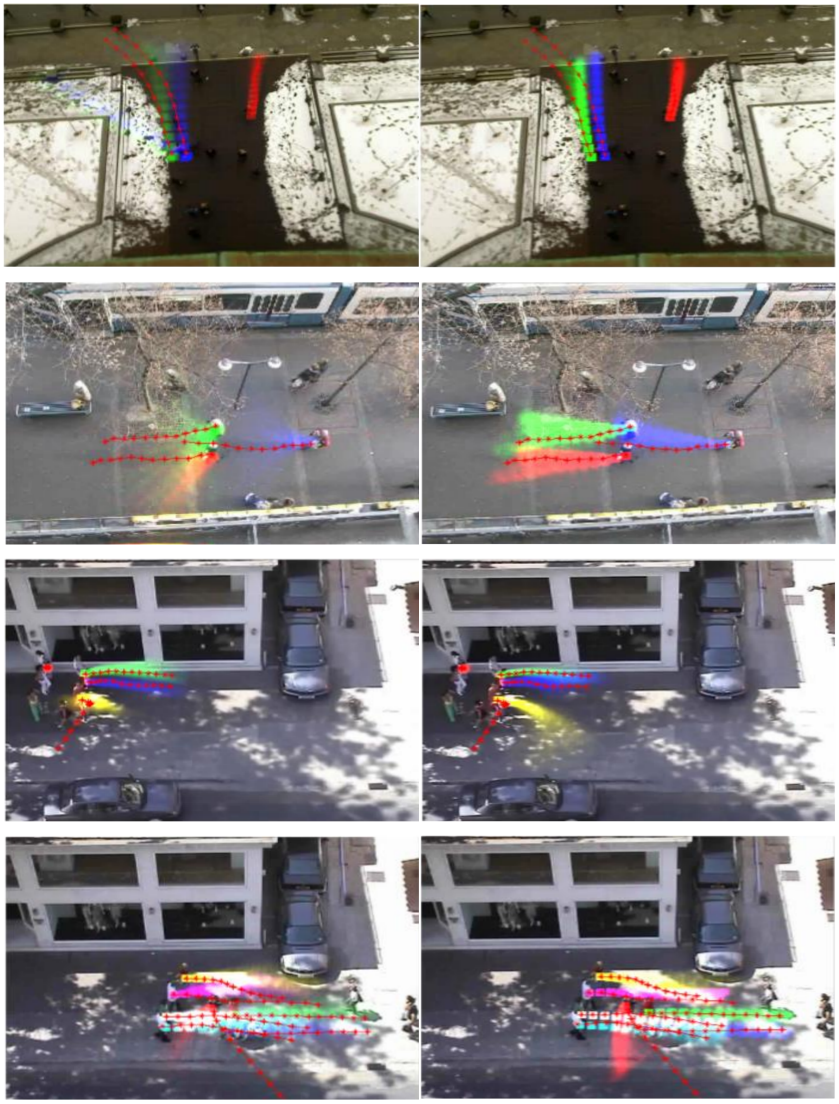
Table 2. The 12-step prediction time for a single pedestrian.

Indices	KF	NKF	SLSTM	SGAN	ASGAN	OURS
Average forecast time/ms	1.69	9.81	403.31	42.52	44.62	47.2
Acceleration effect	×1	×6	×238	×25	×26	×28

In Table 2, based on the KF algorithm, the acceleration effect of the SGAN algorithm is about 25 times, and the acceleration effect of the SC-SIGAN algorithm improved by SGAN in this paper is about 28 times that of KF, so it can be seen that the overall accuracy of this algorithm is improved without spending too much time.

3.3.2. Visual Comparison and Analysis

Different scenes were captured in the above public dataset, and SGAN and SC-SIGAN in this paper were, respectively, used to test the visualization effect with the results shown in Figure 9.



(**a**) (**b**)

Figure 9. Algorithm visual comparison. (**a**) SGAN. (**b**) Our model.

In Figure 9, the red dotted line is the actual trajectory, and the green, blue, and red three-color light bar is the predicted path. It can be seen that the path predicted by the improved algorithm is significantly more accurate.

3.4. Experimental Results and Analysis of Self-Built Dataset

3.4.1. Dataset Information

In the traffic scene simulation platform built by CARLA, set the "controlled target" and the destination with practical significance, and then control the movement of the target, so that the target moves to the specified destination in a "natural" way. The use of 10.4 s to represent the future in the simulation is more conducive to the evaluation of the model for long-term predictions. Figure 10 shows the visualization effect of trajectory prediction in the self-built dataset. In the figure, the yellow dots are the trajectory used for observation, and the green dots are the future real trajectory used to compare the prediction effect.

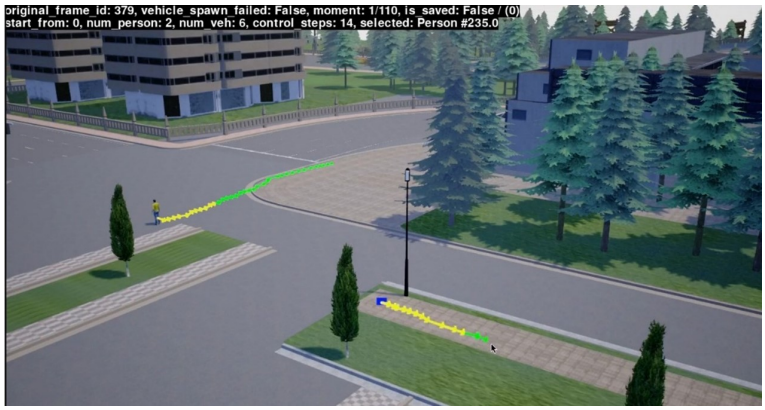


Figure 10. Visualization effect of trajectory prediction of self-built dataset.

Finally, the trajectory data files were made, and a total of 750 data files were formed, of which 230 belonged to sparse traffic scenes with only target pedestrians and a static environment, and 520 belonged to dense traffic scenes with pedestrians gathering.

3.4.2. Experimental Results and Analysis

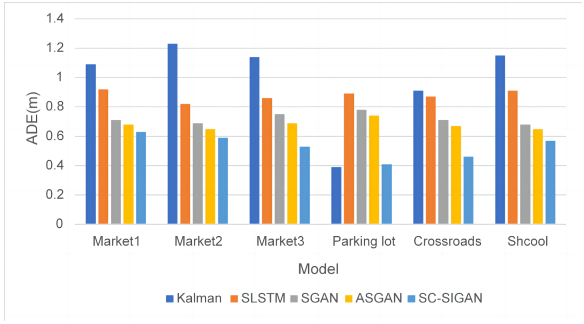
The proposed method and the above methods were tested and evaluated on the datasets VIRAT/ActEV [29] and ETH and UCY, respectively. The first two datasets are generally used for single-person trajectory prediction in crowded scenarios, while the last two datasets are generally used for multi-person trajectory prediction. Test ADE and FDE metrics as shown in Figure 11.

As shown in Figure 11, the accuracy of the proposed model is superior to other methods on all datasets except the Parking lot because the Parking lot dataset is the dataset of the single scene. Compared with SGAN before improvement, the average ADE and FDE of the proposed method decreased by 26.4% and 23.8%, respectively.

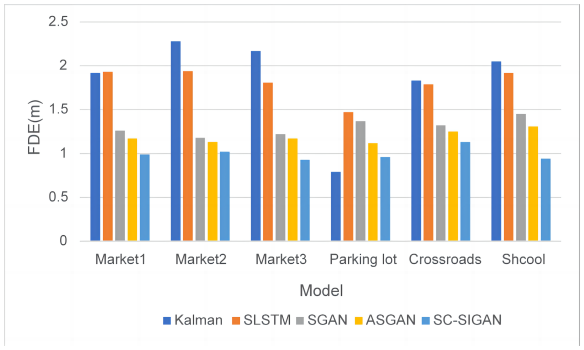
Four typical scenes were selected from the six simulation scenes, and the main view diagram of the scene was used as a visual display of the trajectory prediction effect, as shown in Figure 12.

Figure 12a shows that in the single-person scenario, SGAN cannot distinguish the influence of static obstacles, resulting in the possibility of collision with obstacles in other parts except for the smooth passage of some predicted results. In the multi-person scenario, the prediction of avoiding other pedestrians is made as much as possible, but the problem of collision with static obstacles still exists, and the multiple possibilities of trajectory are not obvious because the impact of hidden code in the network is small.

Figure 12b shows that the model in this paper has a good effect on both the avoidance of static obstacles in a single scene and the processing of interactive information between pedestrians in a multi-person scene.

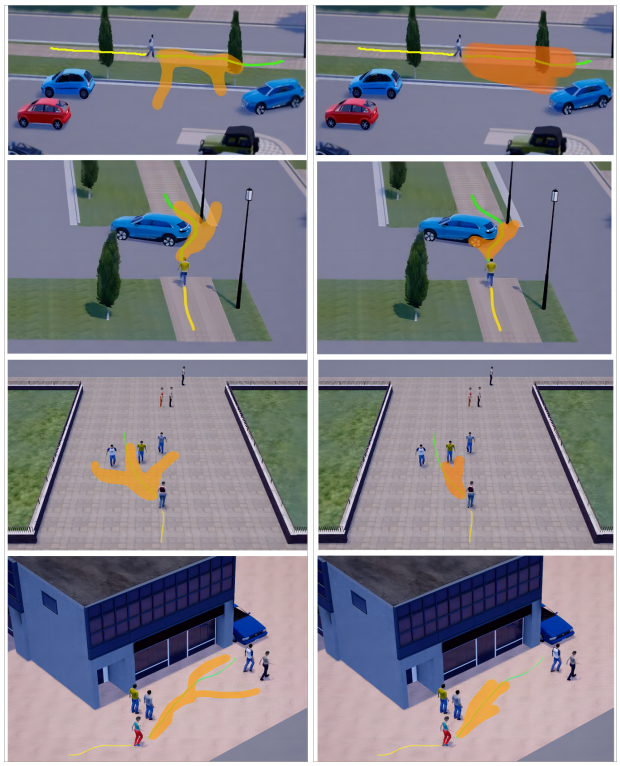


(**a**)



(**b**)

Figure 11. Histogram of test results of five algorithms. (**a**) *ADE* indices. (**b**) *FDE* indices.



(**a**) (**b**)

Figure 12. Visualization of trajectory prediction results. (**a**) SGAN. (**b**) Our model.

4. Conclusions

To solve the problem that the generative adversarial network prediction model lacks an understanding of pedestrian interaction problems and scene constraints, this paper improves the original generative adversarial network trajectory prediction model by introducing a new social pool and adding a scene edge extraction module inspired by the attention mechanism. Thus, the improved model SC-SIGAN not only considers the position between the adjacent pedestrian and the target pedestrian in the scene, but also considers the speed information of the pedestrian, and makes the final output path of the entire model within the passable area in line with the physical scene. Experiments show that this method improves the accuracy of trajectory prediction to some extent on common datasets. In this paper, the CARLA simulation platform is also used to annotate the self-built dataset to test the effect of the proposed method and other existing methods. The SC-SIGAN algorithm achieved excellent results in maintaining multi-mode and accuracy. Finally, in pedestrian movement prediction, the most important thing is to use these models in the application, so there is still room for improvement for the practical application of the model in this paper. This paper is based on the information collected by vehicle-mounted cameras, but there will be some errors in the capture of environmental information by visual sensors. In the future, we can combine the information collected by lidar sensors to model pedestrian trajectory prediction. In addition, the method proposed in this paper is suitable for unmanned vehicles to judge the behavior of pedestrians around them, but there are often car-to-car interaction problems in actual scenes, so in future research, different types of targets need to be predicted at the same time to adapt to more realistic traffic scenes.

Author Contributions: Methodology, Z.M. and Z.S.; Software, J.L. (Jiajia Liu) and J.Q.; Validation, R.A., J.Q. and Z.S.; Formal analysis, Y.C. and J.L. (Juguang Li); Investigation, Y.C.; Resources, Y.T. and G.Z. Data curation, Y.C.; Writing—original draft, Z.M.; Writing—review & editing, R.A.; Visualization, Y.T.; Supervision, J.L. (Jiajia Liu) and J.L. (Juguang Li); Project administration, Z.M.; Funding acquisition, J.L. (Jiajia Liu). All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Sichuan Science and Technology Program China, under Grant No.2022YFS0565; the Key R&D project of the Science and Technology Department of Sichuan Province, under Grant 2023YFG0196 and 2023YFN0077; the Science and Technology achievements transformation Project of the Science and Technology Department of Sichuan Province, under Grant 2023JDZH0023; the Sichuan Provincial Science and Technology Department, Youth Fund project, under Grant 2023NSFSC1429; the Key Laboratory of Lidar and Device, P.R.China LLD2023-010.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: Author Juguang Li was employed by Chengdu Emfuture Technology Co., Ltd. The rest of the authors declare they have no conlficts of interest.

References

1. Xu, S. Research on Pedestrian Trajectory Prediction Method Based on Graph Neural Network. Ph.D. Thesis, Hefei University of Technology, Hefei, China, 2022.
2. Helbing, D.; Molnar, P. Social Force Model for Pedestrian Dynamics. *Phys. Rev. E* **1995**, *51*, 4282. [CrossRef] [PubMed]
3. Alahi, A.; Ramanathan, V.; Fei-Fei, L. Socially-aware large-scale crowd forecasting. In Proceedings of the IEEE Conference on Computer Vision and Patter Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2203–2210.
4. Sumpter, N.; Bulpitt, A. Learning spatio-temporal patterns for predicting object behaviour. *Image Vis. Comput.* **2000**, *18*, 697–704. [CrossRef]
5. Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; Savarese, S. Social lstm: Human trajectory prediction in crowdedspaces. In Proceedings of the EEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 961–971.
6. Bartoli, F.; Lisanti, G.; Ballan, L.; Del Bimbo, A. Context-aware trajectory prediction. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 1941–1946.
7. Varshneya, D.; Srinivasaraghavan, G. Human trajectory prediction using spatially aware deepattention models. *arXiv* **2017**, arXiv:1705.09436.

8. Raipuria, G.; Gaisser, F.; Jonker, P.P. Road infrastructure indicators for trajectory prediction. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 537–543.
9. Gupta, A.; Johnson, J.; Fei-Fei, L.; Savarese, S.; Alahi, A. Social gan: Socially acceptable trajectories with generativeadversarial networks. In Proceedings of the IEEE Conference on Computer Vision and PatternRecognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2255–2264.
10. Amirian, J.; Hayet, J.B.; Pettre, J. Social ways: Learning multi-modal distributions of pedestriantrajectories with gans.In Proceedings of the lEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 16–17 June 2019.
11. Lao, L.; Du, D.; Chen, P. Predicting Pedestrian Trajectories with Deep Adversarial Networks Considering Motion and Spatial Information. *Algorithms* **2023**, *16*, 566. [CrossRef]
12. Pei, Z.; Qiu, W.-T.; Wang, M.; Ma, M.; Zhang, Y.-N. Pedestrian Trajectory Prediction Method Using Dynamic Scene Information Based Transformer Generative Adversarial Network. *Acta Electonica Sin.* **2022**, *50*, 1537–1547. [CrossRef]
13. Li, X. Research on Trajectory Position Prediction Technology Based on Recurrent Neural Network. Ph.D. Thesis, Zhejiang University, Hangzhou, China, 2016.
14. De Brébisson, A.; Simon, É.; Auvolat, A.; Vincent, P.; Bengio, Y. Artificial neural networks applied to taxi destinationprediction. *arXiv* **2015**, arXiv:1508.00021.
15. Khurana, T.; Hu, P.; Held, D.; Ramanan, D. Point Cloud Forecasting as a Proxy for 4D Occupancy Forecasting. *arXiv* **2023**, arXiv:2302.13130.
16. Kuchar, J.K.; Yang, L.C. A review of conflict detection and resolution modeling methods. *IEEE Trans. Intell. Transp. Syst.* **2000**, *1*, 179–189. [CrossRef]
17. Migliaccio, G.; Mengali, G.; Galatolo, R. A solution to detect and avoid conflicts for civil remotely piloted aircraft systems into non-segregated airspaces. *Proc. Inst. Mech. Eng. G J. Aerosp. Eng.* **2016**, *230*, 1655–1667. [CrossRef]
18. Schouwenaars, T.; De Moor, B.; Feron, E.; How, J. Mixed integer programming for multi-vehicle path planning. In Proceedings of the 2001 European Control Conference (ECC), Porto, Portugal, 4–7 September 2001; pp. 2603–2608.. [CrossRef]
19. ILOG, Inc. CPLEX User's Manual. 2003. ILOG CPLEX 9.0 Reference Manual [DB/OL]. 2003. Available online: http://www.ilog.com/ (accessed on 27 December 2023).
20. 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 International Conference on ComputerVision, Kyoto, Japan, 29 September–2 October 2009; pp. 261–268.
21. Alexiadis, V.; Colyar, J.; Halkias, J.; Hranac, R.; McHale, G. The next generation simulation program. *Inst. Transp. Eng. ITE J.* **2004**, *74*, 22.
22. Yang, B.; Liu, C.; Zheng, W.; Liu, S. Motion prediction via online instantaneous frequency estimation for vision-based beating heart tracking. *Inf. Fusion* **2017**, *35*, 58–67. [CrossRef]
23. Wang, N. Research on Pedestrian Detection Algorithm and Its Safety in Unmanned Driving. Ph.D. Thesis, Nanjing University of Posts and Telecommunications, Nanjing, China, 2020.
24. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
25. 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 International Conference on Machine Learning, Lille, France, 7–9 July 2015.
26. Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; Koltun, V. CARLA: An open urban driving simulator. *arXiv* **2017**, arXiv:1711.03938.
27. Zhang, S. Research on Pedestrian Trajectory Prediction Method Based on Generative Adversarial Network. Ph.D. Thesis, Nanjing University of Posts and Telecommunications, Nanjing, China, 2022.
28. Liang, J.; Jiang, L.; Murphy, K.; Yu, T.; Hauptmann, A. The Garden of Forking Paths: Towards Multi-Future Trajectory Prediction. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
29. Oh, S.; Hoogs, A.; Perera, A.; Cuntoor, N.; Chen, C.C.; Lee, J.T.; Mukherjee, S.; Aggarwal, J.K.; Lee, H.; Davis, L.; et al. A large-scale benchmark dataset for event recognition in surveillance video. In Proceedings of the CVPR, Colorado Springs, CO, USA, 20–25 June 2011.