Social NSTransformers: Low-Quality Pedestrian Trajectory Prediction

Zihan Jiang, *Student member, IEEE*, Yiqun Ma, Bingyu Shi, *Student member, IEEE*, Xin Lu, Jian Xing, *Member, IEEE*, Nuno Gonçalves, *Member, IEEE* and Bo Jin, *Member, IEEE*

***Abstract*—This paper introduces a novel model for low-quality pedestrian trajectory prediction, the Social Non-stationary Transformers (NSTransformers), that merges the strengths of NSTransformers and Spatio-Temporal graph transformer (STAR). The model can capture social interaction cues among pedestrians and integrate features across spatial and temporal dimensions to enhance the precision and resilience of trajectory predictions. We also propose an enhanced loss function that combines diversity loss with logarithmic root mean squared error (log-RMSE) to guarantee the reasonableness and diversity of the generated trajectories. This design adapts well to complex pedestrian interaction scenarios, thereby improving the reliability and accuracy of trajectory prediction. Furthermore, we integrate a Generative Adversarial Network (GAN) to model the randomness inherent in pedestrian trajectories. Compared to the conventional standard Gaussian distribution, our GAN approach better simulates the intricate distribution found in pedestrian trajectories, enhancing the trajectory prediction's diversity and robustness. Experimental results reveal that our model outperforms several state-of-the-art methods. This research opens the avenue for future exploration in low-quality pedestrian trajectory prediction.**

***Impact Statement*—Pedestrian trajectory prediction is a common technique in autonomous driving, video surveillance, etc. Highly accurate pedestrian trajectory prediction can make the related fields work better. However, there needs to be more research on low-quality (bad environment) pedestrian trajectory prediction, but the low-quality state does occur frequently in daily life. This paper investigates this issue and proposes a new model for pedestrian trajectory prediction in the low-quality state. Finally, we experimentally demonstrate that the performance of our model can be improved by more than 60% in the standard environment and more than 40% in the low-quality environment.**

***Index Terms*—Pedestrian trajectory prediction, NSTransformers, GAN, Enhanced loss function**

I. Introduction

LOW-QUALITY vision is an essential direction in computer vision research [1, 2, 3, 4, 5, 6]. In addition, similar research situations exist in other fields, such as robust control [7, 8, 9, 10] in the control field. All these research studies try to realize the effect of a typical environment in a complex environment. However, according to the literature research [11, 12, 13], more research must be needed on pedestrian trajectory prediction under low-quality states. However, pedestrian trajectories under low-quality states often need help with prediction. Therefore, this paper conducts a study on predicting pedestrian trajectories under a low-quality state.

The pedestrian trajectory prediction task is a task to predict the future motion trajectory of pedestrians, which is mainly used in the fields of autonomous driving [14, 15, 16], robot [17, 18, 19] and video surveillance [20, 21, 22]. Due to the importance of this task, the pedestrian trajectory prediction task has attracted many researchers to study it. Currently pedestrian trajectory prediction is usually defined as a sequence generation task, i.e., given a sequence of past pedestrian trajectories, to generate a sequence of future pedestrian trajectories. The complete flow of this task is shown in Figure 1.

Pedestrian trajectory prediction can currently be categorized into two types: standard pedestrian trajectory prediction and low-quality pedestrian trajectory prediction. There are three types of standard pedestrian trajectory prediction methods:

(1) physics-based methods [23, 24, 25], such methods typically rely on physical rules and pedestrian dynamics models to predict pedestrian trajectories. This approach emphasizes the interactions between individuals and the relationship between individuals and their environment as a basis for understanding pedestrian behavior. While physically based approaches have had some success in simulating pedestrian behavior in simple scenarios, they often struggle to capture the diversity and uncertainty of pedestrian behavior in more complex scenarios;

(2) machine learning methods [26, 27, 28], such methods rely on feature engineering and statistical models to understand the underlying patterns of pedestrian behavior. While these methods perform well when dealing with datasets with well-defined patterns, they may not be sufficient to deal with highly complex and dynamically changing real-world scenarios;

(3) deep learning methods [29, 30, 31, 32], these methods can capture complex dependencies in time-series data [49, 50, 51, 52], effectively handle interactions between pedestrians, and improve prediction accuracy by learning pedestrian interactions in the social space. The advantage of deep learning methods is that they can automatically learn features from data without manually designing a complex feature extraction process, thus better adapting to complex and changing prediction scenarios.

Researchers have made many successful and influential con-

This work is not supported by any funding.

Zihan Jiang, Yiqun Ma and Xin Lv are at the School of Advanced Technology (SAT), Xijiao Liverpool University, Suzhou 215000, China. (e-mail: 1095773538@qq.com; Yiqun.Ma22@xjtlu.edu.cn; Xin.Lu22@student.xjtlu.edu.cn)

Bingyu Shi and Jian Xing are with the College of Computer and Control Engineering, Northeast Forestry University, Harbin 150000, China. (e-mail: 2656557369sby@nefu.edu.cn; xj@nefu.edu.cn)

Nuno Gonçalves and Bo Jin are with Institute of Systems and Robotics, Department of Electrical and Computer Engineering, University of Coimbra, Coimbra 3030-290, Portugal (e-mail: nunogon@deec.uc.pt; jin.bo@isr.uc.pt).

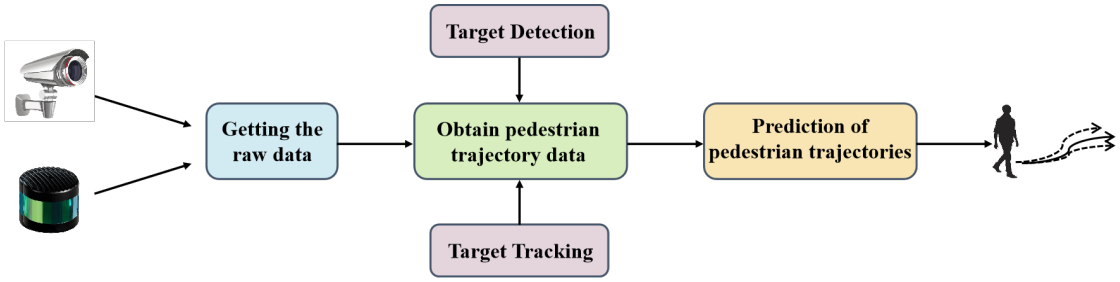

Fig. 1: Complete flow of pedestrian trajectory prediction

tributions to standard pedestrian trajectory prediction. However, standard pedestrian trajectory prediction has its limitations. Although these methods work well in conventional environments, they tend to lose their effectiveness when the environment tends to be more complex. In real-life environments, the critical step of acquiring pedestrian trajectories is often unavoidably affected by various interferences in the acquired pedestrian trajectories.

To address these limitations, low-quality pedestrian trajectory prediction is necessary. Therefore, this paper proposes a new model, social Non-stationary Transformers (NSTransformers), which is based on NSTransformers [33], and uses Spatio-Temporal graph transformer (STAR) [34] as the social interaction layer to extract the interaction between pedestrians and then improves the original loss function by combining variety loss with logarithmic root mean squared error (log-RMSE) to ensure the reasonableness of the generated trajectories. Finally, the generative adversarial network(GAN) [35] is used instead of the original standard Gaussian distribution to model the randomness in the pedestrian trajectories.

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

(1) A new social NSTransformers model based on NSTransformers and STAR is proposed for low-quality pedestrian trajectory prediction. The model can extract the social interaction information between pedestrians and integrate the features in spatial and temporal dimensions to improve the accuracy and robustness of trajectory prediction;

(2) The original loss function is improved by combining the variety loss with the logarithmic root mean squared error (log-RMSE) to ensure the reasonableness and diversity of the generated trajectories. This loss function design can better adapt to complex pedestrian interaction scenarios and improve the accuracy and reliability of trajectory prediction;

(3) The GAN is introduced to model the randomness in pedestrian trajectories. Compared to the original standard Gaussian distribution, GAN can better simulate the complex distribution in pedestrian trajectories, thus improving the diversity and robustness of trajectory prediction.

The rest of the paper is organized as follows. Section II introduces relevant prior knowledge in pedestrian trajectory prediction. In Section III, we present the proposed Social NSTransformers model. Section IV analyzes the experimental results of our proposed model and compares it with several state-of-the-art methods. Finally, in Section V, we provide the conclusions of our study and discuss future research directions in low-quality pedestrian trajectory prediction.

II. Priori knowledge

A. Problem setup

This paper defines the pedestrian trajectory prediction task as a sequence-to-sequence temporal prediction task. This task requires that given a sequence $X = \{x_1, x_2, \ldots, x_t\}$, where x_t represents the pedestrian location and velocity information at historical moment t. The goal is to predict a future trajectory sequence $Y = \{y_1, y_2, \ldots, y_t\}$ containing pedestrians.

At each time t, this paper argues that have a set of N pedestrians $\left\{p_t^i\right\}_{i=1}^N$, where $p_t^i = \left(x_t^i, y_t^i\right)$ denotes the position of the pedestrian in a top view map. At the same time, this paper assumes that the pedestrian pairs $\left(p_t^i, p_t^j\right)$ with a distance less than d would have an undirected edge (i, j). This leads to an interaction graph at each time step $t : G_t = (V_t, E_t)$, where $V_t = \left\{p_t^i\right\}_{i=1}^N$ and $E_t = \{(i, j) \mid i, j$ is connected at time $t\}$. For each node, i at time t, this paper defines the neighbor of the node set as $Nb(i, t)$, where for each node $j \in Nb(i, t), e_t(i, j) \in E_t$.

B. Low-quality pedestrian trajectory prediction problem definition

In the real world, pedestrian trajectory prediction often requires computer vision assistance. Pedestrian trajectories, denoted as $\{(\hat{x}_t, \hat{y}_t)\}_{t=1}^T$, must first be acquired from complex environments by computer vision techniques. These observations are affected by various factors, such as environmental noise, image processing errors, etc. The noise model can represent as (1) and (2).

$$\hat{x}_t = x_t + n_{x,t} \tag{1}$$

$$\hat{y}_t = y_t + n_{y,t} \tag{2}$$

where x_t, y_t is the actual trajectory point, and $n_{x,t}, n_{y,t}$ is the noise term. After obtaining the pedestrian trajectory, a model is needed to predict the pedestrian trajectory, and the randomness of the pedestrian trajectory is more difficult to simulate because the obtained trajectory is already disturbed by noise. In this case, predicting the pedestrian trajectory can be represented as (3).

$$\hat{P}_{\text{future}} = F\left(P_{\text{past}}, \Theta, \mathcal{E}\right) \tag{3}$$

where Θ represents the pedestrian's exposure to environmental influences and noise interference, and $\mathcal{E}$ represents the pedestrian's randomness. While $\hat{P}_{\text{future}}$ and P_{past} denote the future trajectories to be predicted and the historical trajectories that have been observed.

C. NSTransformers

Non-stationary Transformers is a general framework for time series forecasting designed to address the problem of non-smoothness and over-smoothing of time series data. The framework contains two interdependent modules: sequence smoothing and de-smoothing attention. The sequence smoothing module unifies the critical statistics of each input sequence through a normalization strategy. It transforms the output into a form that recovers the statistics to improve the predictability of the sequence. To address the over-smoothing problem, the de-smoothing attention module recovers intrinsic non-smoothness information by approximating the distinguishable attention learned from the original sequence. The structure of the NSTransformers model is schematically shown in Figure 2.

III. Social NSTransformer

In the NSTransformers model, a sequence smoothing module is included to enhance the smoothness of the input data, and a de-smoothing attention module to reintegrate non-smooth information into the time-dependent modeling within the model, thus alleviating the over-smoothing problem. Sequence smoothing consists of two phases: window normalization and denormalization. The NSTransformers model achieves this by dynamically adjusting the number of layers and the attention patterns of the Transformer network, based on the local properties of the time series. Specifically, the model incorporates a gating mechanism that controls the flow of information between different layers of the Transformer network, and a set of adaptive attention mechanisms that allow the model to selectively attend to relevant temporal patterns in the input data.

A. STAR social interaction module

In this paper, we use the encoder layer of the STAR model [51] as the social interaction module of the social NSTransformer. The structure of the STAR social interaction module is shown in Figure 3.

In Figure 3, the STAR module is divided into three parts: (1) spatial transformer, (2) temporal transformer, and (3) graph memory.

1) Spatial transformer: The spatial transformer block extracts the spatial interaction among pedestrians. STAR model proposes a novel transformer-based graph convolution. TGConv is used for message passing on a graph. In this TGConv, the self-attentive mechanism can be viewed as a message passing on an undirected, fully connected graph. In the feature extraction process, it can be represented by (4)-(6).

$$q_i = f_Q\left(h_i\right) \tag{4}$$

$$k_i = f_K\left(h_i\right) \tag{5}$$

$$v_i = f_V\left(h_i\right) \tag{6}$$

In (4)-(6), h_i is defined as a feature. Therefore, this paper defines the message from node j to i in the fully connected graph as (7).

$$m^{j \to i} = q_i^{\mathrm{T}} k_j \tag{7}$$

Combining the above definitions, the formula of the attention mechanism can be shown in (8).

$$\operatorname{Att}(Q, K, V) = \frac{\operatorname{Softmax}\left(\left[m^{j \to i}\right]_{i,j=1:n}\right)}{\sqrt{d_k}} \left[v_i\right]_{i=1}^{n} \tag{8}$$

The graph convolution operation for node i is written as (9) and (10).

$$Att(i) = \frac{\operatorname{Softmax}\left(\left[m^{j \to i}\right]_{j \in Nb(i) \cup \{i\}}\right)}{\sqrt{d_k}} \left[v_j\right]_{j \in Nb(i) \cup \{i\}}^{\mathrm{T}} + h_i \tag{9}$$

$$h_i' = f_{out}(Att(i)) + Att(i) \tag{10}$$

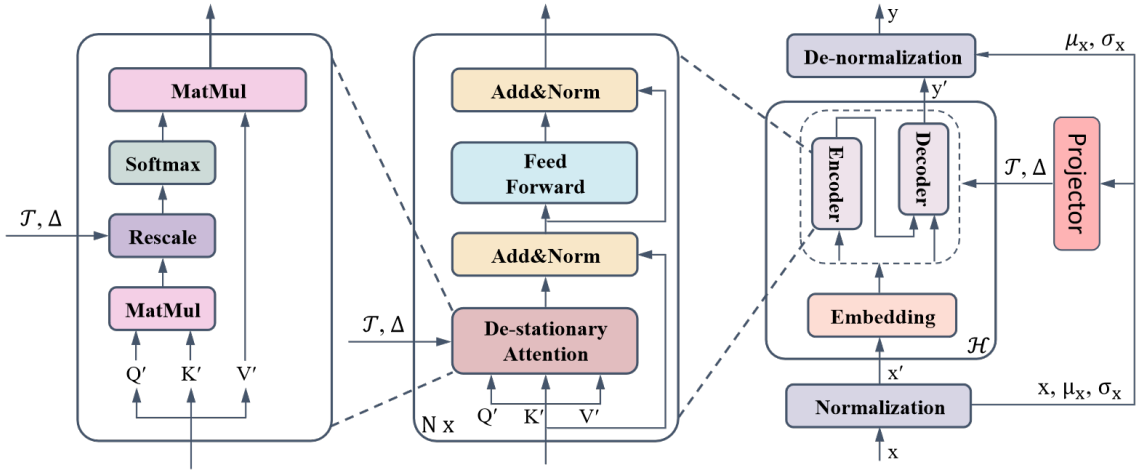

Fig. 2: The NSTransformers model structure diagram

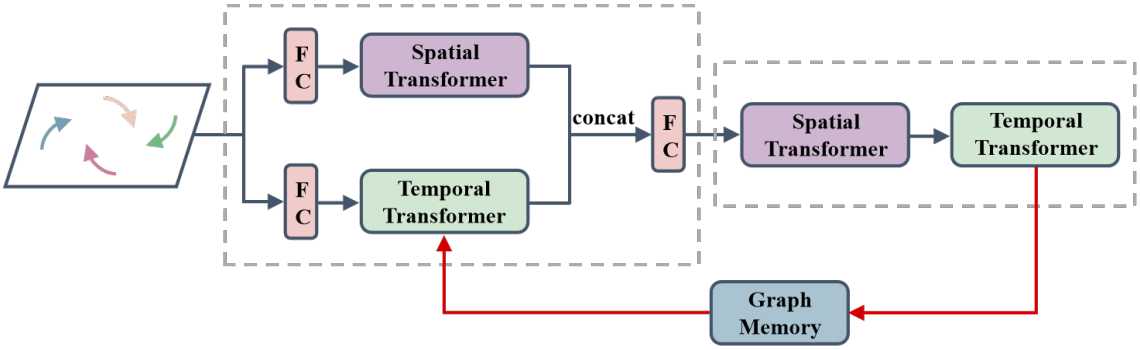

Fig. 3: STAR social interaction module structure diagram

2) Temporal transformer: In the temporal transformer, it is first necessary to input pedestrian information in the form of $\{h_1^i\}_{i=1}^N, \{h_2^i\}_{i=1}^N, \{h_3^i\}_{i=1}^N, \ldots, \{h_n^i\}_{i=1}^N$ and then output a set of $\{h_1^i\}_{i=1}^N, \{h_2^i\}_{i=1}^N, \{h_3^i\}_{i=1}^N, \ldots, \{h_n^{ii}\}_{i=1}^N$ with temporal relations. In this process, each pedestrian is considered an independent individual. The temporal transformer can be calculated as (11)-(13).

$$\mathrm{Att}\left(Q^i, K^i, V^i\right) = \frac{\mathrm{Softmax}\left(Q^i K^{iT}\right)}{\sqrt{d_k}} V^i \tag{11}$$

$$\mathrm{MultiHead}\left(Q^i, K^i, V^i\right) = f_o\left(\left[\text{ head }_j\right]_{j=1}^k\right) \tag{12}$$

$$\mathrm{head}_j = \mathrm{Att}\left(Q^i, K^i, V^i\right) \tag{13}$$

where f_0 represents a fully connected layer, which includes k heads.

3) Graph memory: While Transformer networks have shown remarkable performance in long-horizon sequence modeling through their self-attention mechanism, they may face challenges when handling continuous time-series data requiring temporal solid consistency. This is particularly important for trajectory prediction, as the positions of pedestrians typically do not change abruptly over short periods. Therefore, ensuring temporal consistency is a strict requirement for accurate trajectory prediction. This paper uses a graph memory structure to solve this problem.

First, for each moment t, suppose there are n pedestrians in the scene, and their historical trajectory feature vectors can be expressed as 14.

$$h_1^t, h_2^t, \ldots, h_n^t \tag{14}$$

In (14), h_i^t is the historical trajectory feature vector of the ith pedestrian at moment t.

Then, for each moment t, assuming that there are m pedestrians interacting with each other, their interaction feature vectors can be expressed as 15.

$$e_1^t, e_2^t, \ldots, e_m^t \tag{15}$$

In (15), e_j^t is the feature vector of the jth interaction at moment t. Next, we store these historical trajectory and interaction feature vectors in Graph Memory. Specifically, we store them in each of the $n+m$ learnable storage slots, denoted as:

$$m_1, m_2, \ldots, m_{n+m} \tag{16}$$

In (16), m_i is the ith storage slot corresponding to the ith pedestrian or interaction history trajectory or interaction feature vector.

After the storage is completed, we use the graph attention mechanism to retrieve the historical information stored in the graph memory. Specifically, for each pedestrian i and moment t, we use a set of weight vectors w_i^t to compute a weighted average of the historical trajectory and interaction feature vectors associated with that pedestrian at moment t. This weighted average is the input feature vector for pedestrian i at moment t. The calculation formula is as follows:

$$h_i^t = \sum_{j=1}^{n+m} \alpha_{i,j}^t m_j \tag{17}$$

In (17), $\alpha_{i,j}^t$ is the attention weight of the ith pedestrian between moment t and the jth storage slot, which is calculated by the graph attention mechanism as follows:

$$\alpha_{i,j}^t = \frac{\exp\left(e_{i,j}^t\right)}{\sum_{k=1}^{n+m} \exp\left(e_{i,k}^t\right)} \tag{18}$$

In (18), $e_{i,j}^t$ is the attentional energy of the ith pedestrian between moment t and the jth storage slot, which the following equation can calculate:

$$e_{i,j}^t = a^T \cdot \mathrm{ReLU}\left(W_h h_i^t + W_m m_j + b\right) \tag{19}$$

In (19), W_h and W_m are learnable weight matrices, b is the bias vector, and a is the learnable attention vector. The ReLU denotes the modified linear unit function.

B. The improved variety loss

In the normal walking of pedestrians, there are various possibilities of trajectories because of the influence of several factors. In order to simulate the randomness in pedestrian trajectories, it is vital to generate multiple trajectories reasonably. This paper will start this section from this point and utilize an improved variety loss to generate diversity trajectories.

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2024.3421175

In traditional trajectory prediction methods, the L2 loss function is usually used to generate trajectories for pedestrian trajectory prediction. The L2 loss function calculates the difference between the generated and true trajectories. the expression of L2 loss function is shown in (20).

$$L2 = \sum_{i=1}^{n} \left(y_i - f\left(x_i\right)\right)^2 \tag{20}$$

In (20), y_i is the true value and $f\left(x_i\right)$ is the predicted value. The prediction result predicted by the L2 loss function is usually the average of all possible trajectories, and the prediction effect is shown in Figure 4.

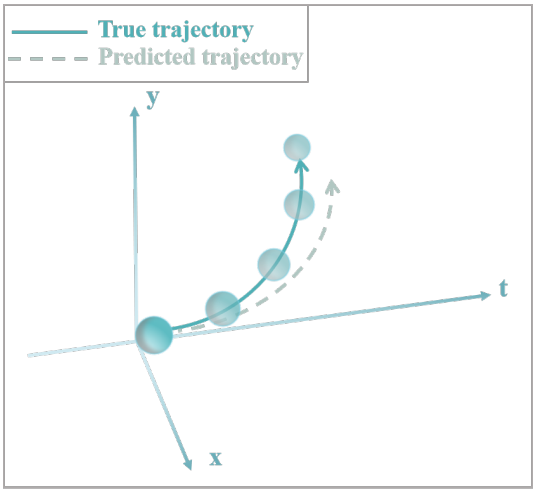

Fig. 4: The single trajectory prediction diagram

However, this single trajectory prediction is limited in that it does not simulate the randomness in the pedestrian motion process. This paper introduces variety loss to solve this problem based on the L2 loss function. The mathematical expression of variety loss is shown in (21).

$$L_{\text{variety}} = \min_{k} \left\| Y_i - \hat{Y}_i^k \right\|_2 \tag{21}$$

where k is a hyperparameter; in variety loss, the best trajectory is selected by generating multiple trajectories, which can prevent the final results from being averaged and obtain accurate trajectory prediction results. Figure 5 shows the prediction results after introducing variety loss.

Although variety loss can encourage neural networks to generate diverse outputs, there are still limitations, such as the possibility of generating irrational results and overfitting due to the network's excessive focus on generating various results. Therefore, we propose an improved variety loss for these limitations in this paper, as shown in (22).

$$L_{var+MSE} = L_{var} + \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\log y_i - \log \hat{y}_i\right)^2} \tag{22}$$

In (22), n denotes the number of samples, y_i represents the real target value for the i sample, and $\hat{y}_i$ means the predicted target value for the i sample. Compared to the original variety

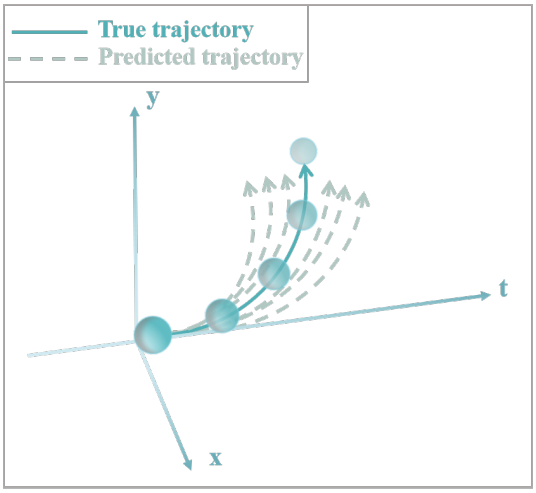

Fig. 5: The multi-trajectory prediction diagram

loss, the variety loss after incorporating Log MSE has the following two advantages.

1) Better focus on the authenticity of trajectories:
The original variety loss encourages the network to generate diverse trajectories by minimizing the similarity between trajectories. However, this approach may cause the network to generate some trajectories that do not match the situation. Adding Log MSE as a part of the loss function can help the network better focus on the authenticity of the trajectories and thus avoid generating unreasonable trajectories.
2) Improved prediction accuracy of the model:
By adding Log MSE, the loss function focuses more on the accuracy of the network's prediction and, therefore, can help the network learn the patterns of the trajectories better and improve the prediction accuracy of the model. The variety loss, on the other hand, can encourage the network to generate diverse trajectories and avoid overfitting the model, thus further improving the model's generalization ability and prediction accuracy.

C. Generative adversarial networks

Standard Gaussian distributions are often used in pedestrian trajectory prediction to model the stochastic nature of pedestrian trajectories, but fixed probability distributions often lead to weak generalization of the model. When the pedestrian trajectory is in a low-quality state, the pedestrian trajectory at this time is often not a standard Gaussian distribution or even a Gaussian distribution. Therefore, this paper uses the generative adversarial network to replace the original probability distribution, using the characteristics of the generative adversarial network to simulate the probability distribution in different situations to achieve a more reasonable simulation of pedestrian randomness for this purpose. The computational formula for generating the adversarial network is shown in (23).

$$\begin{aligned}\min_{G}\max_{D} V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] \\ +\mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1-D(G(\boldsymbol{z})))]\end{aligned} \tag{23}$$

In generative adversarial networks, the role of the generator is to generate a probability distribution that characterizes the randomness of pedestrian trajectories. Based on the pre-trained model, the discriminator receives this probability distribution and determines whether the trajectory vectors are from a real pedestrian trajectory dataset. The generator and the discriminator continuously carry out adversarial learning, the probability distribution generated by the generator will be gradually close to the distribution of the real pedestrian trajectory dataset, and the discriminator will also continuously improve its judgment ability, making the probability distribution generated by the generator more challenging to be identified by the discriminator.

In this process, the generator continuously adjusts its parameters to make the generated probability distribution more likely to be recognized by the discriminator as the accurate pedestrian trajectory distribution. The discriminator is also updated according to the probability distribution generated by the generator, allowing it to better distinguish between the accurate pedestrian trajectory distribution and the generator's generated probability distribution.

In each iteration, the probability distribution generated by the generator is fed into a loss function, which calculates the gap between the generated probability distribution and the accurate pedestrian trajectory distribution, i.e., the improved variety loss, and the generator back-propagates according to the result of this loss function and adjusts its parameters to make the generated probability distribution closer to the accurate pedestrian trajectory distribution. This process is repeated until the generated probability distribution is consistent with the distribution of the real pedestrian trajectory dataset. The structure of GAN is schematically shown in Figure 6.

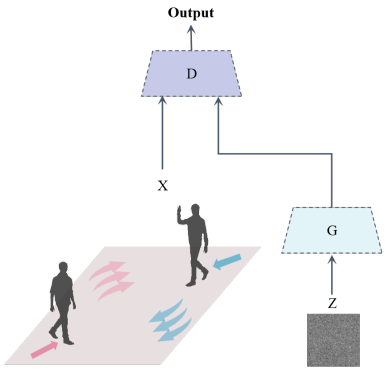

Fig. 6: The structure of GAN

IV. Experiments and Analysis

A. Experimental setup and evaluation indicators

In this paper, experiments are carried out under the hardware environment of Windows 11, i7-11800H, and NVIDIA GeForce RTX 3080, and the software environment used is Python 3.8. Table I shows the hyperparameter settings involved.

By referring to the previous work, the evaluation metrics in this paper are Average Displacement Error (ADE) and Final Displacement Error (FDE). These evaluation metrics are shown in (24) and (25).

TABLE I: Hyperparameter settings

Index	Value
Learning rate	0.0015
Epochs	300
Batch size	4
Observe time	3.2s(8 frames)
Predict time	4.8s(12 frames)
Sampling number	20

$$ADE = \frac{\sum_{n=1}^{N}\sum_{t=t_{\text{obs}}+1}^{t_{\text{pre}}}\left\|Y_n^t - \hat{Y}_n^t\right\|_2}{N * (t_{\text{pre}} - t_{\text{obs}} - 1)} \tag{24}$$

$$FDE = \frac{\sum_{n=1}^{N}\left\|Y_n^{t_{\text{pre}}} - \hat{Y}_n^{t_{\text{pre}}}\right\|_2}{N} \tag{25}$$

In equations (24) and (25), Y_n^t and $\hat{Y}_n^t$ denote the actual and forecasted paths of pedestrian n at time t, respectively. Here, N signifies the present total count of pedestrians.

B. Dataset introduction and preparation

The dataset used in this paper has three parts: (1) the standard pedestrian trajectory dataset, (2) the self-made low-quality pedestrian trajectory dataset, and (3) the self-made animal dataset.

1) Standard pedestrian trajectory dataset: In the standard pedestrian trajectory dataset, two typical pedestrian trajectory public datasets, ETH and UCY, are mainly applied. Humans manually label these datasets so the trajectory information is accurate. The ETH data set is a tilted view of a busy square taken by a still camera with a total of 450 subjects in 5400 frames. Most of the pedestrians stayed on camera for no more than 15 seconds, while others talked to others or waited in place. The UCY data set takes photos of pedestrians in public spaces from the top view, and there are rich multi-person interaction scenes in this data set. The scenes were shot in unrestrained environments, so there were few obstacles to prevent pedestrians from moving. This type of dataset is shown in Figure 7.

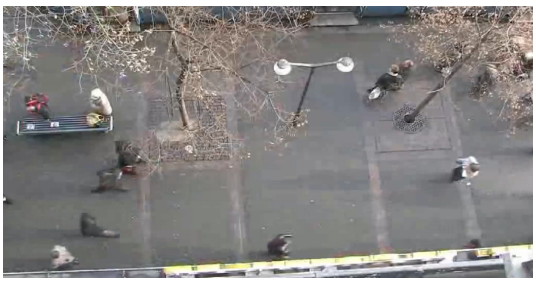

Fig. 7: The standard pedestrian trajectory dataset schematic

The standard pedestrian trajectory prediction dataset production process is shown in Figure 8.

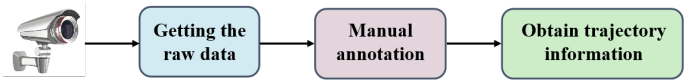

Fig. 8: The standard pedestrian trajectory prediction dataset production process

In Figure 8, the raw pedestrian video data is first acquired via a surveillance camera. Next, the dataset producer performs manual annotation. This step involves labeling the position of each pedestrian in the video frame by frame to capture their movement trajectories accurately. Finally, the researchers could extract detailed trajectory information through these annotations, which were subsequently used to train and validate the pedestrian trajectory prediction algorithm.

2) Low-quality pedestrian trajectory dataset: In the self-made low-quality pedestrian trajectory dataset, the self-made pedestrian trajectory dataset is mainly based on two pedestrian trajectory public datasets, ETH and UCY. The ETH and UCY datasets provide video information in addition to pedestrian information. In this paper, the videos of the ETH and UCY datasets are first fuzzily processed, then YOLOX+deepsort is utilized to mark the pedestrian trajectory information, and finally, the homemade low-quality pedestrian trajectory dataset is obtained. This type of dataset is shown in Figure 9.

Fig. 9: The low-quality pedestrian trajectory dataset schematic

The low-quality dataset production process is shown in Figure 10.

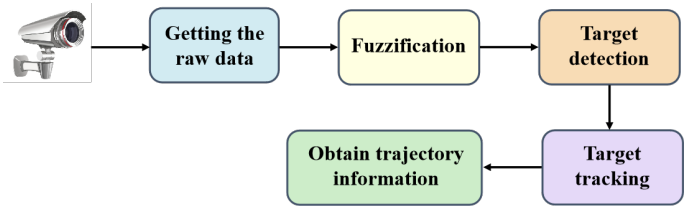

Fig. 10: The low-quality pedestrian trajectory prediction dataset production process

The four critical steps of the low-quality pedestrian trajectory prediction dataset production process are illustrated in Figure 10: Firstly, the raw video data is acquired, followed by a unique blurring process to simulate low-quality environments, followed by target detection, which identifies pedestrians in the video, and finally, target tracking, which is carried out to construct the pedestrians' motion trajectories. The main differences between this process and the production of a standard pedestrian trajectory prediction dataset are the additional blurring step and the clear distinction between target detection and tracking as separate stages, which simulate the low-quality situations that may occur in natural environments.

The critical components in Figure 9 are fuzzification, target detection, and target tracking. These three components are described in more detail below.

(*a*) *Fuzzification*

In order to test the performance of the pedestrian trajectory prediction model in the low-quality state, this paper decides to produce its dataset of pedestrian trajectories in the low-quality state. However, obtaining the low-quality pedestrian trajectory prediction dataset in the natural environment is complex. Hence, this paper chooses to prepare the low-quality state pedestrian trajectory dataset by blurring on the ETH and UCY datasets and then using YOLOX+deepsort to obtain the pedestrian trajectory. The blurring in this process mainly uses adjusting the resolution, adjusting the original resolution of 1900*1000 to 400*210.

(*b*) *Target detection*

In this paper, we utilize the YOLOX [37] algorithm for target detection based on the convolutional neural network target detection method. The approach uses an anchor-free method for target detection and employs new techniques to improve detection accuracy and speed.

YOLOX consists of a lightweight network structure called YOLOX-Nano and two backbone networks, YOLOX-L and YOLOX-XL. These network structures use the Cross-Stage Partial Network (CSPNet) structure, which enhances the expressive power of the model.

Targets are detected using dense feature map sampling instead of predefined anchor boxes. The detection uses a YOLOXHead, which generates candidate boxes and category probabilities. The YOLOXHead samples densely on the feature graph, generates numerous candidate boxes, and filters out the final target boxes using category probabilities and box confidence levels.

To improve the detector's ability to detect objects at different scales, YOLOX employs the Spatial Pyramid Pooling (SPP) structure, which pools feature maps at different scales.

Additionally, YOLOX uses DropBlock regularization to prevent overfitting and improve the model's generalization ability. Overall, the YOLOX algorithm incorporates several innovative techniques to enhance target detection performance, particularly with its anchor-free approach and SPP structure, making it a successful method in target detection. The algorithmic structure of YOLOX is shown in Figure 13.

(*c*) *Target tracking*

The primary algorithm used for target tracking is the deepsort [38] algorithm, which builds upon the sort algorithm. While sorting is simple and fast, it has a problem with reassigning IDs when an object disappears and reappears due to occlusion. Additionally, the sort's matching method only considers the distance between boxes, which can lead to some issues.

To address these shortcomings, deepsort adds a new layer on top of the sort to confirm new trajectories. These trajectories

are classified into two states: Confirmed and Unconfirmed. When a new trajectory is generated, it is labeled as Unconfirmed, and its target is also Unconfirmed. Only after successfully matching to the detected response for three consecutive times will the target transition from Unlabeled to Unconfirmed. Once a target is in the Confirmed state, it will be deleted after 30 consecutive mismatches with the detected response. The entire process of the DeepSort algorithm is tracked, observed, and improved, as shown in Figure 14.

3) Animal trajectory dataset: Since the pedestrian trajectory dataset is complicated and exhibits extreme randomness no matter how it is processed, in this paper, the animal dataset is self-made to test the model's performance in a highly random environment. In this paper, we collected moving videos of animals indoors through the self-built robotics platform. The animal trajectory dataset is shown in Figure 11.

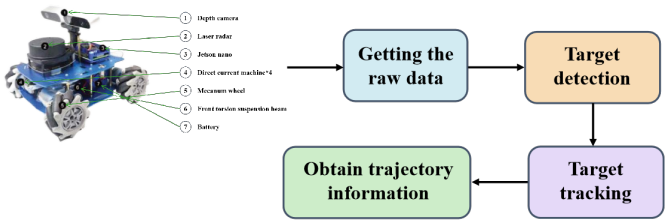

Fig. 11: The animal trajectory prediction dataset production process

Fig. 12: The animal trajectory dataset schematic

The target detection and target tracking used in Figure 12 are also YOLOX and deepsort. Figure 12 shows the schematic diagram of the animal's moving trajectory.

C. Standard pedestrian trajectory prediction experimental results and discussion

In the standard pedestrian trajectory prediction experiments, in this section, experiments are conducted on five subdatasets, ETH, HOTEL, UNIV, ZARA1, and ZARA2, from two datasets, ETH and UCY, and the results of the experiments are compared with numerous state-of-the-art algorithms. The experimental results on the ETH and UCY datasets are shown in Table II.

In Table II, the Social NSTransformers method was evaluated for its performance on several public datasets and compared with several existing mainstream methods. The results show that Social NSTransformers exhibit strong performance on some datasets (especially HOTEL) with an ADE/FDE of 0.35/0.62, indicating the method's effectiveness in dealing with the pedestrian trajectory prediction problem. However, compared to the best-performing methods, such as TUTR, Social NSTransformers have some performance gaps on all datasets, especially on the ETH and UNIV datasets. This suggests that although our method achieves satisfactory results in some aspects, there is still room for improvement in adaptability and prediction accuracy.

In order to verify the effect of different components in models trained on public datasets, ablation experiments are conducted in this section, as shown in Table III.

In our ablation experiments, we meticulously evaluated the different components of the NSTransformers method to understand their impact on the overall performance. The experimental results show that each added component significantly improves the model's performance. The pure NSTransformers method has an average ADE/FDE of 0.54/0.83 on the five datasets. Adding the STAR social interaction module enhances the performance to an average ADE/FDE of 0.49/0.76; introducing variety loss further reduces the average to 0.47/0.76. combining the variety loss and the GAN enhances the performance to 0.44/0.72. Ultimately, the NSTransformers model combining the STAR social interaction module and Variety loss performs the best, achieving an average ADE/FDE of 0.42/0.70. These results demonstrate the importance of each component in improving the accuracy and robustness of the model for pedestrian trajectory prediction. The final Social NSTransformers model performed well on all datasets with an average ADE/FDE of 0.35/0.62, demonstrating the effectiveness of this combined strategy.

D. Low-quality pedestrian trajectory prediction experimental results and discussion

In the self-made low-quality pedestrian trajectory prediction dataset experiment, this paper compares social NSTransformers with other state-of-the-art algorithms, and the experimental results are shown in Table IV.

As seen from Table IV, we conducted experiments against a self-made low-quality pedestrian trajectory dataset to evaluate the performance of the Social NSTransformers method against other methods in the low-quality case. From the experimental results, Social NSTransformers show a significant advantage on all tested datasets, with an average ADE/FDE of 0.61/1.59. This is a significant improvement over the traditional methods, indicating that our method is more efficient in dealing with the trajectory prediction problem in the low-quality case. Social NSTransformers obtain the lowest error rate on almost all datasets compared to other methods. This result emphasizes the robustness and efficiency of our method, especially when dealing with pedestrian trajectories in low-quality environments. Meanwhile, to verify each module's effectiveness, this paper conducts ablation experiments on models trained on low-quality datasets.

As shown in Table V, a series of ablation experiments were conducted for low-quality datasets to evaluate the impact of different components on the performance of the NSTransform-

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2024.3421175

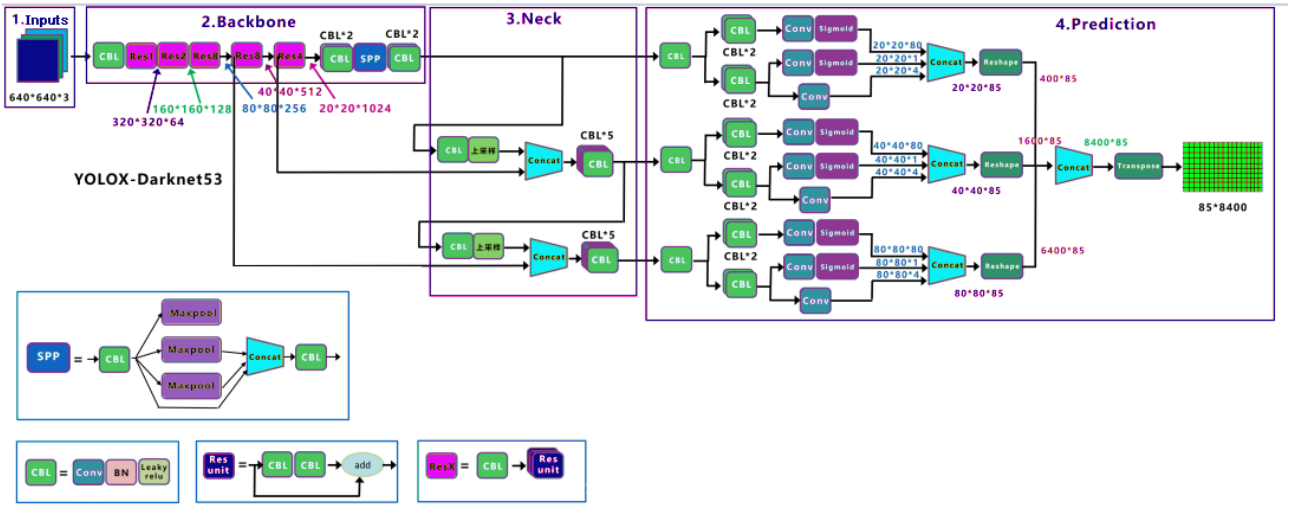

Fig. 13: YOLOX algorithm structure diagram

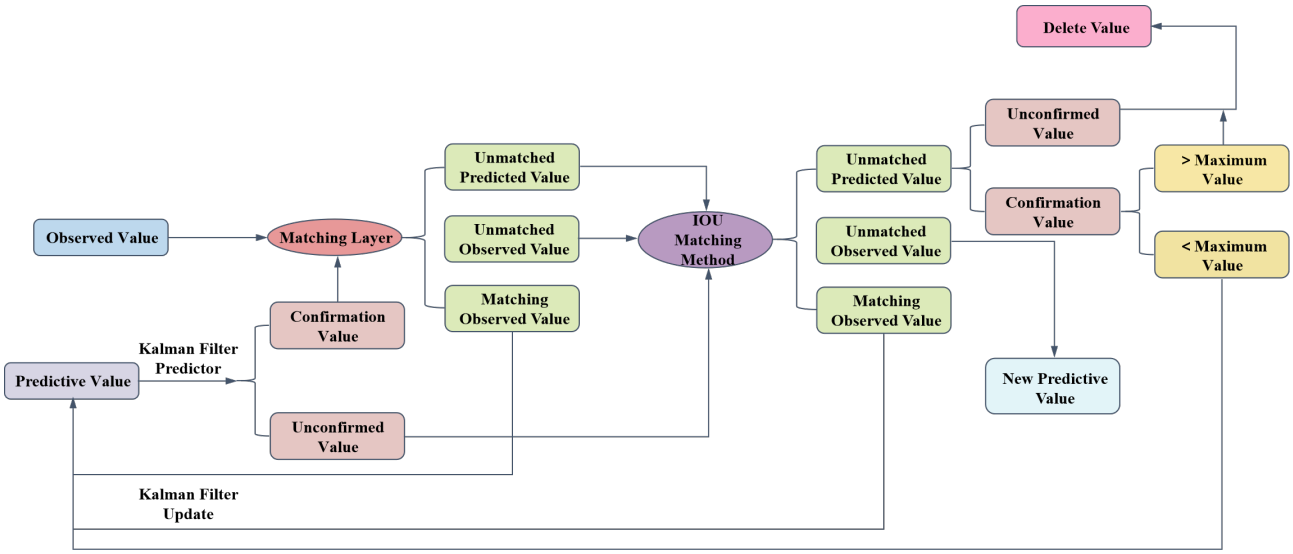

Fig. 14: Flowchart of deepsort algorithm

TABLE II: The experimental results of public dataset

Method (ADE/FDE)	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
Vanilla LSTM[39]	1.09/2.41	0.86/1.91	0.61/1.31	0.41/0.88	0.52/1.11	0.70/1.52
Social LSTM[39]	1.09/2.35	0.79/1.76	0.67/1.40	0.47/1.00	0.56/1.17	0.71/1.53
SGAN[40]	0.87/1.62	0.67/1.37	0.60/1.26	0.34/0.69	0.42/0.84	0.58/1.12
Sophie[41]	0.7/1.43	0.76/1.67	0.54/1.24	0.30/0.63	0.38/0.78	0.54/1.15
GAT[42]	0.68/1.29	0.68/1.40	0.57/1.29	0.29/0.60	0.37/0.75	0.52/1.07
Social BiGAT[42]	0.69/1.29	0.49/1.01	0.55/1.32	0.30/0.62	0.36/0.75	0.48/1.00
Social STGCNN[43]	0.64/1.11	0.49/0.85	0.44/0.79	0.34/0.53	0.30/0.48	0.44/0.75
RSGB[44]	0.8/1.53	0.33/0.64	0.59/1.25	0.40/0.86	0.30/0.65	0.48/0.99
RTN[45]	0.69/1.24	0.43/0.87	0.53/1.17	0.28/0.61	0.28/0.59	0.44/0.90
STAR[34]	0.36/0.65	0.17/0.36	0.26/0.55	0.22/0.46	0.31/0.62	0.26/0.53
E-SR-LSTM[46]	0.44/0.79	0.19/0.31	0.32/0.64	0.27/0.54	0.50/1.05	0.34/0.67
TUTR[47]	0.40/0.61	0.11/0.18	0.23/0.42	0.18/0.34	0.13/0.25	0.21/0.36
STAGP[48]	0.65/1.21	0.41/0.73	0.38/0.68	0.28/0.46	0.25/0.44	0.40/0.70
Social NSTransformers	0.40/0.71	0.29/0.47	0.39/0.73	0.34/0.62	0.31/0.57	0.35/0.62

TABLE III: Results of ablation experiments on public datasets

Method (ADE/FDE)	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
NSTransformers	0.61/0.99	0.53/0.75	0.57/0.91	0.53/0.78	0.45/0.71	0.54/0.83
NSTransformers+STAR social interaction module	0.52/0.91	0.45/0.68	0.50/0.85	0.48/0.72	0.39/0.66	0.49/0.76
NSTransformers+Variety loss	0.53/0.90	0.43/0.65	0.52/0.84	0.47/0.72	0.40/0.67	0.47/0.76
NSTransformers+Variety loss+GAN	0.50/0.86	0.41/0.61	0.48/0.81	0.43/0.70	0.38/0.64	0.44/0.72
NSTransformers+STAR social interaction module+Variety loss	0.48/0.81	0.37/0.58	0.47/0.79	0.41/0.69	0.35/0.63	0.42/0.70
Social NSTransformers	0.40/0.71	0.29/0.47	0.39/0.73	0.34/0.62	0.31/0.57	0.35/0.62

TABLE IV: The experimental results of self-made low-quality pedestrain trajectory dataset

Method (ADE/FDE)	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
Vanilla LSTM[39]	2.53/4.67	1.69/3.87	1.29/2.78	1.07/2.21	1.01/2.06	1.52/3.12
Social LSTM[39]	2.41/4.47	1.62/3.62	1.19/2.56	1.01/2.04	0.92/1.92	1.43/2.92
SGAN[40]	2.21/3.56	1.24/3.25	1.09/2.31	0.87/1.67	0.81/1.53	1.24/2.46
Sophie[41]	2.09/3.21	1.17/3.04	0.99/2.13	0.73/1.43	0.72/1.42	1.14/2.24
GAT[42]	1.89/3.07	1.08/2.93	0.85/1.99	0.65/1.33	0.64/1.31	1.00/2.12
Social BiGAT[42]	1.89/3.07	0.97/2.71	0.80/2.04	0.61/1.33	0.62/1.29	0.98/2.01
Social STGCNN[43]	1.53/2.76	0.82/2.41	0.62/1.81	0.49/1.14	0.45/1.05	0.72/1.83
RSGB[44]	2.13/3.29	1.28/3.11	1.04/2.17	0.81/1.51	0.79/1.49	1.21/2.31
RTN[45]	1.57/2.81	0.84/2.41	0.63/1.79	0.57/1.21	0.49/1.11	0.82/1.87
STAR[34]	1.33/2.67	0.79/2.21	0.59/1.63	0.51/1.15	0.45/1.08	0.73/1.75
E-SR-LSTM[46]	1.43/2.79	0.85/2.36	0.67/1.78	0.65/1.31	0.67/1.49	0.85/1.95
TUTR[47]	1.01/2.23	0.55/1.78	0.49/1.42	0.35/0.88	0.29/0.80	0.54/1.42
STAGP[48]	1.37/2.69	0.79/2.41	0.63/1.65	0.59/1.26	0.53/1.24	0.78/1.85
Social NSTransformers	1.12/2.43	0.65/1.99	0.51/1.48	0.40/1.06	0.38/0.97	0.61/1.59

TABLE V: Results of ablation experiments on low-quality datasets

Method (ADE/FDE)	ETH	HOTEL	UNIV	ZARA1	ZARA2	Average
NSTransformers	1.43/2.72	0.98/2.27	0.81/2.01	0.70/1.33	0.69/1.34	0.92/1.93
NSTransformers+STAR social interaction module	1.29/2.61	0.81/2.21	0.75/1.93	0.57/1.26	0.57/1.27	0.80/1.86
NSTransformers+Variety loss	1.23/2.50	0.75/2.14	0.69/1.85	0.53/1.22	0.52/1.16	0.74/1.77
NSTransformers+Variety loss+GAN	1.21/2.48	0.74/2.11	0.66/1.81	0.51/1.19	0.49/1.12	0.72/1.74
NSTransformers+STAR social interaction module+Variety loss	1.17/2.49	0.68/2.06	0.57/1.56	0.48/1.18	0.44/1.06	0.67/1.67
Social NSTransformers	1.12/2.43	0.65/1.99	0.51/1.48	0.40/1.06	0.38/0.97	0.61/1.59

ers approach. The average ADE/FDE of the base NSTransformers model across datasets is 0.92/1.93. By introducing the STAR social interaction module, the average ADE/FDE improves to 0.80/1.86, which shows the social interaction module's importance in improving the model's performance. With the addition of variety loss, the model performance is further improved, with the average ADE/FDE decreasing to 0.74/1.77. Combining variety loss and GAN, the model performance reaches 0.72/1.74. Ultimately, the NSTransformers model, with the addition of the STAR social interaction module and variety loss, performs the best, with the average ADE/FDE decreasing to 0.67/FDE. FDE is further reduced to 0.67/1.67. Finally, the total Social NSTransformers approach achieves the best performance on all datasets, with an average ADE/FDE of 0.61/1.59. This series of ablation experiments demonstrates the effectiveness of our proposed Social NSTransformers approach in dealing with low-quality pedestrian trajectory prediction and reveals each component's contribution to the overall performance improvement, demonstrating the significant advantages of the model in terms of accuracy and robustness.

E. Self-made animals trajectory prediction experimental results and discussion

Experiments were conducted on a self-made animal dataset. The paper wanted to simulate a strongly stochastic process through the randomness of animal movement, and the final results of the experiments and comparisons are shown in Table VI.

TABLE VI: The experimental results of self-made animals trajectory dataset

Method (ADE/FDE)	Animals dataset
Vanilla LSTM[39]	7.82/15.47
Social LSTM[39]	7.91/15.38
SGAN[40]	7.13/12.15
Sophie[41]	6.52/11.81
GAT[42]	6.33/10.74
Social BiGAT[42]	6.31/10.68
Social STGCNN[43]	6.17/11.36
RSGB[44]	6.74/11.34
RTN[45]	6.29/10.58
STAR[34]	5.89/10.07
E-SR-LSTM[46]	6.05/10.21
TUTR[47]	4.93/9.11
STAGP[48]	6.23/10.97
Social NSTransformers	5.31/9.49

As shown in Table VI, we conducted extensive experiments on the self-made animal trajectory dataset to evaluate the performance of the Social NSTransformers method against other algorithms. The results show that the Social NSTransformers approach significantly outperforms most other comparative

TABLE VII: Rsults of ablation experiments of self-made animals trajectory dataset

Method (ADE/FDE)	Animals dataset
NSTransformers	5.72/9.81
NSTransformers+STAR	5.57/9.74
NSTransformers+Variety loss	5.53/9.68
NSTransformers+Variety loss+GAN	5.49/9.64
NSTransformers+STAR+Variety loss	5.43/9.57
Social NSTransformers	5.31/9.49

algorithms in the animal trajectory prediction task, with an average ADE/FDE of 5.31/9.49 on the animal dataset. This result underscores the effectiveness of Social NSTransformers in dealing with environments with a high degree of dynamism, uncertainty, and sophistication.

As shown in Table VII, we employed ablation experiments in order to dissect the specific contribution of each component of the Social NSTransformers model to its overall performance. The starting point is the basic NSTransformers model, which presents an average ADE/FDE of 5.72/9.81 on this dataset. With the integration of the STAR social interaction module, we observe a significant increase in performance, with the average ADE/FDE decreasing to 5.57/9.74. Further adding the variety of loss components, the model's performance continues to increase, dropping to 5.53/9.68. When the model incorporates variety loss and GAN, its performance is further enhanced, reaching an average ADE/FDE of 5.49/9.64. With the addition of the STAR social interaction module and variety loss, NSTransformers exhibit their best performance, reaching 5.43/9.57. The final version of Social NSTransformers shows its best performance, with a score of 5.43/9.57. The final version of Social NSTransformers performed the best of all configurations tested, achieving an average ADE/FDE of 5.31/9.49. This series of well-designed ablation experiments revealed the critical role of each component in improving trajectory prediction accuracy.

From all the above experimental results, the method proposed in this paper focuses more on trajectory prediction in the low-quality state. It thus performs weaker than other state-of-the-art models in standard pedestrian trajectory prediction tasks.

F. Data visualization results and analysis

In the following analysis of the visualization results, the red line represents the predicted value, the blue line represents the actual value, and the arrows represent the direction of pedestrian travel.

As can be seen from the experimental results in Figure 15, we show a comparison of the performance of four different pedestrian trajectory prediction models in the same scenario. Red lines indicate the prediction results of each model. In contrast, blue lines mark the actual trajectories of the pedestrians, and the arrows indicate the direction in which the pedestrians are traveling. From the figure, it can be observed that Social-LSTM can accurately predict the trajectories of pedestrians in most cases. However, the prediction deviates from the actual trajectories at certain corners.Social-GAN demonstrates good adaptation to changes in pedestrian dynamics, but there is a certain degree of break in the continuity of the trajectories. Social-STGCNN handles the scenarios of walking in a straight line and performs well but needs to improve when predicting complex social interactions. Our method performs superiorly in all test scenarios, mainly when predicting complex pedestrian interactions in crowded scenarios. Despite slight trajectory bias in some extreme cases, overall, our model significantly improves trajectory prediction accuracy, social behavior understanding, and adaptation to future paths. These results indicate that our approach is statistically valid when dealing with the pedestrian trajectory prediction problem and highly reliable in practical applications.

V. Conclusion

In this paper, we proposed a new Social NSTransformers model for low-quality pedestrian trajectory prediction. The model integrates social interaction information between pedestrians in spatial and temporal dimensions to improve the accuracy and robustness of trajectory prediction. We also enhanced the original loss function by combining diversity loss with log-RMSE to ensure the reasonableness and diversity of generated trajectories. Lastly, we introduced GAN to model the randomness in pedestrian trajectories, leading to better diversity and robustness in trajectory prediction. Experimental results show that our proposed model outperforms several state-of-the-art methods regarding prediction accuracy and diversity. Our findings demonstrate the effectiveness of the proposed model for low-quality pedestrian trajectory prediction in complex scenarios.

We aim to advance pedestrian trajectory prediction in future work by focusing on several key areas: We plan to integrate detailed environmental contexts, such as road geometry and weather conditions, to enhance model sensitivity to external factors. Incorporating multi-modal data sources, including video and LiDAR, could significantly enrich model inputs, leading to more accurate predictions. We also aspire to optimize our model for real-time processing applications, which are crucial for autonomous vehicles and intelligent infrastructure, aiming for rapid yet accurate trajectory predictions. Further exploration into human behavior modeling will allow a deeper understanding of pedestrian intentions and social interactions, potentially improving prediction accuracy in complex social scenarios. To broaden its utility and impact, we will investigate the model's applicability across various domains, such as robotics and crowd management. These directions promise to refine our model's performance and extend its relevance to practical applications in real-world scenarios.

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2024.3421175

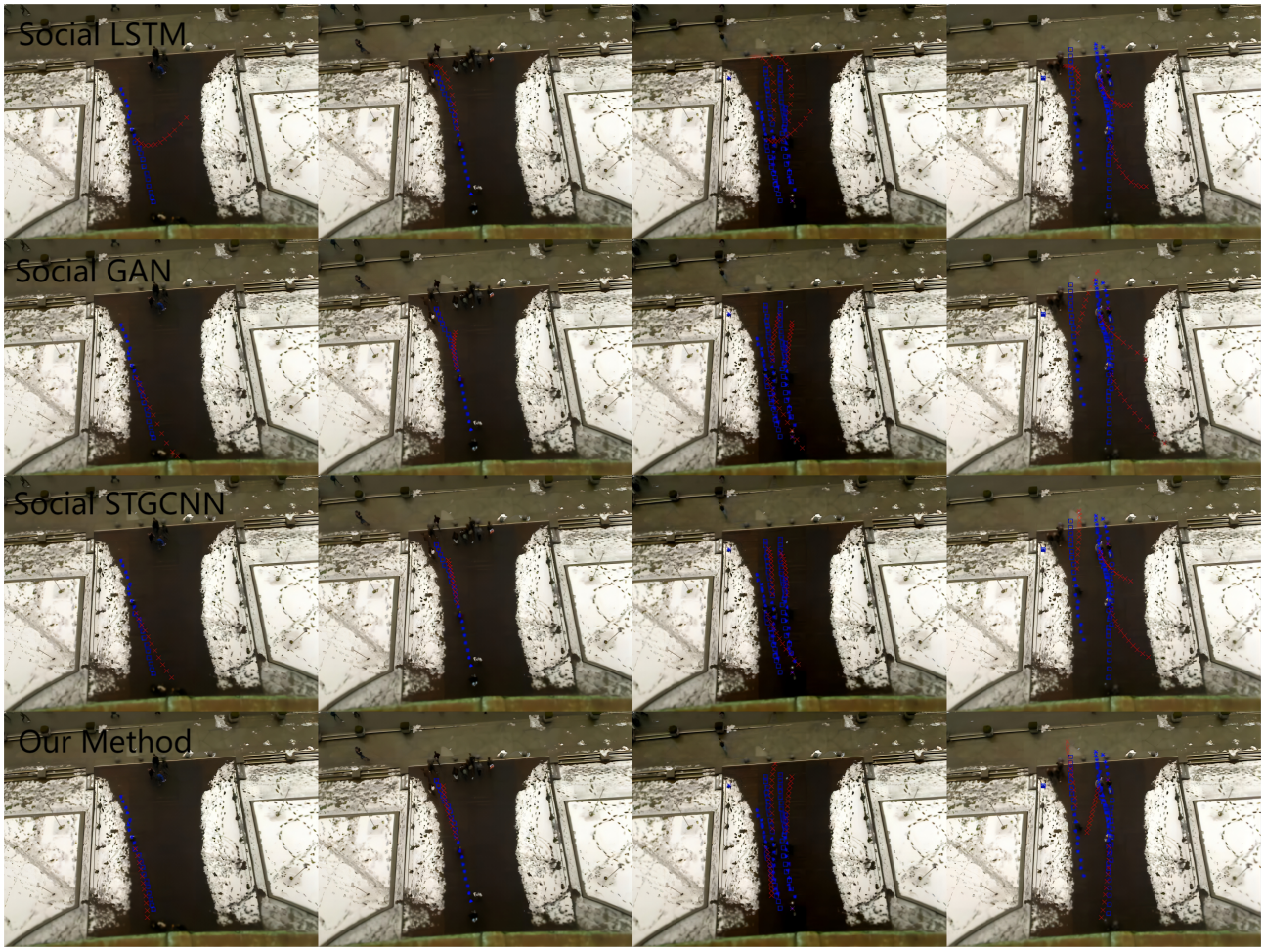

Fig. 15: Visualization of comparative experimental results

REFERENCES

[1] Z. Jiang et al., "Noise Interference Reduction in Vision Module of Intelligent Plant Cultivation Robot Using Better Cycle GAN," IEEE Sensors Journal, vol. 22, no. 11, pp. 11045-11055, 2022

[2] Y. Song, Z. He, H. Qian and X. Du, "Vision Transformers for Single Image Dehazing," *IEEE Transactions on Image Processing*, vol. 32, pp. 1927-1941, 2023

[3] N. Mehta and S. Murala, "Image Super-Resolution With Content-Aware Feature Processing," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 1, pp. 179-191, 2024

[4] T. Barman and B. Deka, "A Deep Learning-based Joint Image Super-resolution and Deblurring Framework," *IEEE Transactions on Artificial Intelligence*, Early Access

[5] H. Wang, L. Peng, Y. Sun, Z. Wan, Y. Wang and Y. Cao, "Brightness Perceiving for Recursive Low-Light Image Enhancement," *IEEE Transactions on Artificial Intelligence*, Early Access

[6] T. Sharma and N. K. Verma, "Adaptive Interval Type-2 Fuzzy Filter: An AI Agent for Handling Uncertainties to Preserve Image Naturalness," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 1, pp. 83-92, 2021

[7] K. Nath, M. K. Bera and S. Jagannathan, "Concurrent Learning-Based Neuroadaptive Robust Tracking Control of Wheeled Mobile Robot: An Event-Triggered Design," *IEEE Transactions on Artificial Intelligence*, vol. 4, no. 6, pp. 1514-1525, 2023

[8] X. He and C. Lv, "Robotic Control in Adversarial and Sparse Reward Environments: A Robust Goal-Conditioned Reinforcement Learning Approach," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 1, pp. 244-253, 2024

[9] X. Li, X. Li, Z. Li, X. Xiong, M. O. Khyam and C. Sun, "Robust Vehicle Detection in High-Resolution Aerial Images With Imbalanced Data," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 3, pp. 238-250, 2021

[10] Zhang Q, and Li C, "Semantic SLAM for mobile robots in dynamic environments based on visual camera sensors," *Measurement Science and Technology*, vol. 34, no. 8, pp. 085202, 2023

[11] R. Korbmacher and A. Tordeux, "Review of Pedestrian Trajectory Prediction Methods: Comparing Deep Learning and Knowledge-Based Approaches," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24126-24144, 2022

[12] M. Golchoubian, M. Ghafurian, K. Dautenhahn and N. L. Azad, "Pedestrian Trajectory Prediction in Pedestrian-Vehicle Mixed Environments: A Systematic Review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 11544-11567, 2023

[13] Y. Huang, J. Du, Z. Yang, Z. Zhou, L. Zhang and H. Chen, "A Survey on Trajectory-Prediction Methods for Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 3, pp. 652-674, 2022

[14] J. Cao et al., "Accelerating Point-Voxel Representation of 3-D Object Detection for Automatic Driving," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 1, pp. 254-266, 2024

[15] Hou J, Yu L, Li C, et al., "Handheld 3D reconstruction based on closed-loop detection and nonlinear optimization," *Measurement Science and Technology*, vol. 32, no. 2, pp. 025401, 2019

[16] S. Mazhar, N. Atif, M. K. Bhuyan and S. R. Ahamed, "Re-

thinking DABNet: Light-weight Network for Real-time Semantic Segmentation of Road Scenes," *IEEE Transactions on Artificial Intelligence*, Early Access

[17] J. Wang, J. Liu, W. Chen, W. Chi and M. Q. . -H. Meng, "Robot Path Planning via Neural-Network-Driven Prediction," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 3, pp. 451-460, 2022

[18] C. Li, L. Yu and S. Fei, "Real-Time 3D Motion Tracking and Reconstruction System Using Camera and IMU Sensors," *IEEE Sensors Journal*, vol. 19, no. 15, pp. 6460-6466, 2019

[19] Z. Jiang et al., "Intelligent Plant Cultivation Robot Based on Key Marker Algorithm Using Visual and Laser Sensors," *IEEE Sensors Journal*, vol. 22, no. 1, pp. 879-889, 2022

[20] C. V., V. S. N. Murthy and S. S. Channappayya, "Siamese Cross-Domain Tracker Design for Seamless Tracking of Targets in RGB and Thermal Videos," *IEEE Transactions on Artificial Intelligence*, vol. 4, no. 1, pp. 161-172, 2023

[21] M. K. Panda, B. N. Subudhi, T. Veerakumar and V. Jakhetiya, "Modified ResNet-152 Network With Hybrid Pyramidal Pooling for Local Change Detection," *IEEE Transactions on Artificial Intelligence*, Early Access

[22] H. Singh, S. Suman, B. N. Subudhi, V. Jakhetiya and A. Ghosh, "Action Recognition in Dark Videos Using Spatio-Temporal Features and Bidirectional Encoder Representations From Transformers," *IEEE Transactions on Artificial Intelligence*, vol. 4, no. 6, pp. 1461-1471, 2023

[23] G. Best and R. Fitch, "Bayesian intention inference for trajectory prediction with an unknown goal destination," *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015, pp. 5817-5823

[24] C. G. Keller and D. M. Gavrila, "Will the Pedestrian Cross? A Study on Pedestrian Path Prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 494-506, 2014

[25] R. Quintero Mínguez, I. Parra Alonso, D. Fernández-Llorca, and M. Á. Sotelo, "Pedestrian Path, Pose, and Intention Prediction Through Gaussian Process Dynamical Models and Pedestrian Activity Recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1803-1814, 2019

[26] E. Rehder and H. Kloeden, "Goal-Directed Pedestrian Prediction," *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, Santiago, Chile, 2015, pp. 139-147

[27] Kim, Sujeong, et al. "Brvo: Predicting pedestrian trajectories using velocity-space reasoning," *The International Journal of Robotics Research*, vol. 34, no. 2, pp. 201–217, 2015.

[28] S. Qiao, D. Shen, X. Wang, N. Han, and W. Zhu, "A Self-Adaptive Parameter Selection Trajectory Prediction Approach via Hidden Markov Models," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 284-296, 2015

[29] W. Chen, F. Zheng, L. Shi, Y. Zhu, H. Sun and N. Zheng, "Multiple Goals Network for Pedestrian Trajectory Prediction in Autonomous Driving," *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, Macau, China, 2022

[30] W. Chen, F. Zheng, L. Shi, Y. Zhu, H. Sun and N. Zheng, "Multiple Goals Network for Pedestrian Trajectory Prediction in Autonomous Driving," *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, Macau, China, 2022, pp. 717-722

[31] C. Yang, H. Pan, W. Sun and H. Gao, "Social Self-Attention Generative Adversarial Networks for Human Trajectory Prediction," *IEEE Transactions on Artificial Intelligence*, Early Access

[32] L. Shi et al., "Representing Multimodal Behaviors With Mean Location for Pedestrian Trajectory Prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 11184-11202, 2023

[33] Liu Y, Wu H, Wang J, et al., "Non-stationary transformers: Exploring the stationarity in time series forecasting," *Advances in Neural Information Processing Systems*, vol. 35, pp. 9881–9893, 2022

[34] Yu C, Ma X, Ren J, et al., "Spatio-temporal graph transformer networks for pedestrian trajectory prediction," *Computer Vision–ECCV 2020: 16th European Conference*, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16

[35] Goodfellow I, Pouget-Abadie J, Mirza M, et al., "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014

[36] Z. Jiang, B. Jin and Y. Song, "A Novel Pet Trajectory Prediction Method for Intelligent Plant Cultivation Robot," *IEEE Sensors Letters*, vol. 7, no. 2, pp. 1-4, 2023

[37] Ge Z, Liu S, Wang F, et al., "Yolox: Exceeding yolo series in 2021". *arXiv preprint arXiv:2107.08430*, 2021

[38] N. Wojke, A. Bewley and D. Paulus, "Simple online and realtime tracking with a deep association metric," *2017 IEEE International Conference on Image Processing (ICIP)*, Beijing, China, 2017, pp. 3645-3649

[39] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei and S. Savarese, "Social LSTM: Human Trajectory Prediction in Crowded Spaces," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 961-971

[40] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese and A. Alahi, "Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 2255-2264

[41] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi and S. Savarese, "SoPhie: An Attentive GAN for Predicting Paths Compliant to Social and Physical Constraints," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019, pp. 1349-1358

[42] V. Kosaraju, R. Martín-Martín, I. Reid, S. H. Rezatofighi, A. Sadeghian and S. Savarese, "Social-BiGAT: multimodal trajectory forecasting using bicycle-GAN and graph attention networks", *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA: Curran Associates Inc., 2019, pp. 137–146.

[43] A. Mohamed, K. Qian, M. Elhoseiny and C. Claudel, "Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 14412-14420

[44] J. Sun, Q. Jiang and C. Lu, "Recursive Social Behavior Graph for Trajectory Prediction," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 657-666

[45] H. Sun, Z. Zhao, Z. Yin and Z. He, "Reciprocal Twin Networks for Pedestrian Motion Learning and Future Path Prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1483-1497, 2022

[46] P. Zhang, J. Xue, P. Zhang, N. Zheng and W. Ouyang, "Social-Aware Pedestrian Trajectory Prediction via States Refinement LSTM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2742-2759, 2022

[47] L. Shi, L. Wang, S. Zhou and G. Hua, "Trajectory Unified Transformer for Pedestrian Trajectory Prediction," *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, Paris, France, 2023, pp. 9641-9650

[48] Z. Liu, L. He, L. Yuan, K. Lv, R. Zhong and Y. Chen, "STAGP: Spatio-Temporal Adaptive Graph Pooling Network for Pedestrian Trajectory Prediction," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2001-2007, 2024

[49] Bo Jin, Nuno Gonçalves, Leandro Cruz, Iurii Medvedev, Yuanyu Yu and Jiujiang Wang, "Simulated multimodal deep facial diagnosis," *Expert Systems with Applications*, pp. 123881, 2024

[50] J. Gao, H. Wang and H. Shen, "Task Failure Prediction in Cloud Data Centers Using Deep Learning," *IEEE Transactions*

on Services Computing, vol. 15, no. 3, pp. 1411-1422, 2022

[51] B. Jin, L. Cruz and N. Gonçalves, "Deep Facial Diagnosis: Deep Transfer Learning From Face Recognition to Facial Diagnosis," *IEEE Access*, vol. 8, pp. 123649-123661, 2020

[52] Zhao M, Jha A, Liu Q, et al., "Faster Mean-shift: GPU-accelerated clustering for cosine embedding-based cell segmentation and tracking," *Medical Image Analysis*, vol. 71, pp. 102048, 2021

Zihan Jiang (Student member, IEEE) received the B.Eng. degree in Northeast Forestry University, Harbin, China, in 2022. He is currently pursuing the M.Sc. degree at the University of Liverpool, UK. His research interests include trajectory prediction and computer vision.

Yiqun Ma received the B.Eng. degree in computer science and technology from Nanjing University of Information Science and Technology in 2021. She is currently pursuing the M.Sc. degree at the University of Liverpool, UK. Her research interests include computer vision and semi-supervised learning.

Bingyu Shi (Student member, IEEE) received the B.Eng degree from Northeast Forestry University in Harbin, China, in June 2022, and is currently continuing to pursue a doctor of philosophy in engineering at Northeast Forestry University. Her research interests include remote sensing data understanding and wildfire prediction.

Xin Lu is currently pursuing the M.S. degree with the School of Advanced, Xi'an Jiaotong-Liverpool university, SuZhou, China. Her current research interests include algorithms and system for artificial intelligence

Jian Xing (Member, IEEE) was born in 1979. He received the Ph.D. degree in instrument science and technology from the Harbin Institute of Technology, Harbin, China, in November 2011. He is currently a Professor and the Dean of Electronic Information Engineering with Northeast Forestry University, Harbin. Several articles were published in Optics express and Optics letter. The main research directions are applying optics and radiation thermometry technology.

Nuno Gonçalves (Member, IEEE) received the Ph.D. degree in Computer Vision from the University of Coimbra, Portugal, in 2008. Since 2008, he has been a Tenured Assistant Professor with the Department of Electrical and Computers Engineering, Faculty of Sciences and Technology, University of Coimbra. He is currently a Senior Researcher with the Institute of Systems and Robotics, University of Coimbra, where he researches since 2000. He has been coordinating several projects centered on the technology transfer to the industry. In 2018, he joined the Portuguese Mint and Official Printing Office as an Innovation Manager.

His main Research topics and projects coordination areas include several lines, such as biometrics, facial recognition, morphing attack detection, presentation attack detection, graphical security, security coding, steganography, and robotics. He has been working in the design and introduction of new products as result of the innovation projects. He is the author of several papers and communications in high-impact journals and international conferences and six patents, pending and already Granted by EUIPO and USPTO. His scientific career has been mainly developed in the fields of computer vision, visual information security, biometrics, computer graphics, autonomous driving and robotics.

Bo Jin (Member, IEEE) was born in Mainland China. He received both his B.Sc. and M.Sc. degrees from the Department of Electrical and Computer Engineering, University of Macau, Macau SAR, China. He earned his Ph.D. degree from the Department of Electrical and Computer Engineering, University of Coimbra - Alta and Sofia, Coimbra, Portugal. Over the years, Bo Jin conducted his doctoral research and was conducting his postdoctoral research with the Visual Information Security Team at the Institute of Systems and Robotics, University of Coimbra, Portugal. To date, he serves as a Postdoctoral Fellow.

He published the research results related to Deep Facial Diagnosis, which was awarded a national invention patent by the People's Republic of China (PRC). He has a broad spectrum of research interests, with a particular focus on computers, genetics, and robotics.