IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

IEEE VEHICULAR TECHNOLOGY SOCIETY SECTION

Received 20 September 2023, accepted 26 October 2023, date of publication 6 November 2023,
date of current version 29 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3330376

RESEARCH ARTICLE

Pedestrian Trajectory Prediction Based on SOPD-GAN Used for the Trajectory Planning and Motion Control of Mobile Robot

HAO LI, (Student Member, IEEE), DONG-HAI QIAN, GUANG-YIN LIU, ZE CUI, AND JING-TAO LEI

School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China

Corresponding author: Dong-Hai Qian (dhqian@shu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 51775323.

ABSTRACT Recent developments in the field of robotics have led to discussions surrounding the human-robot coexistence environments including homes and modern factories. Focusing on the application of mobile robots, the focus of this research is to enhance their performance in dynamic scenarios. To effectively plan the robot's path to avoid pedestrians, a machine learning algorithm is employed to predict the future trajectory of pedestrians, thus improving the accuracy of forecasting their multi-modal motion. The existing prediction methods primarily rely on pedestrian history and current movement attributes to predict future movement, they often overlook the impact of static obstacles on pedestrian movement decision. Therefore, in this study, a static obstacles probability description generative adversarial network (SOPD-GAN) is proposed. The static obstacles probability description (SOPD) represents the future movement space of pedestrians and assesses the likelihood of entry. Additionally, we incorporate pedestrian historical trajectory information using LSTM, and combine it with SOPD to form the generator model. The training of this model is carried out using a generative adversarial network (GAN), which is referred to as SOPD-GAN. In addition, we also introduce an improved dynamic window approach (IDWA) for robot path planning in dynamic scenarios based on pedestrian trajectory prediction. In order to validate the efficacy of our approach, we conduct experiments in real scenarios and compare the model with existing baselines. The results show that this method can construct a suitable prediction model with high accuracy. Specifically, our method achieved an accuracy of 0.0881 and 0.0691 in FDE and AEE of predicting pedestrian trajectory, surpassing the baseline method by 20% and 14%.

INDEX TERMS Human–robot coexistence environments, mobile robot, pedestrian trajectory prediction, LSTM, GAN, Von-Mises-distributed.

I. INTRODUCTION

In recent years, the development of mobile robots has sparked discussions regarding their applications in environments where humans and machines coexist, such as homes, offices, exhibitions, and modern factories. In these scenarios, mobile robots can assist humans in performing various tasks, including carrying and inspection. For instance, in a factory setting, a handling robot may need to transport goods from one designated location to another. To achieve this, the robot must autonomously plan its path based on environmental information. Typically, the robot distinguishes between obstacles and open areas by utilizing a static map. However, during actual operation, the robot will inevitably encounter temporary dynamic obstacles that are not accounted for in the map, such as pedestrians or other working robots. Therefore, planning and controlling the trajectory of the robot in a potentially dynamic environment is a challenging and practical task.

To address this problem, it is crucial to implement recognition and tracking of moving objects, as well as future trajectory prediction, and apply these capabilities to the

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Quan.

motion control of mobile robots. The most commonly used sensors for achieving recognition and tracking of moving objects are lidar, vision sensors, and depth cameras. These sensors provide the robot with external information. Several methods, such as those utilizing point clouds, RGB images, and depth images, have been proven effective in this regard [1], [2], [3].

Accurate prediction of pedestrians' trajectories can significantly enhance the robot's navigation performance. Trajectory refers to the positional change of an object over time. This change in position and orientation within a given time step can be interpreted as velocity information [4]. The trajectory prediction problem entails estimating the future movements of a moving object based on historical information [5] obtained from its behavior within a particular scene.

Predicting trajectories in scenes involving multiple dynamic objects is a challenging task that requires considering the spatio-temporal interactions between pedestrians and the scene, the human analysis and decision-making processes involved in the environment [6], as well as the stochastic nature of human walking patterns. While walking, pedestrians continuously adjust their trajectories based on the motion states of other dynamic objects and the positions of static obstacles [7] in order to avoid collisions. Additionally, the environment surrounding pedestrians and their spatial relationships with other objects change over time, necessitating constant adjustments to their own trajectories. Hence, there exists a natural interplay between pedestrian trajectory prediction and the spatio-temporal environmental information [8].

Currently, several methods have been proposed for trajectory prediction, which can be mainly categorized into two directions.

1) The first approach based on knowledge-based (KB), or called physics-based [9], reasoning-based [10] or traditional approach [11].

2) The second direction involves the deep learning (DL) method, which is data-driven. DL refers to the training of neural network with more than two hidden layers that derive supervised learning from data.

The KB approach involves building a motion model based on intuitive characteristics of pedestrians, which is then used to predict future trajectories using various physical models. Classical physical models such as the uniform acceleration model [12] and uniform velocity model [13] are utilized to describe the motion state of pedestrians. Trajectory prediction has also seen the application of filter-based methods like the extended Kalman filter (EKF) [14]. However, due to the highly random nature of pedestrian motion affected by multiple variables, traditional KB approaches face challenges in handling motion mutability.

Recently, researchers have utilized DL methods, particularly long short-term memory networks (LSTM), convolutional neural networks (CNN), and generative adversarial networks (GAN), to address pedestrian trajectory prediction problems. The researchers have transformed time-varying trajectory prediction into a task of generating time series data [15]. Due to the proven performance of recurrent neural networks (RNN) for this problem [16], [17], the approach of pedestrian trajectory prediction based on modern RNN networks has garnered significant attention. Unlike KB methods, DL methods do not require interpretable parameters and physical modeling. However, they do require extensive data and flexible algorithms.

LSTM is a widely used deep learning method in trajectory prediction and has demonstrated effectiveness. Alahi et al. introduced a Social-LSTM model [15], which combines LSTM with a Social pooling layer. It innovatively incorporates pedestrian interaction information into trajectory prediction.

Social-LSTM assumes that pedestrian trajectories follow a bivariate Gaussian distribution, and we adopt this assumption in our model as well. Taking advantage of GAN's significant advancements and success in other domains, Gupta et al. [18] proposed Social-GAN. GAN is integrated into pedestrian trajectory prediction tasks, combining time series prediction with a generative adversarial architecture. This architecture enhances the resilience and accuracy of network models in complex scenarios by utilizing trajectory predictions generated through iterative adversarial training.

The previously mentioned methods take into account the interaction factors generated during the movement of multiple dynamic targets and incorporate the historical track information of pedestrians. However, they neglect the influence caused by the static environment [19], especially the complex interaction between pedestrians and their surroundings. These effects alter the decision-making process of pedestrian movement, which undoubtedly has a significant impact on trajectory prediction [20]. Therefore, these models have certain limitations in the task of trajectory prediction in occupied areas.

Autonomous obstacle avoidance serves as the foundation for intelligent operation of mobile robots. In the conventional obstacle avoidance algorithm, the artificial potential field (APF) method depicts the robot as a particle [21] in space, influenced collaboratively by the generated repulsive force field and the gravitational field. The force field arises from the obstacle, target point, or target trajectory. By utilizing APF, the robot can achieve optimal direction and speed control. The dynamic window approach (DWA) restricts the robot's velocity samples [22] within specific bounds, employing kinematic constraints such as velocity and acceleration. This approach determines the current optimal motion state and future trajectory based on preset evaluation criteria, including variables such as movement speed and deviation of the robot's direction from the target point.

All of these methods fall within the realm of traditional reactive trajectory planning, which treats obstacles as instantaneous static entities. However, in an environment of human-machine coexistence, these methods overlook the distinctive

aspects of human presence and motion. The robot ought to devise its future trajectory based on the motion characteristics [23] of the dynamic objects, particularly as the scene continues to evolve dynamically.

Aiming at enhancing the capability of robots to evade obstacles in dynamic environments, this paper proposes a method for dynamic obstacle avoidance in mobile robots by leveraging pedestrian trajectory prediction. Initially, a LiDAR-based approach for pedestrian recognition and tracking is introduced, which utilizes point cloud data exclusively to recognize and extract dynamic objects. From the current point cloud information and the mapping of static environment point clouds, the recognition function module combines the clustering growth algorithm. Such an approach enables recognition and extraction of dynamic discrete point cloud clusters, and facilitates tracking of dynamic point clouds.

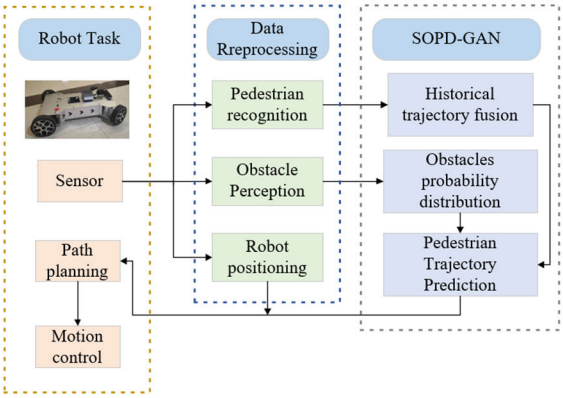


FIGURE 1. Workflow diagram of the motion robot framework.

Secondly, this paper presents a novel deep learning-based model for predicting pedestrian trajectories. Our prediction model combines and models pedestrian movement trends, movement speed, and global environmental information in a time series. The proposed network model incorporates LSTM units and GAN architecture, with the addition of a static obstacle probability distribution module (SOPD). This module analyzes how static obstacles impact the future motion intentions and trajectories of pedestrians. Pedestrian prediction is employed to estimate the probability distribution. Specifically, the future state X_t of the pedestrian $p(X_t|X_t^{obs}, \Theta_t)$ is based on the observation of the pedestrian's historical track X^{obs} and the distribution of the global environment Θ^{obs}. Considering the influence of static obstacles on pedestrian movement in space, this information is introduced as an additional variable to enhance the prediction. The distribution of pedestrian entry probabilities is estimated dynamically.

Thirdly, building on pedestrian trajectory prediction, we present a novel approach called the Improved Dynamic Window Approach (IDWA). Our approach integrates information about robot and obstacle motion to enhance the local path planning performance of robots operating in dynamic environments. Considering the temporal characteristics of pedestrian position and movement, our method incorporates future trajectory avoidance for dynamic objects, thereby overcoming the limitations of traditional path planning methods.

The future trajectory of the mobile robot is calculated based on the velocity sample space of the robot, considering kinematic constraints. This calculation is done for different velocity samples. In order to achieve collision-free trajectory planning in a dynamic environment, the trajectory of the dynamic obstacle is compared to the future trajectory of the robot under time-varying conditions. The purpose of this comparison is to determine the minimum distance between the two trajectories.

Finally, the Lyapunov direct method is employed to achieve tracking control of the robot's reference trajectory.

The main contributions of this paper are as follows:

1) We propose a dynamic object recognition and detection method that utilizes Lidar data. This method allows for real-time detection of dynamic objects using point cloud data and enables long-term tracking of pedestrian motion.

2) We introduce a novel pedestrian trajectory prediction model that combines LSTM with static obstacle spatial probability correction. By considering pedestrian movement attributes and environmental constraints and leveraging GAN architecture, this method significantly improves the accuracy of predicting pedestrian future trajectories in complex scenes.

3) To address human-machine coexistence environments, we propose IDWA, which enables intelligent motion control of robots in dynamic settings. Through experimental validation, the algorithm demonstrates excellent dynamic obstacle avoidance capabilities and generates planned paths that align with human environmental motion rules. The experiments confirm the effectiveness of our method in enhancing motion performance of robots working in human-machine coexistence scenarios.

II. BACKGROUND KNOWLEDGE

In this section, the standard LSTM cell and GAN architecture used in our model are described.

A. LSTM

LSTM is a variant of RNN which can learn from sequential data. LSTM can capture the interrelationship between long-term and short-term information within a sequence. In predicting pedestrian trajectories, the future movements of pedestrians may depend on their long-term and short-term historical trajectories.

In this paper, LSTM is utilized as a component of the future trajectory prediction model. The following formula presents the fundamental update formula for LSTM.

$$i_t = \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i\right) \quad (1)$$

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \quad (2)$$

$$\hat{C}_t = \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right) \quad (3)$$

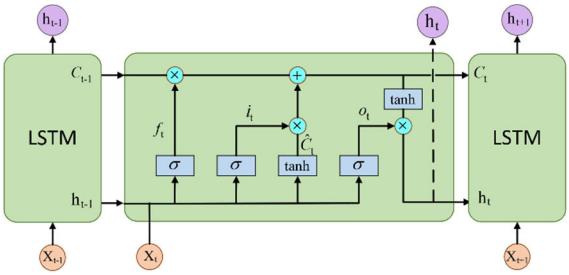


FIGURE 2. Structure of LSTM cell.

$$C_t = f_t \odot C_{t-1} + i_t \odot \hat{C}_t \tag{4}$$

$$o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o\right) \tag{5}$$

$$h_t = o_t \odot \tanh\left(C_t\right) \tag{6}$$

where i_t, f_t, o_t are represented the input door, the forgetting door, and the output door respectively. They decide what information needs to be stored, what needs to be forgotten, and what needs to be output as the state of the current moment. C_t is a long-term memory cell that is used and updated at each time step, and it can use information from earlier time steps to calculate the current moment. x_t, h_t represents the feature vector of the input value of the current moment and the hidden layer state vector of the output, and as the output of the current moment, h_t will be transmitted as short-term memory information to the neural network operation of the next moment. σ denotes a Sigmoid activation function that maps input values to [0, 1], controls the size of the data stream. tanh represents a hyperbolic tangent function. The update of long-term memory cells C_t relies on the input gate i_t and the forgetting gate f_t to control what historical information needs to be abandoned from the state of the previous moment C_{t-1}, so as to avoid noise information affecting the prediction. x_t will be incorporated into the memory cell C_t by i_t, so that the current input is saved to achieve the update of long-term memory.

Based on the three network structure gates, LSTM can analyze time series tasks, apply and generate future predicted changes. In this study, LSTM is used as the basic unit to build a dynamic object motion prediction model.

B. GAN

Trajectory prediction of dynamic objects is a challenge as it involves generating multi-modal trajectories. Traditional methods struggle to address this problem due to the high level of agent randomness [24]. GANs are extensively utilized in various man-machine fields, including image recognition, multi-language translation, and data enhancement, due to their capability of generating multi-modal samples. They have proven to be pioneering in this domain.

The structure of a GAN is composed of generator and discriminator. The discriminator's objective is to differentiate the generator's output and compel it towards generating more realistic trajectories. This process counteracts the training model and enhances the network's performance in complex problem domains.

Social-GAN incorporates the GAN model in the domain of trajectory generation to address the challenge of representing multi-modal trajectories. it utilizes the global pooling method to differentiate the impact of other pedestrians in the scene. However, it encounters challenges in accurately distinguishing the influence of pedestrians with varying speeds and positions on the target. STI-GAN [20] combines the GAN network with the graph attention network (GAT) to discern the impact of pedestrians [25] in different states on the target by utilizing the attention mechanism's influence. This enhancement enables the network model to dynamically consider the potential interference from surrounding pedestrians.

To address the agent prediction problem, our research presents a pedestrian motion estimation method that incorporates the sum of pedestrian dynamics and environmental constraints. This estimation method is then integrated into GAN to model the intricate behavior of targets at the spatio-temporal level, thereby enhancing the trajectory prediction performance.

III. DYNAMIC OBJECT RECOGNITION BASED ON POINT CLOUD

Lidar-derived sensor data is frequently utilized in the context of robots' environmental perception tasks. Due to its exceptional measurement accuracy and robustness [26], Lidar is extensively employed across diverse environments, including challenging lighting conditions. By capturing point cloud information of the environment, Lidar can determine the size and position of the target objects accurately. On the other hand, vision sensor-based approaches can be trained using specific object data for object recognition and tracking [3]. However, point clouds primarily contain spatial information instead of social properties of objects.

This paper addresses the challenge of robot motion in dynamic environments, with a particular focus on the kinematic properties of dynamic obstacles [27]. Therefore, Lidar has become our preferred choice due to its ability to capture dynamic obstacle motion effectively.

In this section, we propose a dynamic object recognition method that utilizes static point cloud matching. This method can effectively identify dynamic objects within the robot's field of vision and is applicable to a variety of environments. It provides the foundational information for subsequent mobile robots to avoid dynamic obstacles and navigate accordingly.

A. POINT CLOUD MATCHING

In this study, we enhance the NDT algorithm [28], [29], a renowned method for point cloud matching, to achieve dynamic point cloud extraction.

First, the space is rasterized and the point cloud information is used to initialize the grids. According to the given size, the space is divided into several grids of the same size, and the points in the source point cloud P are projected into the

corresponding grids according to the coordinates. If there are at least 5 points in the grid, the mean q and covariance matrix $\sum$ of the points contained in it are calculated:

$$q = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{7}$$

$$\sum = \frac{1}{n}\sum_{i=1}^{n} (X_i - q)(X_i - q)^T \tag{8}$$

where $X_i = [x_i, y_i]$ is the point cloud coordinates contained in the grid, n representing the number of point clouds contained.

Next, the grids are initialized based on the calculated q and $\sum$. By constructing the probability density function, the discrete point cloud can be represented continuously by piecewise. The probability density of each grid can be expressed as:

$$p(X) \propto \exp\left(-\frac{(X-q)^T \sum^{-1} (X-q)}{2}\right) \tag{9}$$

The above steps are repeated for all the grids and finally complete the initialization of the spatial grid. The target point cloud Q is matched into the grids and calculate the score of each point cloud $p(X_i)$ and the total score $S(p)$:

$$p(X_i) = \exp\left(-\frac{(X_i-q)^T \sum^{-1} (X_i-q)}{2}\right) \tag{10}$$

$$S(p) = \sum_{i=1}^{n} p(X_i) \tag{11}$$

Among them, $S(p)$ is the objective function to be optimized, and our goal is to find the optimal target transformation matrix T.

$$T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \tag{12}$$

where R represents rotation matrix and t represents translation matrix. Based on the Gauss-Newton method, the total score of the Q in the grid is optimized under the T, which means that the two frames of point clouds get the maximum degree of coincidence matching. That is, the point cloud matching task is completed.

B. DYNAMIC POINT EXTRACTION

In dynamic environments, dynamic objects often exist in the free area of the scene, as described in our previous work [30]. From the perspective of point cloud data analysis, the following two assumptions can be reasonably made: 1) Dynamic point clouds mainly exist in blank grids that are not initialized by source point clouds; 2) Point clouds have a more concentrated spatial distribution which are originating from the same object.

Based on hypothesis 1, the matched target point cloud $\tilde{Q}$ are put into the space, and the point clouds mapped to the blank grid are statistically to the point cloud D which is regarded as potential dynamic point cloud convergence.

$$\tilde{Q} = Q \odot T \tag{13}$$

C. DYNAMIC OBJECT RECOGNITION

In dynamic scenarios, the simultaneous existence of multiple dynamic targets is common. In robot motion, it is crucial to perceive the entire target and distinguish it from the previously acquired dynamic point cluster. Hypothesis 2 serves as the foundation for adopting the region growth method to extract dynamic objects from the point cloud data.

Dynamic point cloud clusters are extracted using the Euclidean clustering algorithm [31]. The fundamental concept of the Euclidean clustering algorithm is to group points within a specified Euclidean distance into clusters. As different dynamic objects have distinct spatial distances, the dynamic point cloud clustering method based on Euclidean distance achieves effective extraction of objects in the scene.

Finally, the collection of point cloud clusters $C = [C^1, C^2, \ldots, C^n]$ can be get, where C^i represents the point cloud information of a dynamic target. The center of mass of C^i can be seen the location information of the current pedestrian.

$$X_t^i = \frac{1}{m}\sum_{j=1}^{m} C_j^i \tag{14}$$

where X_t^i represents the position information of the first dynamic object in the t time step, m represents the number of point clouds in C_j^i and C_j^i represents the j point cloud coordinate information in the i dynamic point cloud cluster.

Through the above method, the location information extraction of dynamic object recognition can be realized based on point cloud, obtain the historical trajectory of dynamic object $X^i = [X_1^i, X_2^i, \ldots, X_t^i]$, realize the target recognition based on Lidar, and provide data support for the subsequent dynamic obstacle prediction and robot control.

IV. PEDESTRIAN TRAJECTORY PREDICTION MODEL

In this section, our aim is to achieve the prediction of dynamic objects by robots in a dynamic environment. The pedestrian is the primary object in the robot's visual field, and accurately predicting their motion trajectory can greatly enhance the robot's performance. Pedestrian behavior and movement are commonly influenced by specific goals, such as the presence of other pedestrians and obstacles that hinder their movement. When planning our path, we take into account the target location and environmental information. Prior research has neglected the significance of the pedestrian environment, particularly static obstacles, which play a crucial role in pedestrians' decision-making process [15], [18], [25], [32]. In such scenarios, multiple alternative paths are often present, and given the varying motion patterns of pedestrians, a predictive model capable of capturing diverse influencing variables becomes necessary.

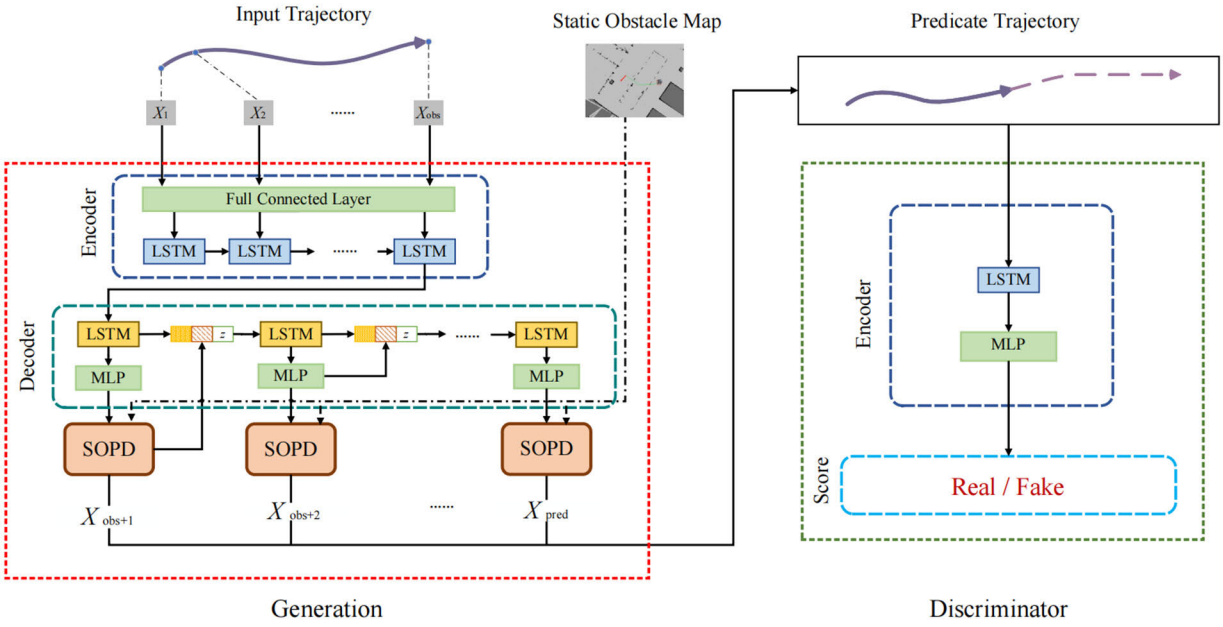


FIGURE 3. We propose a multimodal network architecture that incorporates static obstacle constraints. The network's output is generated using a GAN model and includes four key components: an encoder, a decoder, a SOPD module, and a discriminator.

In this paper, we propose a novel prediction model called SOPD-GAN (Static Obstacles Probability Distribution Generation Adversarial Network). This model comprehensively considers the history of pedestrian trajectories and interacts with its surroundings to accurately predict pedestrian paths. This section introduces the model architecture and provides a detailed explanation of the static obstacles. The pedestrian movement is described based on probability distribution, and a motion estimation correction module is developed accordingly. The model predicts pedestrian trajectories based on historical data and incorporates obstacle information as a potential variable to enhance prediction performance.

A. PROBLEM DEFINITION

Our goal is to forecast and anticipate the future movement of the agent in the scene by analyzing the historical trajectory of the object and the relevant scene information. The position of the pedestrian can be determined using an accurate target detection algorithm. As input for the model, we can predict the future movement trajectory of the pedestrian.

Based on the dynamic object extraction framework proposed, pedestrian location detection is implemented based on point cloud as the data preprocessing part of the proposed model. The input of model is assumed that the pedestrian's history trajectory $\mathbf{X} = X_1, X_2, \cdots, X_n$, and the future predicted trajectory is the output which defined as $\hat{\mathbf{Y}} = \hat{Y}_1, \hat{Y}_2, \cdots, \hat{Y}_n$. Where the historical trajectory of an object is defined as $X_i = \{(x_i^t, y_i^t) | t = 1, \cdots, T_{obs}\}$, and the environment in which the object resides is defined as Θ_t. In addition, the predicted trajectory is defined as $\hat{Y}_i = \{(\hat{x}_i^t, \hat{y}_i^t) | t = T_{obs+1}, \cdots, T_{pred}\}$, and the real future trajectory is defined as $Y_i = \{(x_i^t, y_i^t) | t = T_{obs+1}, \cdots, T_{pred}\}$.

B. OVERALL MODEL

In the scene, pedestrians determine their navigation decisions based on the constraints of fixed obstacles and their own objectives. While some literature has examined the influence of static obstacles on pedestrian paths, these studies primarily focus on local interactions and overlook the impact of the overall scene situation. Our proposed model considers the interaction between pedestrians and the environment, incorporating multiple variables.

To accomplish this, we introduce the SOPD-GAN model, which is comprised of four key modules: the feature encoder module, decoder module, static obstacle probability distribution module (SOPD), and discriminator module. The network structure is depicted in Fig. 3.

The feature encoder module employs a Fully Connected Layer (FC) to extract high-dimensional features from historical track information. These features are then encoded using LSTM to capture the historical information. Subsequently, the encoder processes the motion features within the pedestrian's historical track information, culminating in the generation of feature encoding, assuming pedestrian free movement.

In the decoder module, the input consists of integrated random noise, feature encoding, and SOPD estimation from the previous time. Leveraging the diverse feature set, the generator aims to generate a plausible distribution of pedestrian motion as a predictive output, while simultaneously attempting to deceive the discriminator.

In SOPD, we conduct joint probabilistic modeling to estimate pedestrian movement information and environment information. We model the movement direction and displacement increment of pedestrians as independent variables,

and estimate the probability of the spatial impact on the static environment. By solving the joint probability optimally, we can obtain an estimation of pedestrian trajectory under pedestrian constraints.

Subsequently, the generated trajectories are evaluated using a discriminator to determine their probability of being true or false trajectories. The main purpose of the discriminator is to serve as an auxiliary module for the generator. Through adversarial training, the discriminator compels the generator to produce more realistic samples, thereby enhancing the prediction performance of the generator model. When the discriminator cannot clearly distinguish the authenticity of the trajectory, it signifies that the output is reasonable.

C. FEATURE ENCODER MODULE

The feature encoder module mainly extracts the feature representation of the observed pedestrian trajectory based on the LSTM structure. Firstly, the relative displacement of pedestrians in each time step is embed into a high-dimensional eigenvector e_i^t of fixed length though the FC;

$$e_i^t = \phi(x_i^t, y_i^t; W_{ee}) \tag{15}$$

where $\phi(\cdot)$ represents a RuLU nonlinear activation function, W_{ee} represents the weight of the layer network. Then, LSTM is used to capture the time dependence between pedestrian motion states. e_i^t is used the input to the encoder LSTM cell at time t for pedestrian i.

$$h_{ei}^t = LSTM^{\text{en}}(h_{ei}^{t-1}, e_i^t; W_{encoder}) \tag{16}$$

where $LSTM^{\text{en}}(\cdot)$ represents the encoder LSTM, h_{ei}^{t-1} represents the output state of the LSTM at the last time step and $W_{encoder}$ represents the weight of the cell.

It should be noted that when there are multiple agents in the scene, each object is assigned a separate encoder module, and the weight of module is shared between each module.

D. DECODER MODULE

The motion status of the object is analyzed through the encoder module and historical motion information of pedestrians is stored. In SCP-GAN, the encoder and decoder are symmetrical structures, and the decoder is also composed of a single-layer LSTM cell and an MLP.

$$h_{di}^t = LSTM^{\text{de}}(h_{di}^{t-1}, \rho_i^{t-1}; W_{decoder}) \tag{17}$$

$$\bar{Y}_i^t = \phi(h_{di}^t; W_{ed}) \tag{18}$$

where $LSTM^{\text{de}}(\cdot)$ represents the LSTM cell in decoder module, h_{di}^t represents the hidden state of the layer, ρ_i^{t-1} Indicates the status of the SOPD module which will be introduced in next chapter, $\varphi(\cdot)$ Indicates the MLP, and The weight of them is represented by $W_{decoder}$ and W_{ed}.

The decoder ultimately outputs a set of bivariate GMM predictions $\bar{Y}_i^t$ that treat the incremental movement of pedestrians in the direction as an independent variable.

$$\bar{Y}_i^t = (\Delta\bar{x}_t, \sigma_{xt}^2, \Delta\bar{y}_t, \sigma_{yt}^2) \tag{19}$$

The network's embedding layer and LSTM structure aim to capture valuable information regarding the historical trajectories and complex motion patterns of pedestrians. These layers analyze and predict pedestrians' historical motion information to generate an initial estimation of motion trends.

E. STATIC CONTEXT PROBABILITY DISTRIBUTION

In this section, we will introduce the approach of estimating pedestrian motion attributes and obstacle information as probability distributions. Specifically, what the future state of pedestrians $p(X_t|X_t^{obs}, \Theta_t)$ is concerned based on the distribution of historical observation trajectory X^t and context Θ_t.

The influence of static environment on pedestrian movement at time step T are introduced as a potential variable to improve prediction, where the spatial influence of pedestrians is estimated in real-time. Due to the interference of obstacles, the direction of pedestrian movement will also change accordingly. Therefore, in this module, the directional information is also considered, so the state of pedestrians can be represented as:

$$X_t = (x_t, y_t, \psi_t) \tag{20}$$

As introduced above, we are interested in the state transition of pedestrians from previous time step t-1 to time step t, which can be represented as:

$$X_t = X_{t-1} + \Delta X_t \tag{21}$$

where ΔX_t represents the increment of pedestrian position from time step $t-1$ to t:

$$\Delta X_t = (\Delta x_t, \Delta y_t, \Delta\psi_t) \tag{22}$$

1) PROBABILITY ESTIMATION OF MOTION STATE

In this work, a trajectory prediction neural network model is built based on LSTM to predict the incremental movement of pedestrians without constraints and additional constraints when they only care about historical trajectory information. The increment of x, y are modeled as independent variables. They can be got according to formulas (18) - (19).

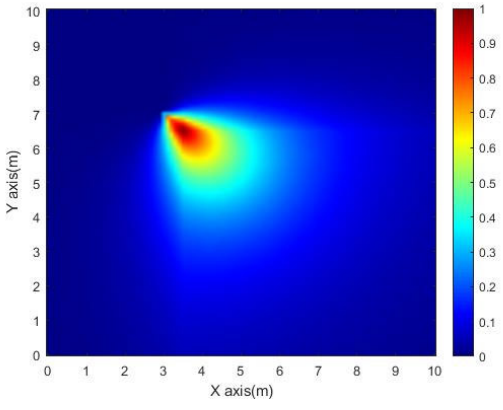


FIGURE 4. Motion probability estimation, the pedestrian moves from upper left corner to lower right corner.

Given the directional inertia of pedestrian motion, the information regarding their direction ψ is incorporated as an independent variable in the prediction model. Previous studies [33] have successfully employed the Von-Mises distribution approach to describing directional distribution. Hence, we model the pedestrian motion direction as a Von-Mises-distributed variable with specific mean and concentration parameters κ_ψ.

Therefore, the incremental estimate of pedestrian movement at time step t can be described as:

$$p\left(\Delta x_t, \Delta y_t, \Delta\psi_t\right) \propto \exp\left(-\frac{(\Delta x_t - \Delta\bar{x}_t)^2}{\sigma_{xt}^2}\right) \cdot \exp\left(-\frac{(\Delta y_t - \Delta\bar{y}_t)^2}{\sigma_{yt}^2}\right) \cdot \exp\left(\Delta\kappa_\psi \cos(\psi_t - \Delta\bar{\psi}_t)\right) \tag{23}$$

$$\Delta\bar{\psi}_t = arctan(\frac{\Delta\bar{y}_t}{\Delta\bar{x}_t}) \tag{24}$$

According to formula (23), an estimate of the incremental distribution $p\left(X_{t-1}^{obs}\right) = p\left(\Delta x_t, \Delta y_t, \Delta\psi_t\right)$ can be obtained, and the future pose estimation of pedestrians can be described as follows:

$$p(X_t|X_t^{obs}) \propto p\left(X_t^{obs}\right) \cdot p(X_{t-1}|X_{t-1}^{obs}) \tag{25}$$

2) PROBABILITY ESTIMATION OF STATIC CONTEXT

As previously stated, pedestrian movement behavior varies across different environments. An illustrative instance of this is when a pedestrian adjusts their direction to avoid obstacles. To account for this, a location prior is introduced within a particular environment, which signifies the likelihood of a pedestrian entering a specific area of space. Within this module, the location prior is represented as follows:

$$p(X_t|X_t^{obs}, \Theta_t) = p(X_t|\Theta_t) \cdot p(X_t|X_t^{obs}) \tag{26}$$

Pedestrians can be influenced in various ways by obstacles in different scenes, such as corridors, factories with goods, and temporarily placed signs. To analyze the impact of obstacles on pedestrian movement behavior, a single-layer perceptron is created.

$$p(X_t|\Theta_t) = \frac{1}{1 + \exp(-\alpha(\Theta_t; W_{ea}))} \tag{27}$$

In formula (27), $\alpha(\cdot)$ represents the single-layer perceptron, W_{ea} represents its weight, and Θ_t represents the observation context information at time step t. In this article, the raster map is used to represent a static environment, and the distance $d_{\min}^{(k)}$ from the nearest k occupied or unknown grids is used as the feature input for the context. We conducted simulation experiments on Matlab, and Fig. 5 shows the simulation effect according to formula (27).

Finally, the estimate of the probability distribution $p(X_t|X_t^{obs}, \Theta_t)$ is got. In this module, the maximum probability position of this estimate is taken as the output.

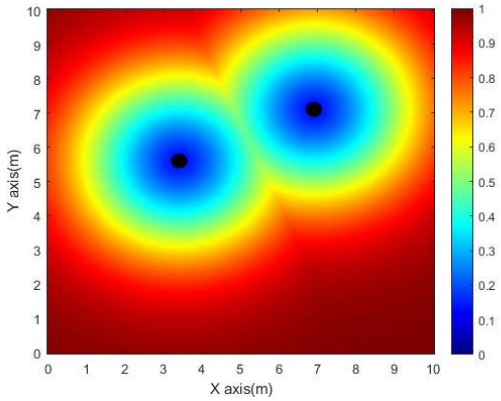


FIGURE 5. Static obstacle probability distribution, the black circle represents two cylindrical obstacles in the environment. Complete information about the obstacle environment can be obtained through the method of pre-mapping.

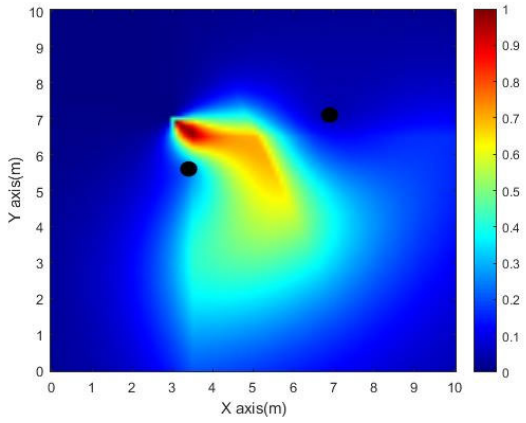


FIGURE 6. In the predicted future movement distribution of pedestrians, the estimated movement path is offset to the right due to the action of obstacles.

3) ESTIMATION OF MAXIMUM PROBABILITY POSITION

According to formulas (23) to (27), upon examining these formulas, it becomes evident that the probability estimation function, which needs to be solved, comprises numerous exponential terms. Because iterative methods, such as the Gauss-Newton method, require multiple calculations of the partial derivative and entail significant computational costs, particle swarm optimization (PSO) is employed to optimize the objective function [34].

PSO is used which is a bionic algorithm that emulates the behavior of biological populations. Each particle in the algorithm represents a potential numerical solution, ultimately facilitating the pursuit of the global optimum. This can be mathematically expressed as:

$$X_i^t = \left(x_i^t, y_i^t, \psi_i^t\right) \tag{28}$$

$$V_i^t = \left(v_{xi}^t, v_{yi}^t, v_{\psi i}^t\right) \tag{29}$$

where X_i and V_i represent the state and velocity of the particle i at time t.The state transition formula from t to $t+1$ for each

particle can be expressed as:

$$V_i^{t+1} = V_i^t + a_1 r_1 \left[p_i^t - X_i^t\right] + a_2 r_2 \left[p_g^t - X_i^t\right] \quad (30)$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (31)$$

where p_i^t represents the optimal position of the i particle in the search process, p_g^t represents the optimal position of all particle groups in the search process, a_1, a_2 represents the acceleration of the particles, and r_1, r_2 are random numbers between [0,1] obey a uniform distribution.

To prevent particles from searching over too wide or too narrow ranges during iteration, which could lead to missing the optimal value or getting trapped in a local optimum, upper and lower limits for particle velocity are defined. It can be expressed as $V_i^t \in [V_{\min}, V_{\max}]$.

Based on the above part, the maximum estimation of the pose probability of the pedestrian $\hat{Y}_t = \left(x_i^t, y_i^t\right)$ can be obtained at the next time, which is the output of the environmentally-constrained motion estimation module.

The prediction of pedestrian movement in time $t$modified by SOPD will be used to predict time step $t + 1$. Finally, the generator outputs $\hat{\mathbf{Y}} = \hat{Y}_1, \hat{Y}_2, \cdots, \hat{Y}_{pred\_len}$ as the final pedestrian trajectory prediction.

In the training stage, $\hat{\mathbf{Y}}$ will be input into the discriminator and conduct adversarial training through the GAN network model to improve the performance of the neural network.

F. DISCRIMINATOR

The discriminator module consists of a separate encoder. It takes the real trajectory and generator output as inputs, uses an LSTM cell and an MLP layer to identify the authenticity of the trajectories, and finally outputs the authenticity confidence of the trajectories:

G. LOSS FUN CTION

The overall loss function $L$of the model consists of two parts, $L_{variety}$ representing the minimum difference between the generated trajectory and the predicted trajectory, and $L_{GAN}(G, D)$ representing the training loss between the generator and the discriminator, forcing the generator to create a trajectory close to the real world. The loss function can be expressed as:

$$L = L_{GAN}(G, D) + \tau \cdot L_{variety} \quad (32)$$

$$L_{GAN}(G, D) = \min_G \max_D (E_{X\sim Pdata(X)}\left[\log D(X)\right] + E_{Z\sim Pdata(Z)}\left[\log\left(1 - D\left(G\left(Z\right)\right)\right)\right]) \quad (33)$$

$$L_{variety} = \min_k \left\| Y_i - \hat{Y}_i^{(k)} \right\|_2 \quad (34)$$

where τ represents the weight parameter that regulates the generator and discriminator, E represents the expectation, and k represents the sampling frequency of the generator. The loss function of discriminator takes the form of binary cross entropy function.

V. DYNAMIC ENVIRONMENT PATH PLANNING AND TRAJECTORY TRACKING

Mobile robot operations are constrained by various factors, including structural limitations, human safety concerns, and static obstacles. Although there are mature methods to address these concerns, planning a route that effectively coexists with the agent in a dynamic environment remains a challenge garnering significant attention from researchers.

This section presents a proposed dynamic environment path planning algorithm, known as Dynamic Window Approach (DWA), introduces a control method based on the Lyapunov direct method, and integrates a dynamic object trajectory prediction model to enable intelligent motion control of mobile robots in complex dynamic environments.

A. PATH PLANNING IN DYNAMIC ENVIRONMENT

1) PROBLEM DEFINITION

In the context of robot movement, the conventional approach to path planning for robots overlooks the distinctive movement patterns of pedestrians. Instead, pedestrians are considered as instantaneously stationary obstacles, leading to an inability to differentiate between dynamic objects and the static environment. To address this gap, the trajectory approach introduced in this section builds upon the principles of DWA [22]. Moreover, it incorporates a trajectory prediction module that takes into account the movements of pedestrians.

The motion state of the robot under the time t includes $[x, y, \theta, v, w]$, respectively, the horizontal coordinate, vertical coordinate, direction, linear velocity and angular velocity under the world coordinate. The displacement of the robot movement between two adjacent moments is relatively short, which can be approximated as a line segment. Then the state of the robot at $t + 1$ can be expressed as:

$$\theta_{t+1} = \theta_t + w_t \cdot \Delta t \quad (35)$$

$$x_{t+1} = x_t + v_t \cdot \Delta t \cdot \cos\left(\theta_{t+1}\right) \quad (36)$$

$$y_{t+1} = y_t + v_t \cdot \Delta t \cdot \sin\left(\theta_{t+1}\right) \quad (37)$$

2) IMPROVED DYNAMIC WINDOW APPROACH

Based on the robot motion model, the future trajectory of the robot can be calculated according to its motion speed. In this work, improved dynamic window approach (IDWA) is proposed. Our method simulates the motion trajectory and control speed of the robot in $t = 1 : T$, which is used as the trajectory planning of the motion control of the robot body.

Where, T is the same as the pedestrian track prediction $pred\_len$. Since there are infinite sets of motion velocities, the velocity samples are screened, and reasonably limit the velocity samples of the robot based on the motion and environmental constraints of the robot body.

Boundary speed limitation: The motion of mobile robot is restricted by hardware, and its motion speed is limited by the boundary speed, which is defined as:

$$V_b = \{v \in [v_{\min}, v_{\max}], w \in [w_{\min}, w_{\max}]\} \quad (38)$$

Kinetic limitation: In adjacent time steps, due to the dynamic constraints of the robot's motor, the robot has a maximum acceleration limit. The sampling speed of the robot should conform to the corresponding dynamic limit, that is, it satisfies:

$$V_d = \left\{ (v, w) \left| \begin{array}{c} v \in [v_c - \dot{v}_a \Delta t, v_c + \dot{v}_b \Delta t] \\ w \in [w_c - \dot{w}_a \Delta t, w_c + \dot{w}_b \Delta t] \end{array} \right. \right\} \tag{39}$$

where, v_c and $w_c$denote linear velocity and angular velocity, $\dot{v}_a$and$\dot{v}_b$denotes maximum deceleration and maximum acceleration of linear velocity, $\dot{w}_a$ and $\dot{w}_b$denotes maximum deceleration and maximum angular acceleration of angular velocity.

Turning radius limitation: The robot is limited by its moving structure, and its minimum turning radius $R$is constrained, so the speed sample $[v, w]$ of robot should satisfy the following requirements:

$$V_r = \left\{ (v, w) \left| \frac{v}{w} \geq R \right. \right\} \tag{40}$$

Therefore, the speed sample of the robot should simultaneously satisfy the above constraints:

$$V = V_b \cap V_d \cap V_r \tag{41}$$

3) EVALUATION FUNCTION

Based on the limitation of velocity samples, the feasible velocity sample space V is sampled to obtain the candidate trajectory of the array. In the next step, each velocity sample is needed to evaluate and select the optimal sample at the current moment. In this paper, pedestrian trajectory prediction is integrated into the evaluation function, which can be expressed as:

$$\begin{aligned} G^t(v, w) = & \alpha \cdot norm\_angle(v, w) \\ & + \beta \cdot goal\_angle(v, w) \\ & + \gamma \cdot goal\_dist(v, w) \\ & + \delta \cdot safe\_dist(v, w) \end{aligned} \tag{42}$$

$$norm\_angle(v, w) = \left| \theta_1^{plan} - \theta \right| \tag{43}$$

$$goal\_angle(v, w) = \left| \sum_{i=1}^{n} \arctan\left(\frac{y^{goal} - y_i^{plan}}{x^{goal} - x_i^{plan}} \right) - \theta_i^{plan} \right| \tag{44}$$

$$goal\_dist(v, w) = \min_{t=1:T} dist\left(X^{goal}, X_t^{plan} \right) \tag{45}$$

$$dist\left(X^a, X^b \right) = \sqrt{\left(X_x^a - X_x^b \right)^2 + \left(X_y^a - X_y^b \right)^2} \tag{46}$$

where $X_{t=1:T}^{plan}$, $X_t^{plan} = \left[x_t^{plan}, y_t^{plan}, \theta_t^{plan} \right]$denotes the future prediction state of the robot corresponding to the velocity sample, $X^{goal} = \left[x^{goal}, y^{goal} \right]$ denotes the position of the target point, $X_{t=1:T}^{pred}$ denotes the predicted trajectory of the dynamic obstacle, $X^{obstacle}$ denotes the coordinates of the nearest static obstacle of the robot, $norm\_angle(v, w)$ denotes the angle difference between the direction corresponding to the motion trajectory and the current direction of the robot to avoid sudden changes or oscillations in angles., $goal\_angle(v, w)$ denotes the angle difference between the motion trajectory and the direction of the target point. $goal\_dist(v, w)$ represents the shortest distance between the trajectory and the target point, $safe\_dist(v, w)$ represents the shortest distance between the trajectory and the obstacle, and $\alpha, \beta, \gamma, \delta$represents the weight.

In the DWA, the approach only considers the current obstacle distance and integrates the predicted trajectory of the dynamic obstacle into the evaluation function to calculate the possibility of collision in dynamic environments. This method considers the impact of dynamic targets on robot movement within a specific future timeframe. The shortest distance between the robot's planned trajectory and the pedestrian's motion trajectory is calculated, thus combining the prediction of pedestrian trajectories with robot path planning.

The future motion state of a reference robot is assumed as $X_t^{ref} = \left[x_t^{ref}, y_t^{ref}, \theta_t^{ref}, v_t^{ref}, w_t^{ref} \right]$, which respectively represent the reference motion state quantity of the robot at time step t.

The IDWA introduced above is used to calculate the trajectory of the reference robot in time $t = 1 : T$, which provides a reference for the trajectory tracking control of the mobile robot.

B. TRAJECTORY TRACKING

As shown in the Fig.7, (x, y, θ) is the current position state of the robot, (v, w) is the linear velocity and angular velocity of the robot, (x_d, y_d, θ_d) is the reference position state of an expected point in the reference trajectory X^{ref}, (v_d, w_d) is the reference speed of the reference point, and (x_e, y_e, θ_e) is the error component in the current coordinate system. The resulting pose error variance is:

$$\begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d - x \\ y_d - y \\ \theta_d - \theta \end{bmatrix} \tag{47}$$

The differential equation of pose error can be obtained by differentiating it by:

$$\begin{cases} \dot{x}_e = v_d \cos\theta_e - v + y_e w \\ \dot{y}_e = v_d \sin\theta_e - x_e w \\ \dot{\theta}_e = w_d - w \end{cases} \tag{48}$$

The goal of the trajectory tracking control module is to find the right linear and angular velocity for the robot and keep its pose error (x_e, y_e, θ_e) bounded.

Based on Lyapunov method [35] and combined with the above differential equation of car position and attitude error, this paper selects the Lyapunov function V:

$$V = \frac{k_1}{2} \left(x_e^2 + y_e^2 \right) + 2 \left(\sin\frac{\theta_e}{2} \right)^2 \geq 0 \tag{49}$$

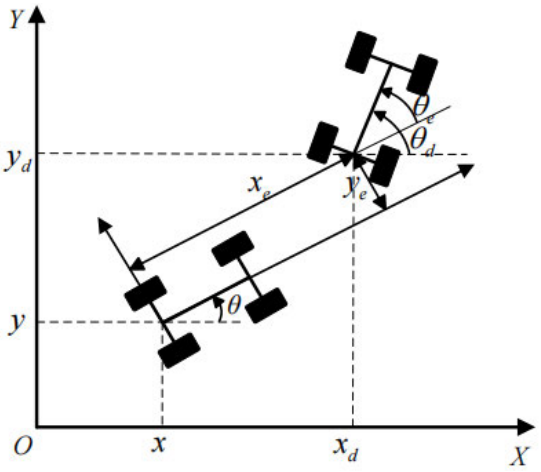


FIGURE 7. Kinematic model of mobile robot.

The control law is designed as:

$$\begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} v_d \cos\theta_e + k_2 x_e \\ w_d + k_1 v_d y_e + k_3 \sin\theta_e \end{bmatrix} \tag{50}$$

where $k_1, k_2, k_3 > 0$ are the control parameter. Take the derivative of V, and bring in the formula (48):

$$\begin{aligned} \dot{V} &= k_1 (x_e \dot{x}_e + y_e \dot{y}_e) + 2 \sin\frac{\theta_e}{2} \cos\frac{\theta_e}{2} \dot{\theta}_e \\ &= k_1 [x_e (v_d \cos\theta_e - v + y_e w) + y_e (v_d \sin\theta_e - x_e w)] \\ &\quad + (w_d - w) \sin\theta_e \\ &= k_1 [x_e v_d \cos\theta_e - v x_e] + k_1 y_e v_d \sin\theta_e \\ &\quad + (w_d - w) \sin\theta_e \\ &= -k_1 k_2 x_e^2 - k_3 \sin^2\theta_e \leq 0 \end{aligned} \tag{51}$$

It can be seen that the derivative of the Lyapunov function is semi-negative definite. According to Lyapunov's stability law [36], the control law of formula (48) can ensure the stability of the system, ensure that the pose error (x_e, y_e, θ_e) is bounded and convergent, and finally meet the robot's tracking of the reference trajectory.

VI. EXPERIMENTS

In this section, the experimental environment is introduced. Next, real-world experiments are conducted for the proposed dynamic object extraction algorithm. To validate our trajectory prediction model in this paper, a practical test is performed in an indoor setting and compared with the baseline model. The results demonstrate the advantages of our predicted model in the indoor setting. Using the proposed path planning algorithm, the robot's intelligent perception, decision-making, and control are successfully realized in the presence of pedestrians, thus demonstrating the effectiveness of our method in dynamic environments.

A. EXPERIMENTAL ENVIRONMENT

This study conducts experiments in a dynamic indoor environment. The mobile robot used is an Ackerman-type robot. The platform includes a JetsonTX1 device running the Ubuntu 18.04 operating system, a LiDAR, and an RGB camera. The LiDAR used is the RpLidar A1 developed by China's SLAMTEC Company, with a measuring distance of 12 meters and an angle measurement range of 270°.

Onboard control and sensor data acquisition of the mobile robot are carried out by an STM32F103 microcontroller, which communicates with the JetsonTX1 device through a UART interface. A distributed communication network is built based on the Robot Operating System (ROS), using a PC equipped with a GeForce RTX 3060Ti as the ROS network primary and the JetsonTX1 device as the secondary.

B. DYNAMIC OBJECT RECOGNITION EXPERIMENT

Experiment was carried out in the working scene of the indoor corridor. The communication network is built based on roscpp under ROS-Melodic. The point cloud clustering and matching processing were developed based on point cloud library (PCL), and finally generate independent function packages of ROS [37].

The feature pack subscribes to point cloud information published by LiDAR and robot position published by EKF fusion of IMU and odometer data. Through the recognition of dynamic objects, the pose information of target pedestrians is published at 5Hz.The experimental scene is shown in the Fig.9.

In the navigation process, the mobile robot first uses LiDAR to sense the indoor environment in real time, and then detects the dynamic objects in the field of view and tracks their motion trajectory.

The Fig.10 shows the point cloud data collected when the mobile robot is working.

In Fig. 10, the blue point cloud represents the static object, while the red point cloud represents the identified cluster of the dynamic object's point cloud. This cluster is utilized to detect the motion state of the dynamic object. The results demonstrate the effective recognition and extraction of dynamic targets using our method.

In order to verify the real-time performance of our method, 300 experiments are conducted to calculate the time cost required.

TABLE 1. Results of dynamic object recognition.

Average number of point clouds	Average time cost	Maximum time cost
4672	0.071 s	0.126 s

As shown in the Fig.10-11 and table 1, the average time cost of our recognition method is less than 0.1s, which can meet the pose update frequency of 10 Hz in most cases, and the maximum time cost is 0.126s, which can meet the update

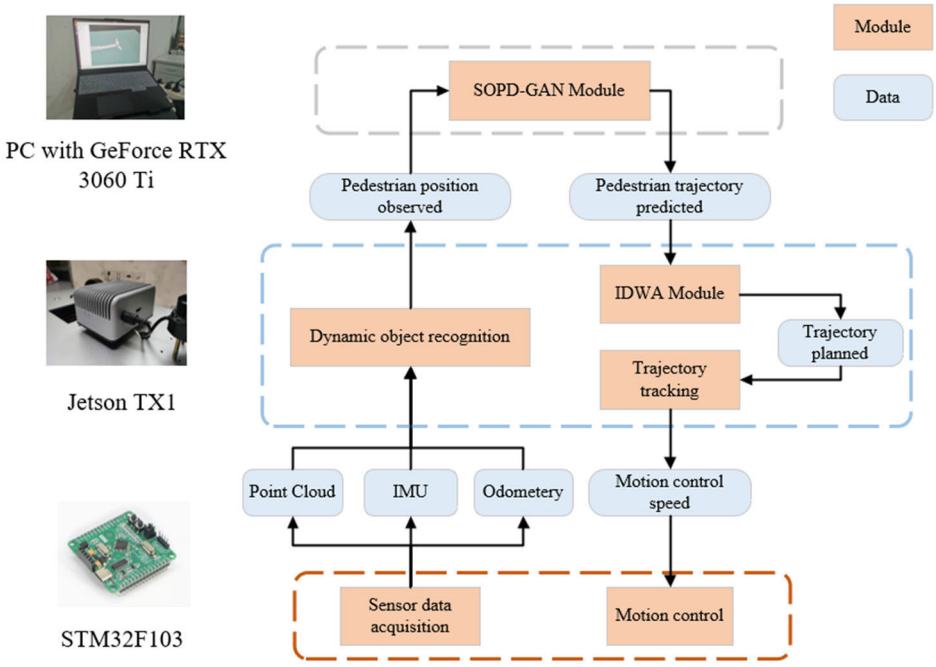


FIGURE 8. Architecture diagram of experimental platform. Our experimental platform comprises three modules. The STM32 module handles sensor data acquisition and actuator control, including motor driving. The JetsonTX1 module performs dynamic target recognition, path planning, and tracking calculations. The PC module is responsible for performing pedestrian trajectory prediction calculations, accelerated by the loaded GPU.

FIGURE 9. Experimental environment.

frequency of less than 5Hz. Our trajectory prediction module needs to meet the update frequency of 0.4s per frame, so this method can satisfy the needs of subsequent prediction tasks.

C. PEDESTRIAN TRAJECTORY PREDICITION

1) EVALUATION METRICS AND BASELINES

Similar to prior work [15], two error metrics, namely Average Displacement Error (ADE) and Final Displacement Error (FDE), are employed. However, as Zyner et al. [38] discussed, these commonly used indicators fail to consider outliers. Consequently, this can lead to a scenario where a prediction with a slightly incorrect direction is deemed equally erroneous as a prediction with a completely wrong direction, thereby producing significantly inferior results.

As such, Average Angle Error (AAE) is also included as an evaluation metric.

$$AAE = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{t_{pred}}\sum_{t=t_{obs}+1}^{t_{obs}+t_{pred}}\left|\arctan\left(\frac{y_i^t - y_i^{t-1}}{x_i^t - x_i^{t-1}}\right) - \arctan\left(\frac{\hat{y}_i^t - \hat{y}_i^{t-1}}{\hat{x}_i^t - \hat{x}_i^{t-1}}\right)\right| \tag{52}$$

AAE reflects the average angular deviation between the true and predicted trajectory directions at all time steps during the prediction process.

Social GAN [18] is used as a baseline model to compare with our model, which is the state of the art method.

2) EXPERIMENTAL DETAILS

Similar to SGAN [18], the last 8 timesteps track sequence is used as the observation input and predicted the next 12 timesteps track sequence as the model output at a frame rate of 0.4 seconds. That is, 3.2 seconds of historical data as the input, and then output the predicted state of 4.8 seconds.

The model proposed in this paper is built based on pytorch [39]. The hidden state dimension of the encoder is

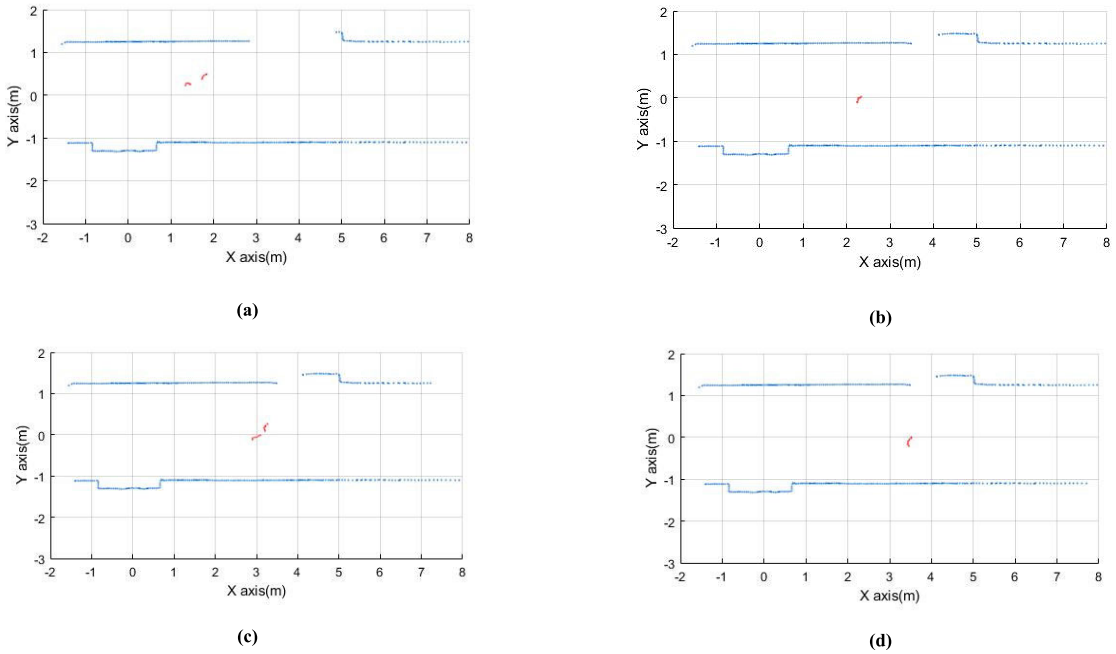


FIGURE 10. Results of dynamic object recognition. The experiment was carried out in an indoor corridor environment, in which blue point clouds represent static objects and red point clouds represent recognized moving objects. Lidar is located at the origin. With the movement of pedestrians, the static background is blocked to varying degrees.

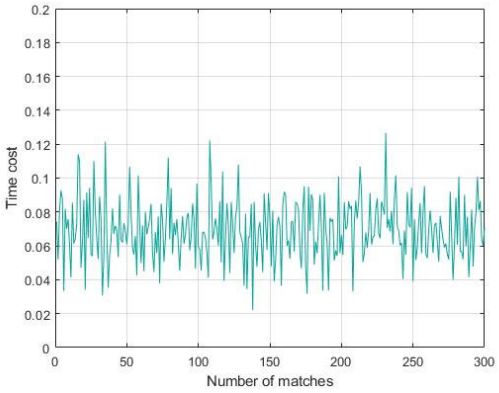


FIGURE 11. Time cost of experiment.

16 and the hidden state dimension of the decoder is 32. The Adam optimizer is used for 200 echo trainings, where the initial learning rate was 0.001. In addition, our training process is carried out on the PC equipped with GeForce RTX 3060Ti.

During the experiment, the mobile robot side collects the surrounding environment and its own state in real time, processes it through the dynamic object recognition module, obtains the real-time pose of the pedestrian and publishes it in the Topic published to the ROS network. The host side subscribes the Topic information, and prediction of the future pedestrian trajectory is accelerated the operation of the neural network by GPU. The pedestrian prediction trajectory is published to the ROS Topic in nav_msgs/Path data format and updated frequency is 0.4 seconds for each frame.

3) RESULTS AND DISCUSSION

To verify the validity of the model proposed in this paper, the performance of our model is compared with baseline in the indoor scene experiment, the experimental results are shown in Fig. 12.

As can be seen from the Fig.12 (f), compared with Social-GAN, our model takes environmental information into consideration for pedestrian trajectory prediction. In the experiment, the comparison algorithm predicted that the pedestrian would tend to keep the original motion state and move towards the corridor wall, leading to the failure of trajectory prediction. Our model combines the static obstacle information, revises the historical trajectory and obtains the predicted trajectory, which can realize the pedestrian trajectory prediction in the static environment. Meanwhile, the experimental results are also shown in Table 2:

TABLE 2. Comparison of trajectory prediction.

Metrics	ADE	FDE	AEE
Social-GAN	**0.0706**	0.2854	0.2048
Ours	0.0735	**0.0881**	**0.0619**

Our model performs on FDE and AEE metrics, and in particular on AEE, our approach has significant advantages over the baseline model. As can be seen from the results, our

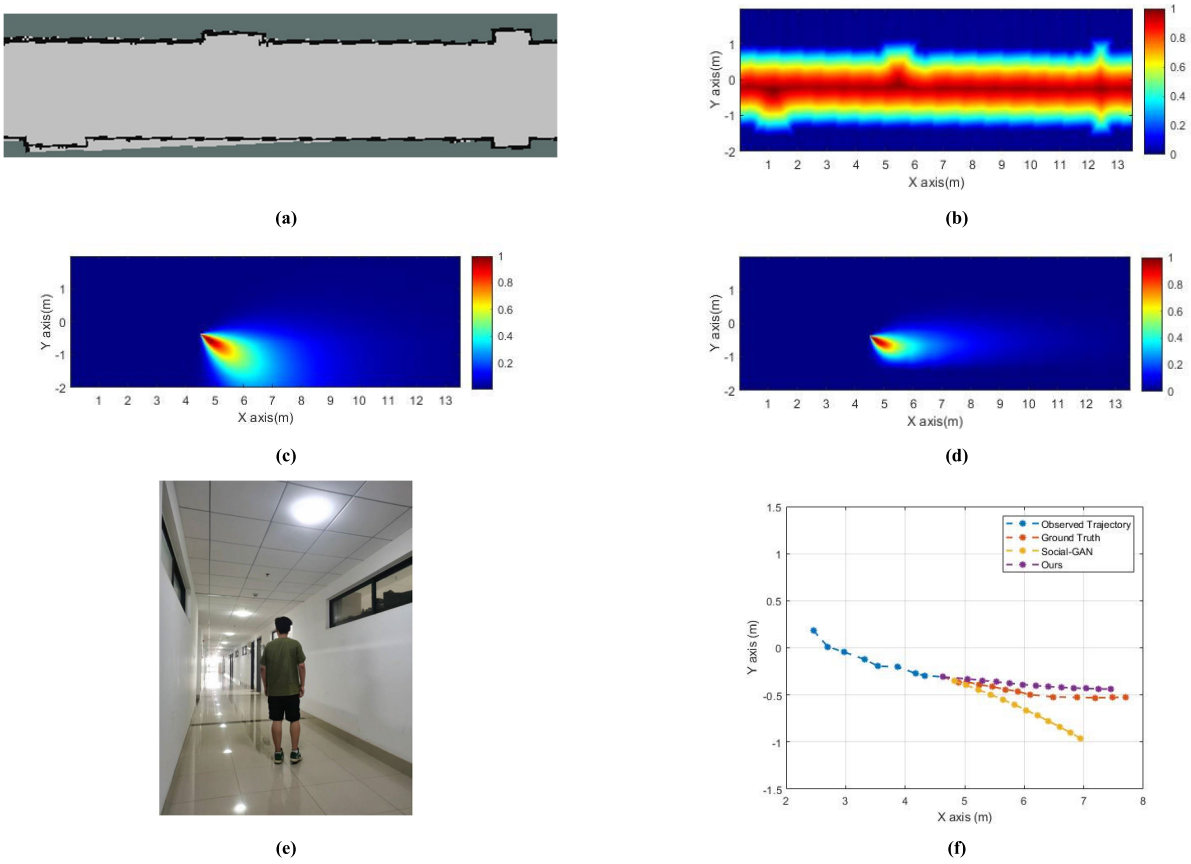


FIGURE 12. Results of pedestrian trajectory prediction experiment. (a) Grid map of static environment. (b) Thermal diagram of probability description of static obstacles. (c) Probability distribution of pedestrian motion state. (d) Description of Future Trajectory after Fusion of Pedestrian Kinematics and Environmental Constraints. (e) Experimental scene. (f) Comparison of prediction performance between SGAN and our model in experiments. Among them, the blue line represents the observation trajectory, and the red line represents the real trajectory of pedestrians' future movement. The yellow line represents the prediction result of SGAN, and the purple line represents the prediction output of our method.

method shows certain advantages in scenarios with abundant environmental information.

In comparison with the trend estimation of the movement direction, the environment on the pedestrian movement trend is taken into account which effectively provides a more accurate motion estimation, which has a significant impact on the robot's motion in a dynamic environment.

D. PATH PLANNING AND TRAJECTORY TRACKING

In this section, experiments on mobile robot motion in dynamic environment is conducted. Based on the pedestrian trajectory prediction module in the previous section, the dynamic motion function of the mobile robot is verified in a unified experimental scene, and our proposed IDWA method is compared with DWA.

1) EXPERIMENTAL DETAILS

The path planning and trajectory tracking module of the mobile robot is developed based on rospy and is deployed and run in JetsonTX1 which is mounted on the mobile robot. The path planning module subscribes to the predicted trajectory of SOPD-GAN module and the pose state of the robot.

The update frequency of the planned path module is in 1Hz. At the same time, the trajectory tracking module published the control data $[v, w]$ at a frequency of 10Hz, so as to realize the tracking of the planned trajectory of the robot.

2) RESULTS

Based on the prediction of the future trajectory of dynamic objects, the path planning and trajectory tracking experiments of mobile robots are carried out. In the experiment, the mobile machine is made start from point A and move to point B autonomously. The linear distance between the two points was 10 m, as shown in the Fig.13.

As can be seen from the Fig.13 (a), at the moment T=1, since the future trajectory of the pedestrian is taken into consideration, the robot determines that there will be collision risk when going straight, so it actively avoids to the bottom. Although this route is not the fastest route to reach the target, it is a reasonable strategy in the dynamic scene.

At the T=5, based on the pedestrian movement trajectory, the robot chooses to carry out the deceleration-turn strategy. By reducing its own speed, the robot allows the pedestrian to pass preferentially, successfully obtains the free movement

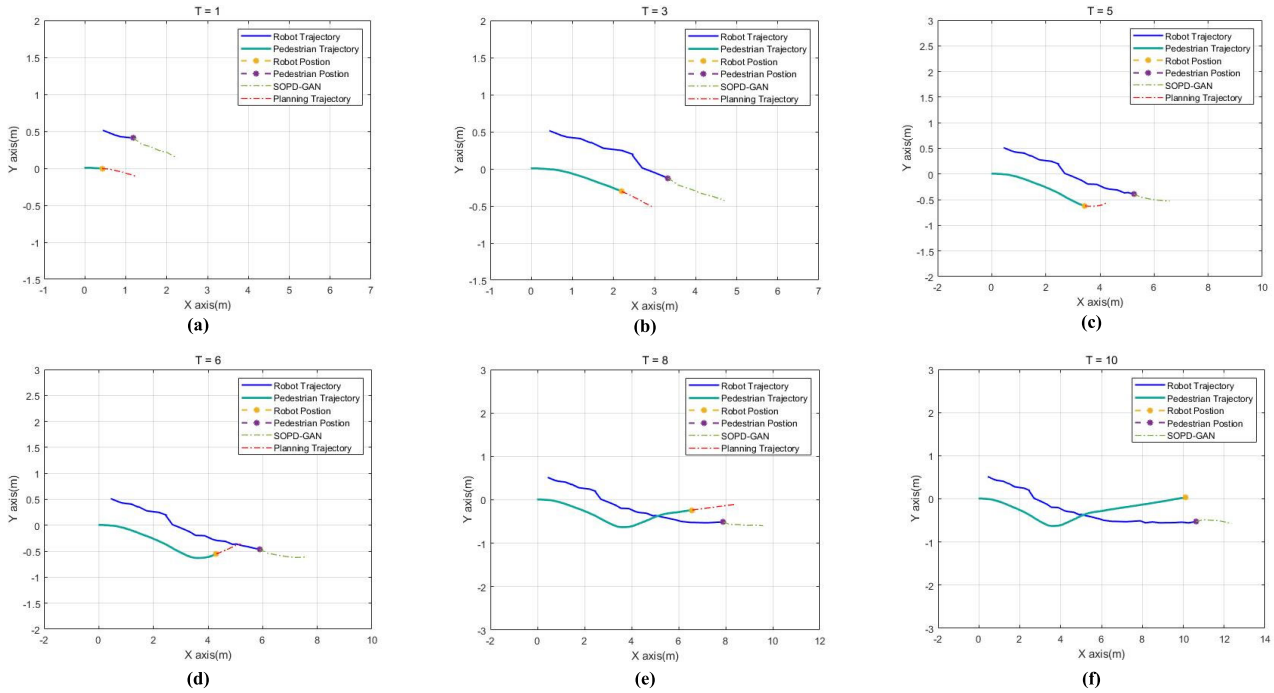


FIGURE 13. Results of path planning and trajectory tracking experiment. The purple line represents the historical track of the pedestrian, and the cyan line represents the historical track of the robot. The light green dashed line represents the predicted trajectory of the pedestrian in the next 12 frames of the SOPD-GAN, and the red dashed line represents the future planned trajectory of the robot itself. It can be seen from (b), (c), and (e) that our method plans a motion strategy of deceleration avoidance and acceleration for the robot.

space at the time T=6, and starts to carry out the dynamic target obstacle avoidance task. At the time T=8, the robot completes the obstacle avoidance task, obtains the conditions to accelerate to the target point B, and finally reaches the point B at the time T=10, completing the motion task.

In the experiment, parameters of path planning and motion control are shown in the table 3 -5:

TABLE 3. Mobile robot hardware parameters.

Quantity	Value
Minimum linear velocity (m/s)	-0.2
Minimum linear velocity (m/s)	0.5
Maximum linear acceleration (m/s^2)	0.6
Minimum angular velocity (rad/s)	-0.3
Maximum angular velocity (rad/s)	0.3
Maximum angular acceleration (rad/s^2)	20

From the results, it can be seen that our method combined with obstacle trajectory prediction can independently propose strategies such as advance steering and deceleration avoidance, which significantly improves the robot's working performance.

TABLE 4. Path planning evaluation function parameters.

α	β	γ	δ
-1	-1	-2	-1

TABLE 5. Lyapunov direct method parameters.

k_1	k_2	k_3
1	0.2	1

The basic task of the mobile robot is to reach the target place in time without colliding with the dynamic obstacles and maintaining a certain safe distance.

In the experiment, comparative experiments are conducted using traditional DWA and the path planning algorithm based on improved DWA proposed in this paper, and the results are shown as follows:

The Fig.14 shows the position and course Angle changes of the mobile robot during the experiment.

As can be seen from the Fig.14, our method realizes the existence of dynamic obstacles earlier, the heading Angle

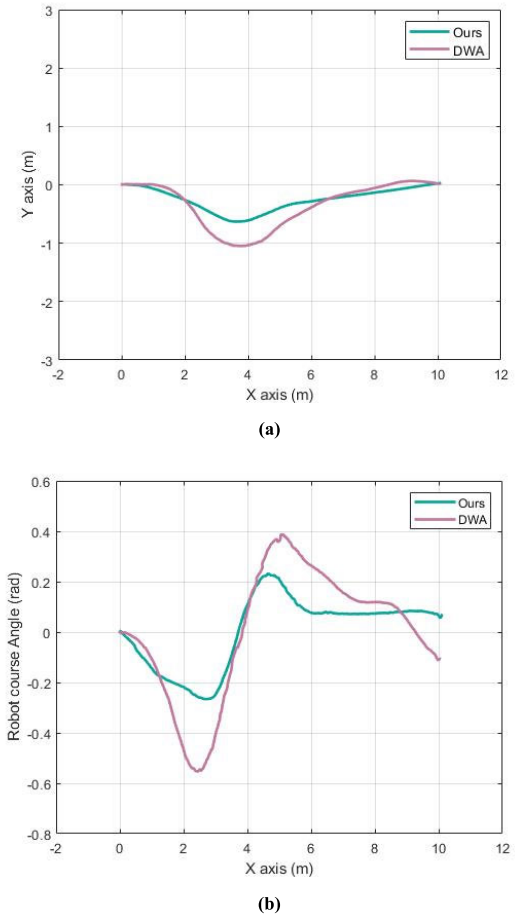


FIGURE 14. Results of mobile robot obstacle avoidance experiment in dynamic environment. Our method is compared with DWA, and the blue line represents our method and the purple represents DWA. (a) Comparison of motion paths of mobile robots. (b) Comparison of heading angles of robots in motion.

begins to change, and the medium response robot begins to avoid obstacles.

TABLE 6. Obstacle avoidance experiment result.

Quantity	Our Method	Our Method
Average pedestrian speed (m/s)	0.4	0.4
Navigation time cost(s)	28.8	32.4
Total distance of motion (m)	11.2	12.9
Minimum distance from pedestrians (m)	0.73	0.32

As can be seen from the table 6, our method can reach the destination in less time and shorter driving distance under the same conditions. At the same time, it can be found that our approach maintains a sufficient safe distance from pedestrians.

Due to the early start of avoiding obstacles, our overall path is smoother. The fluctuation of the heading Angle is always maintained at a relatively stable value, which ensures the smoothness of the robot's movement.

By incorporating the future trajectory of dynamic targets into path planning, the performance of traditional reactive path planning algorithm is improved in dynamic scenarios.

VII. CONCLUSION

In this paper, we propose a framework for motion robots operating in dynamic environments. We verify the feasibility and performance of this method through experiments. The proposed methods encompass dynamic object recognition and tracking, prediction of dynamic object trajectories, path planning in dynamic environments, and motion control.

Our object recognition algorithm is implemented using LiDAR technology. By leveraging the proposed hypothesis on the spatial distribution of dynamic point clouds, we match the acquired point cloud data with the static point cloud to extract clusters of interesting points, enabling the retrieval of location information for dynamic objects.

To address the trajectory prediction problem of dynamic agents, we introduce SOPD-GAN. This approach transforms the prediction problem into a decision problem by estimating the kinematic properties of pedestrians and considering the effects of static obstacles as potential variables. Integrated with LSTM, this module incorporates movement environment information and the object's historical trajectory into the network prediction stage. As the motion of the agent is characterized by uncertainty and high intelligence, we introduce GAN into the model architecture to generate reasonable trajectory predictions using data-driven methods, thereby simulating multi-modal trajectories. he results show that this method can construct a suitable prediction model with high accuracy. Specifically, our method achieved an accuracy of 0.0881 and 0.0691 in FDE and AEE of predicting pedestrian trajectory, surpassing the baseline method by 20% and 14%.

Building upon the trajectory prediction of dynamic obstacles, we propose IDWA, which accounts for the future trajectory avoidance task of dynamic objects. The future trajectory of the mobile robot is evaluated against different speed samples alongside the trajectory of dynamic obstacles. This evaluation yields future path planning for the mobile robot in a time-varying environment.

The experimental results demonstrate the feasibility of the proposed framework in dynamic robot environments. The trajectory prediction model accounting for environmental constraints proposed in this paper proves effective in scenarios with static obstacles, outperforming the representative baseline model across key evaluation indicators.

Traditional mobile robots are typically based on preprogrammed or reactive modes of operation. With ongoing research and increasing demand, there is an urgent need to enhance the intelligence of robots. In dynamic scenes, robots must differentiate between static obstacles and dynamic objects. The handling method of dynamic objects, to some

extent, determines the depth of integration that robots can achieve in human society. This area of research presents interesting and challenging opportunities. Future work will focus on optimizing the prediction model and resolving the path planning problem in dynamic robot environments, aiming to improve the intelligence of robots in scenarios involving interactions and movement patterns with pedestrians.

REFERENCES

[1] M. Mahendru and S. K. Dubey, "Real time object detection with audio feedback using YOLO vs. YOLO_v3," in *Proc. 11th Int. Conf. Cloud Comput., Data Sci. Eng. (Confluence)*, Noida, India, Jan. 2021, pp. 734–740.

[2] G. Pengcheng, L. Zheng, and W. Jingjing, "Radar group target recognition based on HRRPs and weighted mean shift clustering," *J. Syst. Eng. Electron.*, vol. 31, no. 6, pp. 1152–1159, Dec. 2020.

[3] H. Chen and X. Guo, "Multi-scale feature fusion pedestrian detection algorithm based on transformer," in *Proc. 4th Int. Conf. Comput. Vis., Image Deep Learn. (CVIDL)*, Zhuhai, China, May 2023, pp. 536–540.

[4] R. Korbmacher and A. Tordeux, "Review of pedestrian trajectory prediction methods: Comparing deep learning and knowledge-based approaches," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 24126–24144, Dec. 2022.

[5] P. Zhang, J. Xue, P. Zhang, N. Zheng, and W. Ouyang, "Social-aware pedestrian trajectory prediction via states refinement LSTM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2742–2759, May 2022.

[6] D. Zhao, T. Li, X. Zou, Y. He, L. Zhao, H. Chen, and M. Zhuo, "Multimodal pedestrian trajectory prediction based on relative interactive spatial–temporal graph," *IEEE Access*, vol. 10, pp. 88707–88718, 2022.

[7] G. Chen, L. Hu, Q. Zhang, Z. Ren, X. Gao, and J. Cheng, "ST-LSTM: Spatio-temporal graph based long short-term memory network for vehicle trajectory prediction," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Abu Dhabi, United Arab Emirates, Oct. 2020, pp. 608–612.

[8] S. Kwag, B. Kang, W. Kim, and Y. Hwang, "Chebyshev transform-based robust trajectory prediction using recurrent neural network," *IEEE Access*, vol. 10, pp. 130397–130405, 2022.

[9] R. Möller, A. Furnari, S. Battiato, A. Härmä, and G. M. Farinella, "A survey on human-aware robot navigation," *Robot. Auto. Syst.*, vol. 145, Nov. 2021, Art. no. 103837.

[10] H. Cheng, F. T. Johora, M. Sester, and J. P. Müller, "Trajectory modelling in shared spaces: Expert-based vs. deep learning approach?" in *Proc. Int. Workshop Multi-Agent Syst. Agent-Based Simulation*. Cham, Switzerland: Springer, 2020, pp. 13–27.

[11] A. Bighashdel and G. Dubbelman, "A survey on path prediction techniques for vulnerable road users: From traditional to deep-learning approaches," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Auckland, New Zealand, Oct. 2019, pp. 1039–1046.

[12] S. Paris, J. Pettré, and S. Donikian, "Pedestrian reactive navigation for crowd simulation: A predictive approach," *Comput. Graph. Forum*, vol. 26, no. 3, pp. 665–674, Sep. 2007.

[13] Q. Xu, M. Chraibi, A. Tordeux, and J. Zhang, "Generalized collision-free velocity model for pedestrian dynamics," *Phys. A, Stat. Mech. Appl.*, vol. 535, Dec. 2019, Art. no. 122521.

[14] M. Meuter, U. Iurgel, S.-B. Park, and A. Kummert, "The unscented Kalman filter for pedestrian tracking from a moving host," in *Proc. IEEE Intell. Vehicles Symp.*, Eindhoven, The Netherlands, Jun. 2008, pp. 37–42.

[15] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 961–971.

[16] A. Azzouni and G. Pujolle, "A long short-term memory recurrent neural network framework for network traffic matrix prediction," 2017, *arXiv:1705.05690*.

[17] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Vancouver, BC, Canada, May 2013, pp. 6645–6649.

[18] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GAN: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 2255–2264.

[19] S. Eiffert, K. Li, M. Shan, S. Worrall, S. Sukkarieh, and E. Nebot, "Probabilistic crowd GAN: Multimodal pedestrian trajectory prediction using a graph vehicle-pedestrian attention network," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 5026–5033, Oct. 2020.

[20] R. Akabane and Y. Kato, "Pedestrian trajectory prediction based on transfer learning for human-following mobile robots," *IEEE Access*, vol. 9, pp. 126172–126185, 2021.

[21] Y. Li, B. Tian, Y. Yang, and C. Li, "Path planning of robot based on artificial potential field method," in *Proc. IEEE 6th Inf. Technol. Mechatronics Eng. Conf. (ITOEC)*, Chongqing, China, Mar. 2022, pp. 91–94.

[22] Z. Zhu, J. Xie, and Z. Wang, "Global dynamic path planning based on fusion of A* algorithm and dynamic window approach," in *Proc. Chin. Autom. Congr. (CAC)*, Hangzhou, China, Nov. 2019, pp. 5572–5576.

[23] W. Zhi-Wen, L. M. Kun, and W. Li-jing, "Path planning for UUV in dynamic environment," in *Proc. 9th Int. Symp. Comput. Intell. Design (ISCID)*, Hangzhou, China, Dec. 2016, pp. 211–215.

[24] C. Zhao, G. G. Yen, Q. Sun, C. Zhang, and Y. Tang, "Masked GAN for unsupervised depth and pose prediction with scale consistency," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 12, pp. 5392–5403, Dec. 2021.

[25] L. Ye, Z. Wang, X. Chen, J. Wang, K. Wu, and K. Lu, "GSAN: Graph self-attention network for learning spatial–temporal interaction representation in autonomous driving," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9190–9204, Jun. 2022.

[26] J. Woo, H. Jeong, and H. Lee, "Comparison and analysis of LiDAR-based SLAM frameworks in dynamic environments with moving objects," in *Proc. IEEE Int. Conf. Consum. Electron.-Asia*, Nov. 2021, pp. 1–4.

[27] S. Chen, J. Liu, T. Wu, W. Huang, K. Liu, D. Yin, X. Liang, J. Hyyppä, and R. Chen, "Extrinsic calibration of 2D laser rangefinders based on a mobile sphere," *Remote Sens.*, vol. 10, no. 8, p. 1176, Jul. 2018.

[28] M. Magnusson, "The three-dimensional normal-distributions transform: An effificient representation for registration surface analysis and loop detection," Ph.D. dissertation, Örebro Univ., Örebro, Sweden, 2009.

[29] H. Li, D. Qian, G. Liu, and W. Zhao, "Research on high precision point cloud registration algorithm in dynamic environment," in *Proc. 3rd Int. Conf. Comput., Control Robot. (ICCCR)*, Shanghai, China, 2023, pp. 87–94.

[30] E. Bayram and V. Nabiyev, "Image segmentation by using K-means clustering algorithm in Euclidean and Mahalanobis distance calculation in camouflage images," in *Proc. 28th Signal Process. Commun. Appl. Conf. (SIU)*, Gaziantep, Turkey, Oct. 2020, pp. 1–4.

[31] L. Knoedler, C. Salmi, H. Zhu, B. Brito, and J. Alonso-Mora, "Improving pedestrian prediction models with self-supervised continual learning," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4781–4788, Apr. 2022.

[32] M. Costa, V. Koivunen, and H. V. Poor, "Estimating directional statistics using wavefield modeling and mixtures of von-Mises distributions," *IEEE Signal Process. Lett.*, vol. 21, no. 12, pp. 1496–1500, Dec. 2014.

[33] E. A. Abdullah, I. A. Saleh, and O. I. A. Saif, "Performance evaluation of parallel particle swarm optimization for multicore environment," in *Proc. Int. Conf. Adv. Sci. Eng. (ICOASE)*, Duhok, Iraq, Oct. 2018, pp. 81–86.

[34] L. Ling, Y. Niu, and H. Zhu, "Lyapunov method-based collision avoidance for UAVs," in *Proc. 27th Chin. Control Decis. Conf. (CCDC)*, Qingdao, China, May 2015, pp. 4716–4720.

[35] A. Kumar and S. K. Bhagat, "Application of Lyapunov's method for multi-machine power system stability analysis," in *Proc. 2nd Int. Conf. Adv. Electr., Comput., Commun. Sustain. Technol. (ICAECT)*, Bhilai, India, Apr. 2022, pp. 1–6.

[36] Z. Ma, L. Zhu, P. Wang, and Y. Zhao, "ROS-based multi-robot system simulator," in *Proc. Chin. Autom. Congr. (CAC)*, Hangzhou, China, Nov. 2019, pp. 4228–4232.

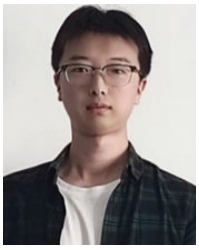
[37] M. Miknis, R. Davies, P. Plassmann, and A. Ware, "Near real-time point cloud processing using the PCL," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, London, U.K., Sep. 2015, pp. 153–156.

[38] A. Zyner, S. Worrall, and E. Nebot, "Naturalistic driver intention and path prediction using recurrent neural networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1584–1594, Apr. 2020.

[39] M. J. Rasch, D. Moreda, T. Gokmen, M. Le Gallo, F. Carta, C. Goldberg, K. E. Maghraoui, A. Sebastian, and V. Narayanan, "A flexible and fast PyTorch toolkit for simulating training and inference on analog crossbar arrays," in *Proc. IEEE 3rd Int. Conf. Artif. Intell. Circuits Syst. (AICAS)*, Washington, DC, USA, Jun. 2021, pp. 1–4.

HAO LI (Student Member, IEEE) received the B.S. degree from the School of Mechanical and Automotive Engineering, Shanghai University of Engineering Science, in 2020. He is currently pursuing the M.S. degree with the Department of Precision Machinery, Shanghai University, Shanghai, China. His current research interests include environment perception, path planning, and control of mobile robots in dynamic environments.

DONG-HAI QIAN received the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, in 1999. He is currently an Associate Professor with the School of Mechatronic Engineering and Automation, Shanghai University. His research interests include robot control and SLAM of mobile robots.



GUANG-YIN LIU received the B.S. degree from the Department of Mechanical Engineering, Qingdao University of Technology, in 2020. He is currently pursuing the M.S. degree with the Department of Precision Machinery, Shanghai University. His current research interests include mobile robot SLAM, path tracking, and drive control.

ZE CUI received the Ph.D. degree from the Harbin Institute of Technology, Harbin, in 2002. He is currently an Associate Professor with the School of Mechatronic Engineering and Automation, Shanghai University. His research interests include bionic robot and mechatronics.

JING-TAO LEI received the Ph.D. degree from Beihang University, Beijing, in 2007. She is currently a Professor with the School of Mechatronic Engineering and Automation, Shanghai University. Her research interests include medical robot and bionic robot.