Machine Intelligence Research
www.mi-research.net

22(4), August 2025, 713-729
DOI: 10.1007/s11633-024-1537-x

Multimodal Pretrained Knowledge for Real-world Object Navigation

Hui Yuan[1,2,3] Yan Huang[3] Naigong Yu[1,2] Dongbo Zhang[4]
Zetao Du[3,5] Ziqi Liu[1] Kun Zhang[3]

[1]School of Information Science and Technology, Beijing University of Technology, Beijing 100124, China

[2]Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing University of Technology, Beijing 100124, China

[3]New Laboratory of Pattern Recognition (NLPR) & State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS), Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

[4]School of Automation and Electronic Information, Xiangtan University, Xiangtan 411105, China

[5]School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China

Abstract: Most visual-language navigation (VLN) research focuses on simulate environments, but applying these methods to real-world scenarios is challenging because of misalignments between vision and language in complex environments, leading to path deviations. To address this, we propose a novel vision-and-language object navigation strategy that uses multimodal pretrained knowledge as a cross-modal bridge to link semantic concepts in both images and text. This improves navigation supervision at key-points and enhances robustness. Specifically, we 1) randomly generate key-points within a specific density range and optimize them on the basis of challenging locations; 2) use pretrained multimodal knowledge to efficiently retrieve target objects; 3) combine depth information with simultaneous localization and mapping (SLAM) map data to predict optimal positions and orientations for accurate navigation; and 4) implement the method on a physical robot, successfully conducting navigation tests. Our approach achieves a maximum success rate of 66.7%, outperforming existing VLN methods in real-world environments.

Keywords: Visual-and-language object navigation, key-points, multimodal pretrained knowledge, optimal positions and orientations, physical robot.

Citation: H. Yuan, Y. Huang, N. Yu, D. Zhang, Z. Du, Z. Liu, K. Zhang. Multimodal pretrained knowledge for real-world object navigation. *Machine Intelligence Research*, vol.22, no.4, pp.713–729, 2025. http://doi.org/10.1007/s11633-024-1537-x

1 Introduction

Service robots, which enable human-robot interactions, are widely used in indoor environments. In particular, vision-language object navigation tasks require the robot to locate remote objects on the basis of natural language instructions via its perception and understanding of the visual environment.

Visual navigation research can be broadly classified into two approaches. The first is map-dependent navigation[1–3], which relies on map accuracy and does not require external supervision. The second is reinforcement learning-based navigation[4–6], where robots autonomously interact with the environment to collect and store data. However, this approach faces challenges, including high exploration costs and limited generalizability when transitioning from simulation to real-world environments.

Current research on visual-language navigation (VLN) focuses primarily on simulation platforms. Anderson et al.[7] introduced the VLN with the room-to-room (R2R) dataset, and successfully applied it to the Matterport3D Simulator[8]. Recent advances, including self-monitoring agents[9], NvEM[10], RecBert[11], BEVbert[12], Landmark RxR[13], and REVERIE[14], have enhanced cross-modal action matching, visual-language pretraining, and map-based navigation. However, these innovations remain confined to simulations, which differ significantly from real-world environments. In real-world environments, challenges such as dynamic or complex key locations, large-scale spaces, and complex decision-making processes hinder model generalization.

Few studies have explored the application of VLNs in real indoor environments. Hong et al.[15] introduced a candidate waypoint predictor to bridge the gap between discrete and continuous environments, advancing VLN generalization to the real world. Anderson et al.[16] successfully translated the discrete actions of a VLN agent into

Research Article
Special Issue on Embodied Intelligence
Manuscript received on September 24, 2024; accepted on December 23, 2024; published online on June 26, 2025
Recommended by Guest Editor Zhichuang Wang
Colored figures are available in the online version at https://link.springer.com/journal/11633

continuous actions on a real robot via the TurtleBot 2 platform. However, these methods still face generalization challenges in complex or dynamic environments, particularly in regard to distinguishing key semantic concepts in visual and language instructions, such as differentiating similar objects or capturing the semantics of occluded objects at critical navigation points.

Multimodal pretraining techniques face several challenges when they are generalized to real-world vision-language object navigation: 1) Robots struggle to understand the relationships between semantic concepts in complex and dynamically changing scenes at key navigation points, leading to reduced robustness in object retrieval; 2) Existing cross-modal alignment-based VLN methods require extensive supervised training for real-world testing, leading to prohibitive deployment costs. Recently, Huang et al.[17] developed multimodal alignment concept knowledge (MACK) to improve unpaired image-text matching accuracy. This knowledge directly links objects to text without model training, providing highly generalizable semantic concepts for easy application in unknown environments. Despite its potential, no research has yet applied it to solve these challenges.

In this study, we propose a novel visual-language object navigation method that leverages multimodal pretrained knowledge and cross-modal alignment between vision and language at critical navigation points to effectively guide robot navigation. Specifically, 1) we collect word-region pairs for 100 categories of common indoor objects via the Open Images V7 dataset and real-world laboratory data. For each word, we compute its prototype region representation by averaging the associated region representations, creating the indoor object knowledge MACK (indoor). 2) We randomly generate waypoints within a specific density range, optimizing them for challenging locations as key-points. MACK (indoor) is used at these key-points for visual-language matching to supervise real-world robot navigation. 3) To enable the robot to reach the optimal navigation position near the target object, we propose a target navigable position prediction method that combines the target's depth information with SLAM map data to predict the optimal position and orientation. 4) This algorithm was implemented on a physical robot and successfully passed visual-language object navigation tests. The results demonstrate its superior performance compared with existing methods.

The main contributions of this research are as follows:

1) We propose a real-world object navigation method driven by multimodal pretrained knowledge, which effectively supervises robot navigation at key-points.

2) We collect an indoor object database and develop MACK (indoor) to support object retrieval across various scenarios.

3) We propose a target position prediction strategy that accurately predicts the optimal position and orientation for the robot to approach the target object.

4) We implement the method on a physical robot, successfully conducting navigation tests that demonstrate its superior performance over existing methods.

2 Related work

2.1 Visual-language navigation

Visual-language navigation has made great progress in the simulator environment. The research methods for VLN have become increasingly diverse. In terms of the feature extraction network structure, it has progressed from LSTM-based[9, 18] to transformer-based[19–22]. The navigation action space is divided into discrete[7, 14] and continuous[15, 23–25] environments. Researchers have enhanced the alignment between natural language instructions and ground truth paths through action strategy learning[26–29] and multimodal representation learning[10, 30–33]. Qi et al.[14] first proposed the REVERIE benchmark to promote intelligent agents to identify a remote object in real indoor environments. Cui et al.[34] proposed a grounded entity-landmark adaptive (GELA) pretraining paradigm to learn fine-grained cross-modal semantic alignment between entity phrases and environment landmarks. However, these methods all rely on simulator environment implementation. Few works have explored real-world object navigation tasks. Batra et al.[35] revisited the evaluation of embodied agents for navigating to objects. Deitke et al.[36] developed a platform for object semantic navigation with simulation-to-real transfer in a supervised learning manner. Recently, a few studies[37, 38] have explored zero-shot object navigation. Inspired by [34], we propose a multimodal pretrained knowledge based on real-world environments to facilitate cross-modal alignment between entity phrases and objects at navigation key-points.

2.2 Multimodal knowledge

Some works have used unaligned multimodal knowledge for vision and language understanding tasks. For example, Zhu et al.[39] built heterogeneous graphs corresponding to the visual, semantic and factual features for visual question answering, and Wang et al.[40] combined visual and textual knowledge to find discriminative parts for few-shot learning. In the framework of multimodal aligned knowledge representation, Huang et al.[17] proposed a multimodal alignment concept knowledge (MACK) method, which explicitly stores paired word-visual prototype representations for unpaired image-text matching. In addition, some works have proposed for the use of multimodal knowledge for visual language navigation. For example, Li et al.[41] proposed the emphasized reasoning model (KERM), which uses the factual information of associated navigation scenes in an external knowledge graph to match the current context of the environ-

ment, enhancing the model's understanding and decision-making capabilities for complex navigation scenes. Lin et al.[42] proposed a memory model based on a multimodal transformer, which integrates visual and language information into a variable-length memory module to provide support for the robot's long-term decision-making. An et al.[12] proposed BEVBert, which enables robots to adaptively build semantic-spatial map memory. Different from them, we construct a multimodal pretrained knowledge of indoor scene data for supervised object retrieval in real-world robot navigation.

3 Multimodal pretrained knowledge for robotic object navigation

Fig. 1 presents the comprehensive framework of multimodal pretrained knowledge for object navigation in real-world applications. Detailed descriptions of each component are provided below.

3.1 Multimodal pretrained knowledge

Inspired by Huang et al.[17], we develop multimodal pretrained knowledge to improve cross-modal alignment for robot navigation. We leveraged the Open Image V7 database to compile a set of indoor environmental concept words and their corresponding image regions. This knowledge consists of paired semantic concepts $\{t_i, v_i\}$, where t_i represents the Chinese description of the i-th semantic concept, and $v_i \in \mathbf{R}^D$ denotes its corresponding prototypical region representation. The total number of semantic concepts is denoted by K. This approach allows us to create pretrained knowledge encompassing textual and visual prototype representations.

For each Chinese text category t_i, the visual prototype representation v_i is obtained by averaging all associated region representations r_i, which is calculated as follows:

$$v_i = \frac{1}{J_i} \sum_{i=1}^{J_i} r_i. \tag{1}$$

Each region representation r_i is generated by inputting the bounding box and the image into the pretrained object detection model, Faster-RCNN[43].

3.2 Key-points screening and strategic optimization

As shown in Fig. 2(a), we randomly generate several waypoints with varying densities on the 2D grid map preconstructed via the Gmapping algorithm. Waypoints in non-navigable areas (gray regions) are then removed, and the remaining waypoints are designated as key-points. However, in the current navigation environment, challenges typically arise in scenarios such as intersections, entrances, and visually complex areas, which can lead to deviations from the planned trajectory. The randomly generated waypoints may not necessarily capture these critical locations.

To ensure that the generated waypoints are effective at key locations, as illustrated in Fig. 2(b), we strategically optimize them as follows: 1) Dense waypoints, which

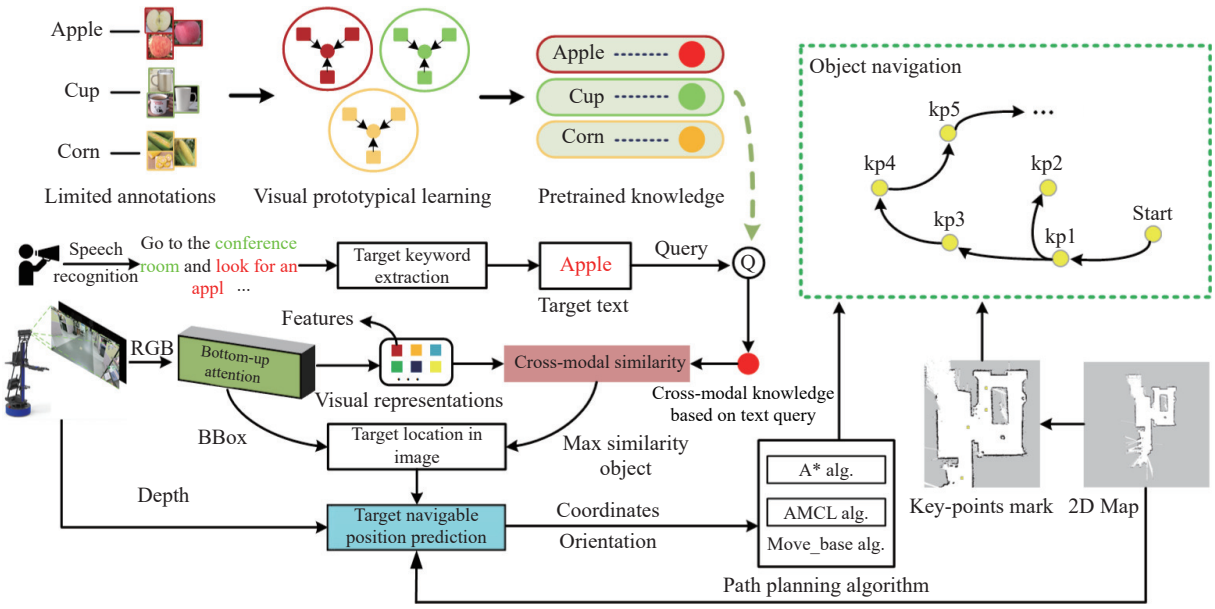


Fig. 1 An overview of the real-world object navigation method based on multimodal pretrained knowledge includes the following five main components: construction of multimodal pretrained knowledge, extraction of object keywords using natural language toolkit (NLTK), cross-modal alignment of vision and language for navigation supervision, prediction of optimal position and orientation, and map-based path planning for object navigation. (Colored figures are available in the online version at https://link.springer.com/journal/11633)

could negatively impact navigation efficiency, are removed; 2) waypoints near intersections, entrances, or visually complex areas are adjusted to align with these critical positions, where cross-modal alignment can better guide the navigation process. The remaining optimized waypoints are referred to as key-points. Each key-point serves as a reference for navigation, specifying the robot's position and orientation in the world coordinate system at that location.

3.3 Vision-language cross-modal matching for navigation supervision

For the RGB images captured by the Kinect v2.0 camera, we apply a bottom-up attention mechanism to extract 100 region features $V = \{v_1, \cdots, v_k\}, v_i \in \mathbf{R}^D$ from each image I via the bottom-up attention mechanism[43] from each image I, which covers most of the semantic content in the image. Each feature v_i corresponds to a specific region or salient object within the image, along with its associated bounding box information.

For spoken commands in Chinese, we use the iFlytek speech recognition system to convert speech to text. The target text is then extracted via NLTK's keyword extraction technique and mapped to its corresponding feature representation t via multimodal pretrained prototype knowledge. During matching, the similarity between the target text representation t and each feature v_i in the regional feature set $V = \{v_1, \cdots, v_k\}, v_i \in \mathbf{R}^D$, is evaluated via the cosine similarity function.

$$S_i(t, v_i) = t \times v_i \tag{2}$$

$$S_{\max} = \max(S_i), i \in [1, 2, \cdots, 100]. \tag{3}$$

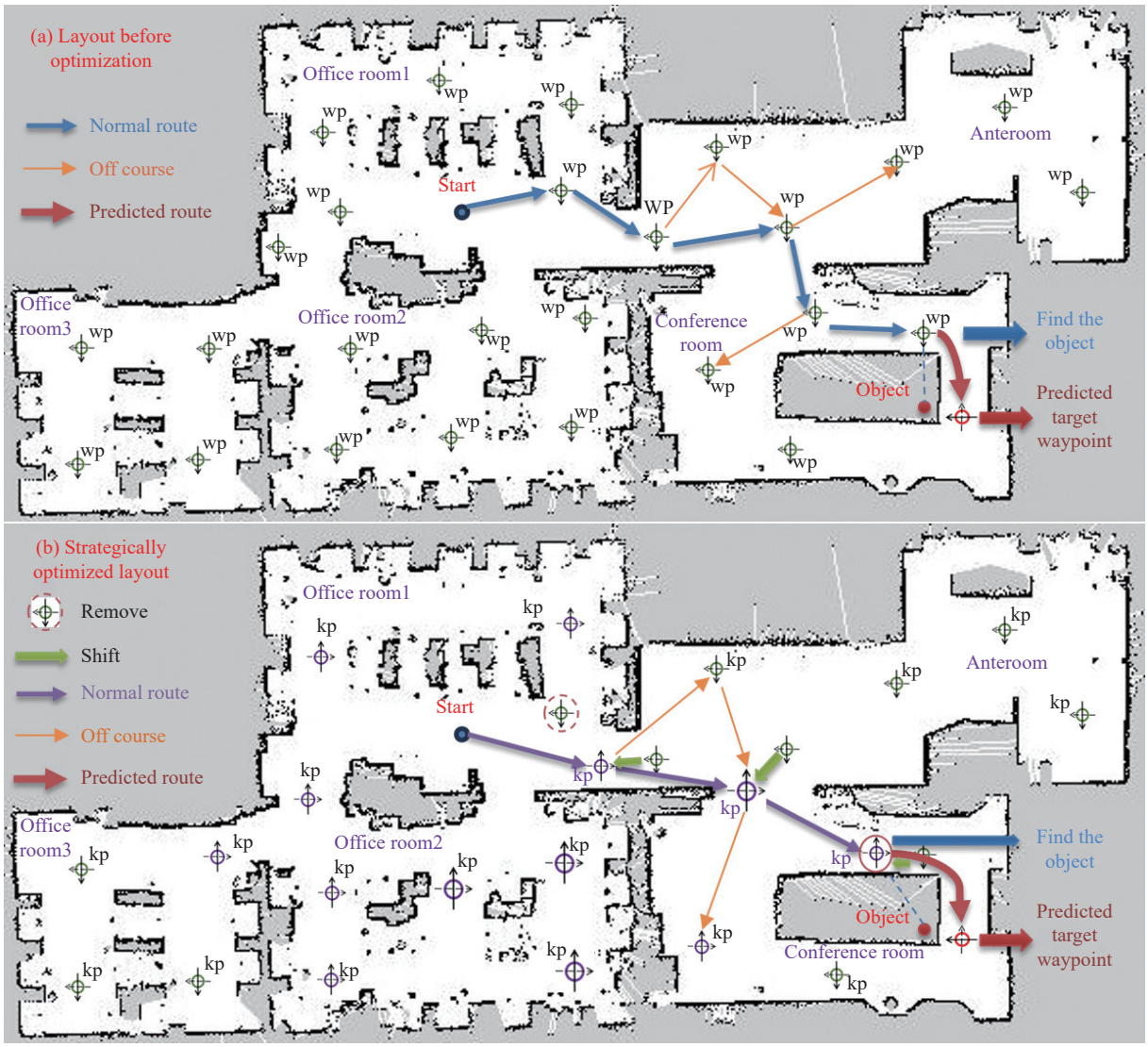


Fig. 2 Key-point screening and optimization layout diagram (Colored figures are available in the online version at https://link.springer.com/journal/11633)

During navigation, the robot performs cross-modal matching at each key-point to monitor the process. It calculates the maximum similarity score S_o between the target text and the image region, and then compares it to a preset threshold δ. If it is below the threshold, the target is deemed unsuccessfully matched at the key-point, prompting the robot to move to the next key-point with the closest Euclidean distance. Conversely, if S_o meets or exceeds δ, the target is successfully matched. The navigation system then predicts the optimal position and orientation close to the target object, guiding the robot towards it.

$$S_M = \begin{cases} 0, & \text{if } S_{\max} < \delta \\ 1, & \text{if } S_{\max} \geq \delta \end{cases} \tag{4}$$

where S_M denotes a successful match, and $S_{\max}$ represents the maximum similarity score achieved at key-point k. The set of all matching points is defined as $A = \{j \mid j = 1, 2, 3, \cdots\}$.

3.4 Target navigable position prediction

Once the target object has been successfully matched on the basis of the maximum similarity score, the robot must determine the object's precise location on the map. This process requires the use of depth information to aid in path planning, enabling the robot to approach the target efficiently.

The first step in the localization process is to confirm the pixel coordinates of the detected target object. Then, the RGB map must be aligned with the depth map. This alignment allows the extraction of the target object's coordinates from the depth map through coordinate mapping, obtaining the depth pixel values $\{pixel_i\}$ for the pixel segment encompassing the target object, where T_a represents the collection of all the pixels in the region and $pixel_i$ refers to the value of the i-th pixel. A filtering operation is performed to ensure data accuracy to exclude pixels with depths between 500 and 4 500. Finally, the depth distance between the Kinect v2.0 camera and the target object is calculated via the following formula:

$$depth = M_0\left(pixel_i\right), i \in T_\alpha. \tag{5}$$

As illustrated in Fig. 3, the depth distance $depth$ is used to calculate the object's coordinates in the world coordinate system.

$$X_c = (\mu - C_x) \times (depth/f_x) \tag{6}$$

$$Y_c = (\nu - C_y) \times (depth/f_y) \tag{7}$$

$$Z_c = depth \tag{8}$$

where u and v represent the pixel coordinates, while X_c, Y_c, and Z_c are the coordinates of the object in the camera coordinate system. The parameters f_x and f_y are the ratios of the camera's focal length f to the actual physical dimensions of the pixel d_x and d_y, respectively. C_x and C_y represent the differences between the pixel coordinates of the image center and the pixel coordinates of the point of interest in the horizontal and vertical directions, respectively.

Using the robot's mileage data, we can convert the target object's coordinates into the map coordinate system by applying the following coordinate transformation formula:

$$X_s = OD + DF = X_c \times \cos\theta + Y_c \times \sin\theta \tag{9}$$

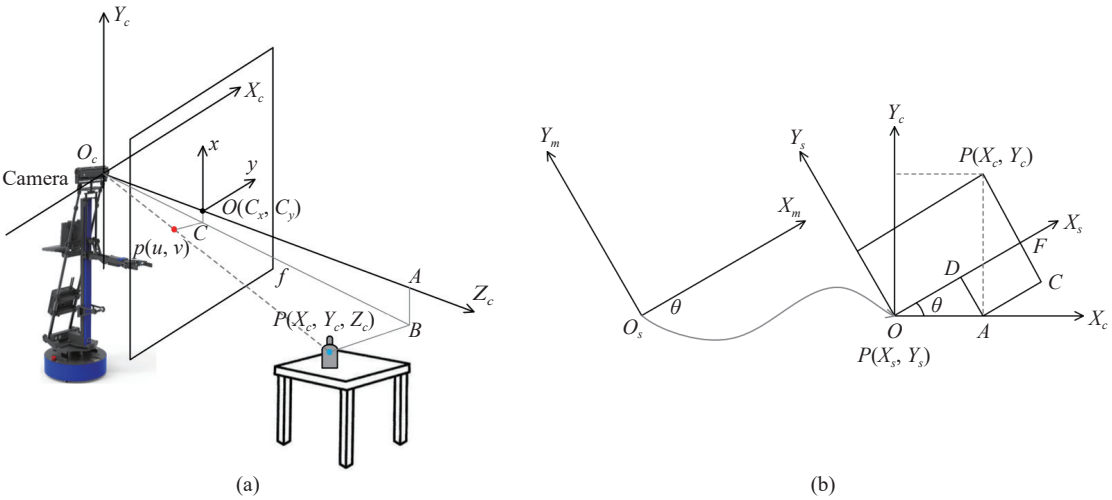


Fig. 3 Schematic diagram of the visual localization algorithm: (a) Three-dimensional coordinate calculation diagram; (b) Coordinate conversion diagram. (Colored figures are available in the online version at https://link.springer.com/journal/11633)

$$Y_s = PC - FC = Y_c \times \cos\theta - X_c \times \sin\theta \tag{10}$$

$$X_m = X_r + X_s,\ Y_m = Y_r + Y_s \tag{11}$$

where X_m and Y_m represent the world coordinates of the target object, while X_r and Y_r denote the current world coordinates recorded by the robot's odometer. In contrast, X_s and Y_s indicate the displacement in the world coordinate system between the robot's current position and the target object's location.

However, target objects such as beverages and fruits are often placed on tabletops or inside refrigerators, which are not part of the navigable area on a map. Even when the exact coordinates of these objects are obtained through localization methods, the robot still faces the challenge of being unable to reach these locations directly. To address this issue, we propose a strategy that integrates the locations of target objects with SLAM maps. The goal of this strategy is to identify nearby waypoints within the navigable region, enabling the robot to approach the target object. As illustrated in Fig. 4, the parameters of the neighboring navigable region are defined as follows: A large circle R with a 1-meter radius represents the operating range of the robot arm, whereas a smaller circle r with a 0.2-meter radius represents the space required by the robot chassis.

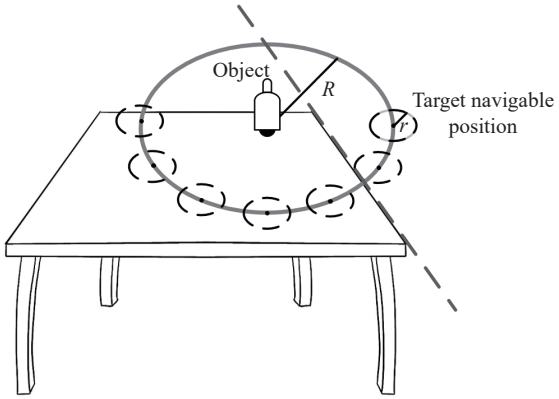


Fig. 4 Schematic diagram of the target navigable position prediction

A circle of radius R is drawn with the target object's coordinates as the center. For the robot to reach a waypoint, the center of that waypoint must lie on the boundary of this circle. Reachable waypoints are selected every 30 degrees along this boundary. Additionally, a circle of radius r is defined for each reachable waypoint. If no obstacles are found within the boundary of this circle, the waypoint is considered valid. Finally, to reach the target object from the navigation point, the position of the robot at the navigation point must be calculated using the following formula:

$$Y_{aw} = \cos^{-1}\left(\frac{\boldsymbol{start} \times \boldsymbol{end}}{|\boldsymbol{start}| \times |\boldsymbol{end}|}\right) = \beta \tag{12}$$

$$\begin{cases} q_i = \sin\left(\frac{\gamma}{2}\right) \times \cos\left(\frac{\beta}{2}\right) \times \cos\left(\frac{\alpha}{2}\right) - \\ \qquad \cos\left(\frac{\gamma}{2}\right) \times \sin\left(\frac{\beta}{2}\right) \times \sin\left(\frac{\alpha}{2}\right) \\ q_j = \cos\left(\frac{\gamma}{2}\right) \times \sin\left(\frac{\beta}{2}\right) \times \cos\left(\frac{\alpha}{2}\right) + \\ \qquad \sin\left(\frac{\gamma}{2}\right) \times \cos\left(\frac{\beta}{2}\right) \times \sin\left(\frac{\alpha}{2}\right) \\ q_k = \cos\left(\frac{\gamma}{2}\right) \times \cos\left(\frac{\beta}{2}\right) \times \sin\left(\frac{\alpha}{2}\right) - \\ \qquad \sin\left(\frac{\gamma}{2}\right) \times \sin\left(\frac{\beta}{2}\right) \times \cos\left(\frac{\alpha}{2}\right) \\ q_r = \cos\left(\frac{\gamma}{2}\right) \times \cos\left(\frac{\beta}{2}\right) \times \cos\left(\frac{\alpha}{2}\right) + \\ \qquad \sin\left(\frac{\gamma}{2}\right) \times \sin\left(\frac{\beta}{2}\right) \times \sin\left(\frac{\alpha}{2}\right) \end{cases} \tag{13}$$

where Y_{aw} represents the plane's angle of rotation when $\alpha = \beta = 0$. The term $[q_i \quad q_j;\ q_k \quad q_r]^{\mathrm{T}}$ represents the projection from four-dimensional space to three-dimensional space. The vector conversion formula is given by $p' = qpq^{-1}$, where p' is the transformed vector, q is the quaternion, p is the original vector, and q^{-1} is the inverse of the quaternion.

3.5 Gmapping diagram construction and move_base path planning

Gmapping for map construction: In our experimental setup, we utilize the Gmapping method[44] for environmental mapping. This method primarily employs a particle filtering algorithm to convert data from a 2D LiDAR sensor into a 2D raster map. Compared with Hector SLAM[45], Gmapping demonstrates superior localization accuracy and robustness. Additionally, in small-scale environments, Gmapping requires significantly fewer computational resources than cartographer does while maintaining comparable map accuracy.

Move_base path planning: In this study, we employ the move_base package of the ROS framework for path planning between key-points. The global path planner computes the optimal trajectory, whereas the local path planner refines the navigation routes to achieve optimal path planning. We integrate particle filters to track the robot's orientation in real-time on a 2D raster map, ensuring precise positioning.

Fig. 5 illustrates the integrated navigation framework developed on the ROS platform. In this framework, the robot constructs global and local cost maps by processing sensor data, particularly from LiDAR sensors. Using pre-existing SLAM maps, along with position and

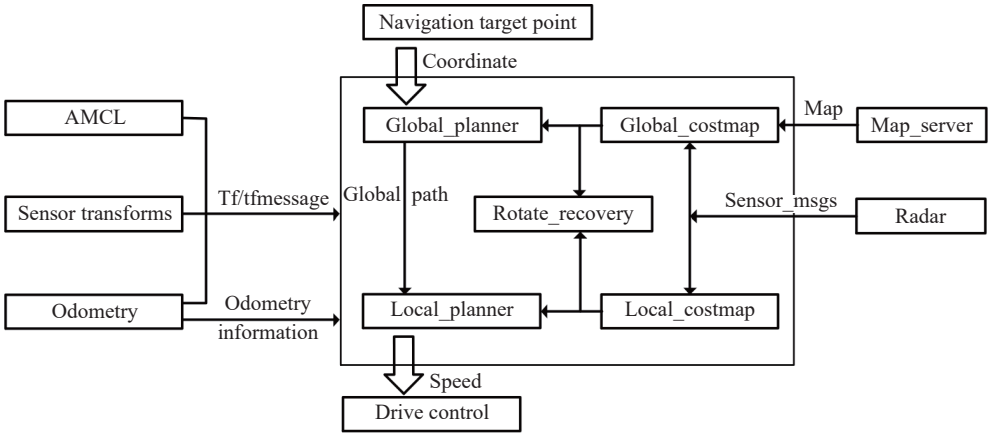


Fig. 5 Move_base path planning framework based on the ROS

mileage data provided by the adaptive monte carlo localization (AMCL) package, the system employs two planners – a global planner and a local planner – from the move_base function package. These planners collaborate to execute path planning on the basis of the global and local cost maps.

4 Robot navigation platform

Fig. 6 illustrates the physical architecture of the robot navigation platform we developed, which comprises seven key components, each serving a specific function: ① The Kinect v2.0 RGB-D camera captures both visual and depth data; ② a support bar ensures that the camera is positioned at the optimal height for data acquisition; ③ a lifting platform elevates the robotic arm as needed; ④ Lenovo 9 000P PCs with RTX4 090 graphics cards handle data processing, storage, and model deployment; ⑤ a gripper with two degrees of freedom enables flexible operation and interaction; ⑥ a dedicated stand safely and securely stores the robotic arm; and ⑦ the mobile chassis provides primary mobility for the robot. Together, these components form the foundation of the navigation platform.

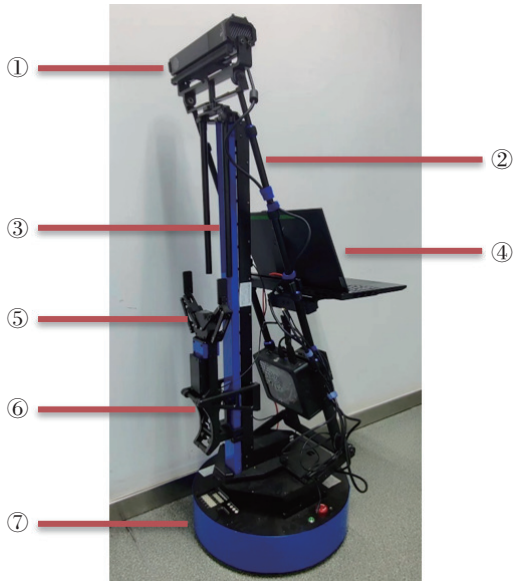


Fig. 6 Robot physical platform

4.1 Robot motion control model

For various application scenarios, researchers have developed a range of kinematic models to meet the specific motion requirements of robots. Real-world mobile robot navigation relies on a continuous action space, typically driven by continuous stochastic control systems to manage chassis motion. Compared with the discrete action spaces used in simulated environments, this approach is better suited for practical applications. However, navigation control in simulators, which is based on reinforcement learning[4] or discrete control systems[46–49] within discrete environments, generally offers greater stability. As illustrated in Fig. 7, an omnidirectional wheeled mobile robot is an effective solution for indoor environments. Its omnidirectional chassis design features three wheels, that are evenly distributed at 120° angles. The chassis supports two fundamental movements: linear motion (v) and rotational motion around the geometric center (ω).

Fig. 7 presents a coordinate system centered on the chassis, which enables the discretization of linear motion v in any direction into two components: v_x and v_y. Here, θ represents the angle between the velocity vector v and the v_x component along the x-axis, α is the angle between the axis of the third wheel and v_x, and r denotes the distance from each wheel to the center of the chassis. This approach simplifies complex linear and rotational motion into three independent wheel movements. For clarity, the motion decomposition of the first wheel is illustrated in this paper.

$$v_1 = -v_x \times \sin\alpha - v_y \times \cos\alpha + \omega \times r. \tag{14}$$

Similarly, the motion of wheel 2 and wheel 3 can be decomposed into:

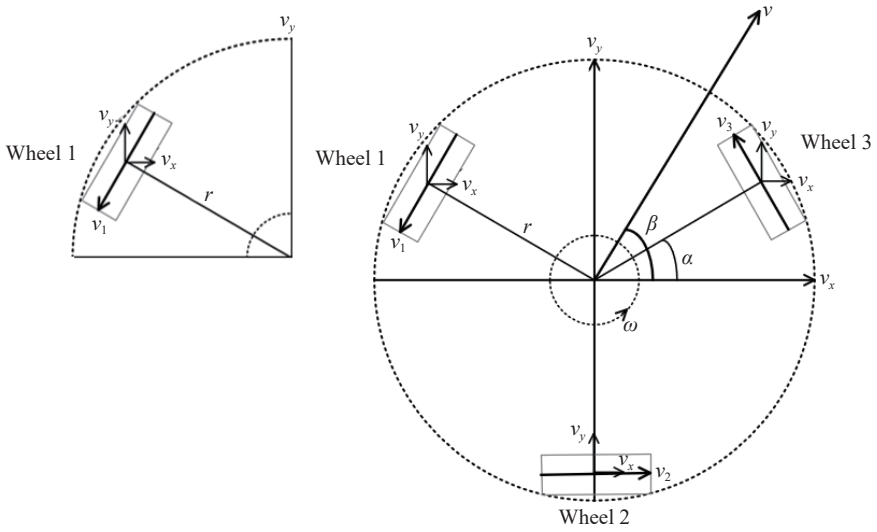


Fig. 7 Kinematic modeling diagram of the omnidirectional wheel chassis

$$v_2 = v_x + \omega \times r \tag{15}$$

$$v_3 = -v_x \times \sin\alpha + v_y \times \cos\alpha + \omega \times r \tag{16}$$

where $\alpha = 30°$ is set and on the basis of the velocity messages (including linear and angular velocities) sent by the ROS core node, the inverse kinematics model for the three wheels of the chassis is derived as follows:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} & r \\ 1 & 0 & r \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & r \end{bmatrix} \times \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix}. \tag{17}$$

The forward kinematics model is as follows:

$$\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = \begin{bmatrix} -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} \\ -\frac{\sqrt{3}}{3} & 0 & \frac{\sqrt{3}}{3} \\ \frac{1}{3r} & \frac{1}{3r} & \frac{1}{3r} \end{bmatrix} \times \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}. \tag{18}$$

To this end, by setting motion state commands for the omnidirectional chassis, we can regulate the robot's wheel speed via (17), which is the inverse kinematics model. Additionally, the forward kinematics model of (18) is used to calculate the data during the robot's motion.

4.2 Environment perceptron system

The Kinect v2.0 camera mounted on the robot platform captures both RGB color images and depth information of corresponding pixels. The technical specifications of the Kinect v2.0 camera are provided in Table 1. Additionally, the platform incorporates a 2D LiDAR sensor for omnidirectional environmental sensing. These sensors are essential for map construction and for achieving precise positioning and navigation via simultaneous localization and mapping (SLAM) technology. As shown in Fig. 1, the LiDAR is mounted on a mobile chassis with a 360° scanning range, a detection distance of up to 12 m, and a ranging accuracy of 0.3 cm, whereas the scanning accuracy is 0.9°.

Table 1 Kinect v2.0 camera parameters

Parameter name	Parameter value
Color camera resolution	1 920 × 1 080
Color camera FPS	30 fps
Depth camera resolution	512 × 424
Depth camera FPS	30 fps
The angle of view (FOV)	70° × 60°
Perception range	0.5–4.5 m

5 Experimental analysis

5.1 Experimental setup and implementation details

All the experiments in this paper were conducted in a realistic robotic environment in which a robot operating system (ROS)[50] was used to manage and deploy various modules. Computationally intensive tasks, such as bottom-up attention detection modeling, were performed on a Lenovo 9 000P computer equipped with an RTX 4 090 GPU. ROS mechanisms, including topics, communication, and services, were used to integrate all system components and enable centralized node management.

During the experiments, the robot received voice commands such as "find a coke in the conference room". These commands were converted to text via the iFlytek speech service, and then forwarded by the ROS server for execution. The robot navigated from the starting point to the designated meeting room on the basis of an existing map. Navigation was considered successful if the robot approached the target within 1 meter, with no obstacles along the path. The Rviz tool recorded each navigation path for future reference. Using Python scripts, we extracted image IDs and category information from the Open Images V7 dataset, parsing the bounding box annotations in the CSV to build the multimodal pretrained knowledge. Ultimately, the multimodal pretrained knowledge and regional feature representations reached 2 048 dimensions.

5.2 Datasets and protocols

Our data sources are divided into two main categories: the Open Images V7 open-source dataset and real-world scene data collected from the laboratory. The former is used to build a multimodal prototype knowledge, whereas the latter is employed for testing object matching in real environments. Open Images V7 contains over 9 million images with category annotations, approximately 1.9 million of which have precise annotations with clear boundaries, making them highly suitable for multimodal knowledge construction. These images cover 601 categories, from which we selected 100 categories of indoor objects relevant to robot navigation, including office supplies, conference room items, beverages, and fruits.

The multimodal pretrained knowledge studied in this work primarily consists of dataset-independent semantic concepts, aiming for general applicability across different scenarios. Table 2 presents the statistical properties of MACK (indoor), including images, regions, and words. In MACK (indoor), the mean number of regions per word is 546, with the maximum number of regions per word reaching 2 479. This is a significant advantage, as it allows the knowledge base to cover a wide range of semantic content.

The laboratory data, which were primarily used for testing, were periodically collected by a manually operated robot. To ensure the effective application of multimodal knowledge in real-world environments, we intentionally gathered data from diverse settings, including elevators, office areas, conference rooms, and break rooms. A Kinect v2.0 camera was used to capture 30 images of each object category from multiple angles, resulting in a total of 3 000 test images across 100 categories, each at a resolution of 1 920 × 1 080. Fig. 8 illustrates the selected scenes and corresponding image data.

To evaluate the effectiveness of multimodal pretrained knowledge in associating image regions with textual descriptions, we employ the metrics "R@1", "R@3", and "R@5", which correspond to the recall rates of the top 1, 3, and 5 results, respectively. Additionally, we record the average maximum similarity score, denoted as $A@S_{\max}$, and set the threshold δ accordingly.

To assess navigation performance, we introduce metrics such as average path length (APL), success rate (SR), success weighted by path length (SPL)[51], and average navigation time. Importantly, path length and navigation time data are derived only from successful navigation attempts to ensure the accuracy of the averages. The recorded navigation time includes the robot's movement, human-robot voice interactions, critical data collection, data loading, and algorithmic reasoning.

5.3 Experiment on associating the most similar object region with multimodal pretrained knowledge

To evaluate the effectiveness of our constructed multimodal pretrained knowledge in retrieving target objects, we assessed its cross-modal retrieval efficiency for common objects in laboratory environments of varying sizes. Table 3 presents the average maximum similarity scores and retrieval accuracies. The average score for each category exceeds 0.7, demonstrating that the pretrained knowledge can effectively adapt to new environments and accurately localize targets. In the navigation experiments, the object categories are classified on the basis of their respective thresholds, δ, which is set as the average maximum similarity score minus 0.1. The results show that most object categories can be successfully retrieved within the first five images, with a retrieval recall rate exceeding 95% in the first three images. However, smaller or more distant objects, such as mice, calculators, and staplers, have a retrieval recall rate of less than 90% in the first image. This decrease in recognition accuracy is likely due to the small size and distance of these objects in complex scenes, which affects the localization effectiveness of the pretrained knowledge. Accurate recall of high-similarity image regions retrieved in previous iterations is crucial for navigation experiments. To improve this process, we developed a simple yet effective strategy: First, crop the central portion of the original image and apply a quadrature; next, crop $\frac{1}{4}$ of the region from the top-left to the bottom-right; and finally, match the original im-

Table 2 Statistical properties of multimodal pretrained knowledge (indoor) for indoor environment

Data source	The number of words	The number of selected words	The number of selected regions	Visual knowledge	The mean number of regions per word	The max number of regions per word	The min number of regions per word
Open Images V7	601	100	54.6G	Region-level	546	2 479	85

Fig. 8 The image data of some scenes (Colored figures are available in the online version at https://link.springer.com/journal/11633)

age six times, including the five cropped regions, to identify the most similar target objects.

As shown in Table 4, the optimized search strategy significantly improves two key metrics: $A@S_{\max}$ and $R@1$. The $A@S_{\max}$ metric increases to over 0.85, while the recall rate for $R@1$ surpasses 90%, and $R@3$ achieves a perfect 100% recall. These results indicate that the optimization strategy improves the accuracy of localizing small target objects while preserving image information.

5.4 Object navigation with varying key-point densities

To evaluate the impact of different key-point density ranges on the efficiency of our proposed navigation method, we defined five distinct density ranges and comprehensively assessed several key metrics for real robot navigation in the task of searching for a "computer keyboard". These metrics include the navigation success rate (SR), average navigation time, average path length (APL), and success rate adjusted for the path length penalty (SPL). For each density range, we conducted 30 navigation experiments, and the results before and after key-point optimization are presented in Table 5. We observed that different key-point density ranges resulted in varying levels of navigation performance in the same environment.

Specifically, all the performance metrics improved after the strategic optimization of key-points. When navigating across one or two rooms, the optimized density range of 3.5–4.5 m/point achieved the highest navigation success rate of 66.7%, with an SPL of 60.5%, highlighting the importance of accurate key-point localization during multi-room navigation. Furthermore, the data show that both the optimized APL and average navigation time are lower than those optimizations before, suggesting that the optimized navigation process is more efficient and robust. In contrast, when the key-points were too densely packed (2.5–3.5 m/point), the navigation performance decreased. This is likely because a high density of key-points requires additional exploration time, reducing overall efficiency and limiting the flexibility of path planning, which negatively impacts overall navigation performance.

Table 3 Comparison of the matching performance of multimodal pretrained knowledge associations for different target objects

Test category	$A@S_{max}$	$R@1$	$R@3$	$R@5$	δ
Small objects					
Mouse	0.79	88	98	100	0.69
Mobile phone	0.75	90	98	100	0.65
Calculator	0.75	87	95	100	0.65
Stapler	0.76	88	97	100	0.66
Fruit objects					
Apple	0.82	90	99	100	0.72
Banana	0.80	89	98	100	0.70
Watermelon	0.82	92	100	100	0.72
Pineapple	0.79	93	99	100	0.69
Large objects					
Flowerpot	0.72	91	97	100	0.62
Computer keyboard	0.74	95	100	100	0.64
Laptop	0.77	95	99	100	0.67
Computer monitor	0.80	96	100	100	0.70

Table 4 Performance comparison of the two solutions of "cropping first, matching later" and direct matching of the original image for searching for small target objects

Test category	$A@S_{max}$	$R@1$	$R@3$	$R@5$	δ
Mouse	0.79	88	98	**100**	0.69
Mouse-crop	**0.86**	**96**	**100**	**100**	**0.76**
Calculator	0.75	87	95	**100**	0.65
Calculator-crop	**0.85**	**94**	**100**	**100**	**0.75**
Stapler	0.76	88	97	**100**	0.66
Stapler-crop	**0.85**	**93**	**100**	**100**	**0.75**
Apple	0.82	90	99	**100**	0.72
Apple-crop	**0.88**	**95**	**100**	**100**	**0.78**
Banana	0.82	92	**100**	**100**	0.72
Banana-crop	**0.87**	**96**	**100**	**100**	**0.77**

5.5 Performance comparison with the advanced method

We compared several key navigation performance metrics between the proposed method and other advanced methods[16, 36] in a real-world environment. The detailed results are presented in Table 6. Although the experimental equipment differed slightly, with Sim-to-Real employing a Theta V 360° panoramic camera with a wider field of view and higher resolution, whereas we used a Kinect v2 camera. Both experiments were conducted in offices or laboratories at different locations. The results show that, despite the narrower field of view of our sensors, our method significantly outperforms the Sim-to-Real method in terms of the success rate (SR) and success rate as affected by path length (SPL), with improvements of 19.9% and 16.6%, respectively. Additionally, while both methods require supervised model training for real-world navigation, our method directly associates images and text without the need for model training.

Table 5 Comparison of navigation performance across different key-point density ranges for the same navigation target

Key-point density	Optimized	8–10 (m/point)	6–8 (m/point)	4.5–6 (m/point)	3.5–4.5 (m/point)	2.5–3.5 (m/point)
Episodes	–	30	30	30	30	30
Successes	×	10	12	14	18	15
SR (%)	×	33.3	40.0	46.7	60.0	50.0
APL (m)	×	19.0	19.8	20.6	21.2	21.8
SPL (%)	×	30.7	35.4	39.7	49.5	40.1
Average time (s)	×	27.4	31.5	34.2	38.9	43.5
Successes	√	11	13	16	**20**	18
SR (%)	√	36.7	43.3	53.3	**66.7**	60.0
APL (m)	√	18.2	18.5	19.0	**19.3**	20.2
SPL (%)	√	35.3	41.0	49.1	**60.5**	52.0
Average time (s)	√	26.0	29.2	32.5	**37.1**	41.3

Table 6 Comparison of performance in real scenarios with that of the advanced method

Methods	Camera	Env	APL (m)	SR (%)	SPL (%)
ROBOTHOR[36]	Intel RealSense	Sim-2-Real (home)	30.16	50.00	28.50
Sim-to-Real[16]	Theta V	Coda	11.32	46.80	43.90
Knowledge (ours)	Kinect v2	Laboratory	19.30	**66.70**	**60.50**

5.6 Influence of different objects on navigation efficiency

In our experiments, we tested a range of objects and observed significant variations in navigation performance, as shown in Table 7. This study investigated how object characteristics affect navigation efficiency. Notably, the ability of the Kinect v2 camera to perceive and localize the environment was affected by these characteristics. Specifically, transparent objects fail to generate accurate depth maps because of laser penetration issues. Additionally, variations in depth recordings at different distances lead to inaccurate object localization, which in turn impacts the robot′s path planning, are reflected in the SR and SPL.

Table 7 Navigation efficiency of different objects

Test category	*Episodes*	*Success*	SR (%)	APL (m)	SPL (%)
Apple	30	16	53.3	19.1	48.8
Banana	30	15	50.0	19.8	44.2
Stapler	30	14	46.7	19.4	42.1
Calculator	30	15	50.0	19.4	45.1
Mobile phone	30	18	60.0	20.2	52.0
Mouse	30	16	53.3	19.5	47.8
Watermelon	30	17	56.7	19.3	51.4
Coke	30	18	60.0	19.5	53.8
Flowerpot	30	18	60.0	19.6	53.6
Computer keyboard	30	**20**	**66.7**	19.3	**60.5**

For example, in Scene 4 (Fig. 8), the system incorrectly localized a "colored plastic box" at the back edge of a table, causing the robot to choose a longer path to reach the target. This highlights the complexity of depth localization in navigation and its significant impact on planning, which must be addressed in practical applications.

We also found that larger objects are easier to retrieve and yield better navigation performance. For smaller objects at greater distances, the matching similarity score tends to be lower, which negatively affects the success rate of object navigation. However, performance improves with the "cropping first, matching later" strategy. The experimental results are shown in Table 8.

5.7 Robustness analysis of navigation in dynamic and visually complex environments

To explore the challenges of achieving robust cross-modal alignment for robots in dynamic or visually complex environments, we conducted navigation experiments in environments with varying lighting conditions and structural layouts. Specifically, we examined three different lighting conditions – daytime, nighttime (no lights), and nighttime (with lights) – to analyze the robustness of cross-modal navigation for the object "smartphone". The experimental results are shown in Table 9. We observed that the navigation performance under both daytime and nighttime conditions (with lights) was similar, with slightly better performance during nighttime with lights. However, under dark conditions, both object retrieval performance and robot navigation performance decreased significantly, demonstrating a clear lack of robustness. This indicates that the visual perception of the robot and the retrieval of cross-modal objects are heavily based on adequate lighting or optimal illumination conditions.

Table 8 Navigation performance comparison of the two solutions of "cropping first, matching later" and direct matching of the original image for searching small target objects

Test category	*Episodes*	$A@S_{\max}$	SR (%)	APL (m)	SPL (%)
Mouse	30	0.79	53.30	19.50	41.90
Mouse-crop	30	**0.86**	**60.00**	19.40	**54.10**
Calculator	30	0.75	50.00	19.40	45.10
Calculator-crop	30	**0.85**	**53.30**	19.60	**47.60**
Stapler	30	0.76	46.70	19.40	42.10
Stapler-crop	30	**0.85**	**56.70**	19.40	**51.10**
Apple	30	0.82	53.30	19.10	48.80
Apple-crop	30	**0.88**	**60.00**	19.20	**54.70**
Banana	30	0.82	50.00	19.80	44.20
Banana-crop	30	**0.87**	**53.30**	19.70	**47.30**

Table 9 Comparison of the object navigation performance of "mobile phone" under different lighting conditions and environment layouts

Setting	*Episodes*	$A@S_{\max}$	$R@1$	SR (%)	SPL (%)
Lighting conditions					
Daytime	20	0.73	90.00	60.20	52.30
Evening	20	0.56	75.00	48.50	40.70
Nighttime	20	0.78	90.00	61.20	53.60
Environment layouts (daytime)					
Single scene	20	0.73	90.00	60.20	52.30
Semantic occlusion	20	**0.62**	**70.00**	**48.70**	**42.90**
Similar interference	20	**0.71**	**75.00**	**54.60**	**48.20**
Complex background	20	**0.67**	**85.00**	**51.40**	**45.80**

Furthermore, we tested various environmental layouts to assess the robot′s ability to handle interference in complex and dynamic environments: 1) a single scene, where the background is simple and objects are easily distinguishable; 2) semantic occlusion, where objects are partially obstructed, making it difficult for the robot to

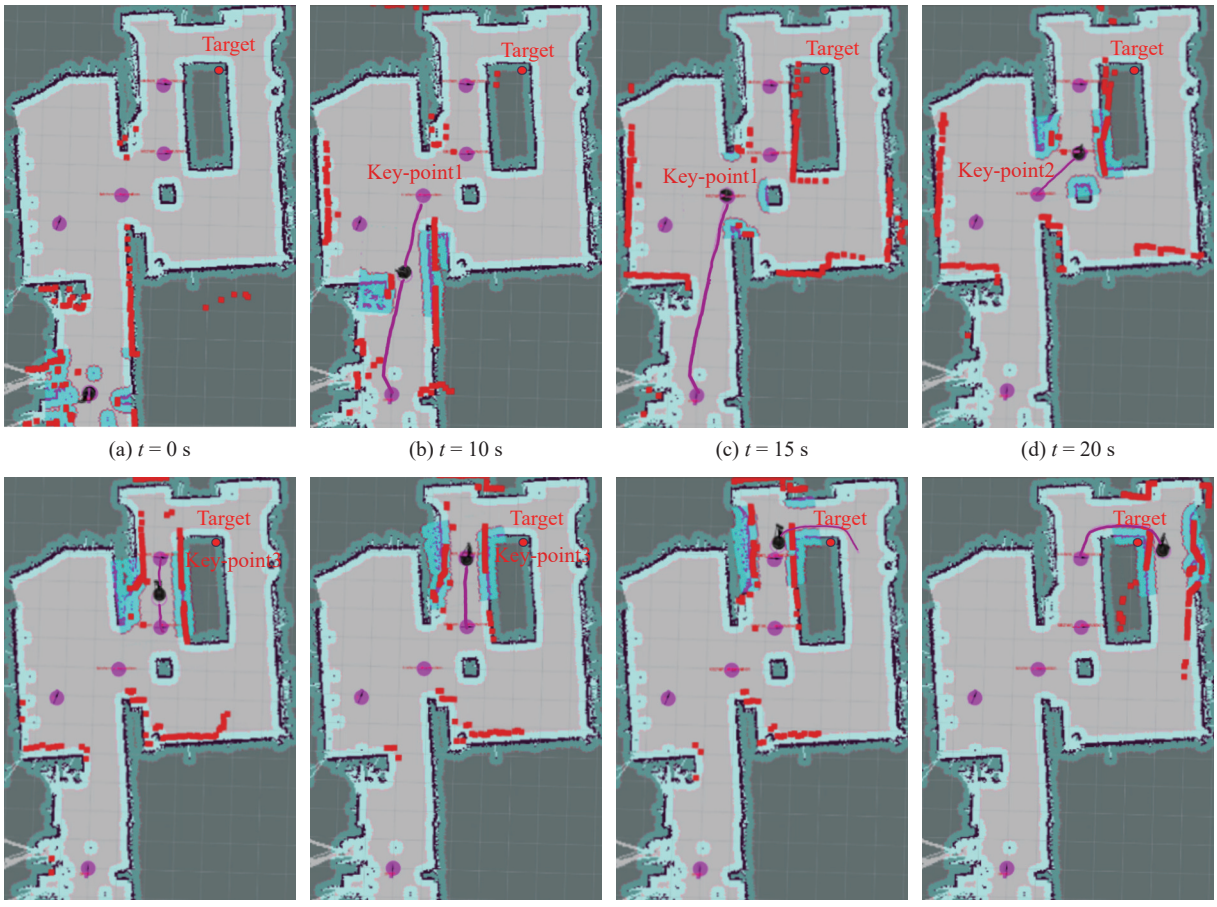
understand the semantic meaning of the scene; 3) similar interference, where other objects of similar class near the target object cause interference; and 4) complex background, where the background of the scene is intricate and varied, introducing semantic interference. From these results, we can conclude that the multimodal pretrained knowledge we developed has strong generalizability. In environments with semantic occlusion, interference from similar objects, and complex backgrounds, the average cross-modal alignment similarity $A@S_{\max}$ consistently exceeds 0.6. Moreover, the robot's navigation SR surpasses that of the Sim-to-Real[16] method. Notably, under semantic occlusion conditions, the SPL metric of our method is also comparable to that of the advanced Sim-to-Real method.

5.8 Visualization analysis of the navigation process

For a thorough analysis, we examined a path defined by three key points at various time intervals, as shown in Fig. 9. The robot moved at a speed of 0.5 meters per second, completing the task in approximately 40 seconds. During this time, it also performs tasks such as voice interaction, visual feature extraction, and cross-modal object search. Fig. 10 shows the navigation results in a real-world environment: The left image displays the initial view from the Kinect v2.0 camera mounted on the robot, whereas the right image shows the robot successfully docking with the target object (a bottle of COLA), with the camera consistently aligned. This analysis highlights the effectiveness and real-world applicability of our method.

6 Conclusions

In this paper, we propose a real-world object navigation method driven by multimodal pretrained knowledge, leveraging cross-modal alignment between vision and language at critical navigation points to supervise robot navigation. First, we collect word-region pairs for 100 indoor object categories via the Open Images V7 dataset and real-world laboratory data to create the indoor object knowledge MACK (indoor). Next, we optimize randomly generated waypoints on the basis of key navigation positions, selecting the optimized points as key-points. MACK (indoor) is then applied at these key-points for



(a) $t = 0$ s (b) $t = 10$ s (c) $t = 15$ s (d) $t = 20$ s

(e) $t = 25$ s (f) $t = 30$ s (g) $t = 35$ s (h) $t = 40$ s

Fig. 9 Navigation path visualization (Colored figures are available in the online version at https://link.springer.com/journal/11633)

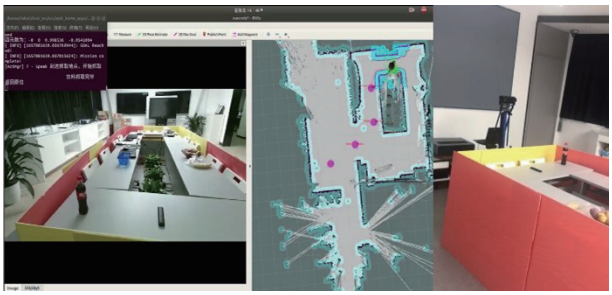


Fig. 10 Robot navigation demonstrated in a real-world environment (Colored figures are available in the online version at https://link.springer.com/journal/11633)

visual-language matching to supervise robot navigation. We also introduce a target position prediction strategy that accurately predicts the optimal position and orientation for the robot to approach the target object. Finally, our method is implemented on a physical robot, that passes visual-language object navigation tests and demonstrates superior performance over existing methods.

In future work, we plan to equip the robot with a panoramic camera and a high-precision 3D radar system, as well as extend existing vision-and-language navigation (VLN) models such as discrete-continuous-VLN[15], VLN-CE[23], Habitat[52], and ETPNav[53] to real-world environments to expand their applicability. Our ongoing efforts aim to enhance the robustness, reliability, and adaptability of visual-linguistic navigation, and bridge the gap between theory and practice, thereby maximizing its real-world impact.

Acknowledgements

This work was jointly supported by the National Natural Science Foundation of China (Nos. 62236010, 62322607, 62276261 and 62076014), the Youth Innovation Promotion Association of Chinese Academy of Sciences, China (No. 2021128), the Joint Fund of Natural Science of Hunan Province, China (No. 2023JJ50242), and the Key Projects of Education Department of Hunan Province, China (No. 22A0115).

Declarations of conflict of interest

The authors declared that they have no conflicts of interest to this work.

References

[1] X. Li, X. Wu, B. Zhu, X. Dai. A visual navigation method using a hand-drawn-route-map in dynamic environments. *Robot*, vol. 33, no. 4, pp. 490–501, 2011. DOI: 10.3724/SP.J.1218.2011.00490.

[2] F. Bonin-Font, A. Ortiz, G. Oliver. Visual navigation for mobile robots: A survey. *Journal of Intelligent and Robotic Systems*, vol. 53, no. 3, pp. 263–296, 2008. DOI: 10.1007/s10846-008-9235-4.

[3] D. M. Lyons, M. Rahouti. WAVN: Wide area visual navigation for large-scale, GPS-denied environments. In *Proceedings of IEEE International Conference on Robotics and Automation*, London, UK, pp. 2039–2045, 2023. DOI: 10.1109/ICRA48891.2023.10160511.

[4] L. Ma, Y. Liu, J. Chen. Using RGB image as visual input for mapless robot navigation, [Online], Available: https://arxiv.org/abs/1903.09927, 2019.

[5] Y. Wang, W. Liu, J. Liu, C. Sun. Cooperative USV–UAV marine search and rescue with visual navigation and reinforcement learning-based control. *ISA Transactions*, vol. 137, pp. 222–235, 2023. DOI: 10.1016/j.isatra.2023.01.007.

[6] S. Wang, Z. Wu, X. Hu, Y. Lin, K. Lv. Skill-based hierarchical reinforcement learning for target visual navigation. *IEEE Transactions on Multimedia*, vol. 25, pp. 8920–8932, 2023. DOI: 10.1109/TMM.2023.3243618.

[7] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, A. van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, USA, pp. 3674–3683, 2018. DOI: 10.1109/CVPR.2018.00387.

[8] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Nießner, M. Savva, S. Song, A. Zeng, Y. Zhang. Matterport3D: Learning from RGB-D data in indoor environments, [Online], Available: https://arxiv.org/abs/1709.06158, 2017.

[9] C. Y. Ma, J. Lu, Z. X. Wu, G. AlRegib, Z. Kira, R. Socher, C. Xiong. Self-monitoring navigation agent via auxiliary progress estimation, [Online], Available: https://arxiv.org/abs/1901.03035, 2019.

[10] D. An, Y. Qi, Y. Huang, Q. Wu, L. Wang, T. Tan. Neighbor-view enhanced model for vision and language navigation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 5101–5109, 2021. DOI: 10.1145/3474085.3475282.

[11] Y. Hong, Q. Wu, Y. Qi, C. Rodriguez-Opazo, S. Gould. VLN↻BERT: A recurrent vision-and-language Bert for navigation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, USA, pp. 1643–1653, 2020. DOI: 10.1109/CVPR46437.2021.00169.

[12] D. An, Y. Qi, Y. Li, Y. Huang, L. Wang, T. Tan, J. Shao. BEVBert: Multimodal map pre-training for language-guided navigation. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, Paris, France, pp. 2737–2748, 2023.

[13] K. He, Y. Huang, Q. Wu, J. Yang, D. An, S. Sima, L. Wang. Landmark-RxR: Solving vision-and-language navigation with fine-grained alignment supervision. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, Article number 50, 2021.

[14] Y. Qi, Q. Wu, P. Anderson, X. Wang, W. Y. Wang, C. Shen, A. van den Hengel. REVERIE: Remote embodied visual referring expression in real indoor environments. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, USA, pp. 9979–9988, 2020. DOI: 10.1109/CVPR42600.2020.01000.

[15] Y. Hong, Z. Wang, Q. Wu, S. Gould. Bridging the gap

between learning in discrete and continuous environments for vision-and-language navigation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, USA, pp. 15418–15428, 2022. DOI: 10.1109/CVPR52688.2022.01500.

[16] P. Anderson, A. Shrivastava, J. Truong, A. Majumdar, D. Parikh, D. Batra, S. Lee. Sim-to-real transfer for vision-and-language navigation. In *Proceedings of Conference on Robot Learning*, pp. 671–681, 2021.

[17] Y. Huang, Y. Wang, Y. Zeng, L. Wang. MACK: Multimodal aligned conceptual knowledge for unpaired image-text matching. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, New Orleans, USA, Article number 573, 2022.

[18] D. Fried, R. Hu, V. Cirik, A. Rohrbach, J. Andreas, L. P. Morency, T. Berg-Kirkpatrick, K. Saenko, D. Klein, T. Darrell. Speaker-follower models for vision-and-language navigation. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montreal, Canada, pp. 3318–3329, 2018.

[19] W. Hao, C. Li, X. Li, L. Carin, J. Gao. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, USA, pp. 13134–13143, 2020. DOI: 10.1109/CV-PR42600.2020.01315.

[20] A. Moudgil, A. Majumdar, H. Agrawal, S. Lee, D. Batra. SOAT: A scene- and object-aware transformer for vision-and-language navigation. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, Article number 763, 2021.

[21] A. Pashevich, C. Schmid, C. Sun. Episodic transformer for vision-and-language navigation. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, Montreal, Canada, pp. 15922–15932, 2021. DOI: 10.1109/ICCV48922.2021.01564.

[22] S. Chen, P. L. Guhur, C. Schmid, I. Laptev. History aware multimodal transformer for vision-and-language navigation. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, Article number 446, 2021.

[23] J. Krantz, E. Wijmans, A. Majumdar, D. Batra, S. Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Proceedings of the 16th European Conference on Computer Vision*, Glasgow, UK, pp. 104–120, 2020. DOI: 10.1007/978-3-030-58604-1_7.

[24] J. Krantz, S. Lee. Sim-2-sim transfer for vision-and-language navigation in continuous environments. In *Proceedings of the 17th European Conference on Computer Vision*, Tel Aviv, Israel, pp. 588–603, 2022. DOI: 10.1007/978-3-031-19842-7_34.

[25] X. Li, D. Guo, H. Liu, F. Sun. REVE-CE: Remote embodied visual referring expression in continuous environment. *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1494–1501, 2022. DOI: 10.1109/LRA.2022.3141150.

[26] S. Chen, P. L. Guhur, M. Tapaswi, C. Schmid, I. Laptev. Think global, act local: Dual-scale graph transformer for vision-and-language navigation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, USA, pp. 16516–16526, 2022. DOI: 10.1109/CVPR52688.2022.01604.

[27] Y. Qi, Z. Pan, S. Zhang, A. van den Hengel, Q. Wu. Object-and-action aware model for visual language navigation. In *Proceedings of the 16th European Conference on Computer Vision*, Glasgow, UK, pp. 303–317, 2020. DOI: 10.1007/978-3-030-58607-2_18.

[28] Y. Hong, C. Rodriguez-Opazo, Y. Qi, Q. Wu, S. Gould. Language and visual entity relationship graph for agent navigation. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Vancouver, Canada, Article number 644, 2020.

[29] H. Huang, V. Jain, H. Mehta, A. Ku, G. Magalhaes, J. Baldridge, E. Ie. Transferable representation learning in vision-and-language navigation. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, Seoul, Republic of Korea, pp. 7403–7412, 2019. DOI: 10.1109/ICCV.2019.00750.

[30] G. Ilharco, V. Jain, A. Ku, E. Ie, J. Baldridge. General evaluation for instruction conditioned navigation using dynamic time warping, [Online], Available: https://arxiv.org/abs/1907.05446, 2019.

[31] H. Wang, W. Wang, T. Shu, W. Liang, J. Shen. Active visual information gathering for vision-language navigation. In *Proceedings of the 16th European Conference on Computer Vision*, Glasgow, UK, pp. 307–322, 2020. DOI: 10.1007/978-3-030-58542-6_19.

[32] J. Y. Koh, H. Lee, Y. Yang, J. Baldridge, P. Anderson. Pathdreamer: A world model for indoor navigation. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, Montreal, Canada, pp. 14718–14728, 2021. DOI: 10.1109/ICCV48922.2021.01447.

[33] L. Ke, X. Li, Y. Bisk, A. Holtzman, Z. Gan, J. Liu, J. Gao, Y. Choi, S. Srinivasa. Tactical rewind: Self-correction via backtracking in vision-and-language navigation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, USA, pp. 6741–6749, 2019. DOI: 10.1109/CVPR.2019.00690.

[34] Y. Cui, L. Xie, Y. Zhang, M. Zhang, Y. Yan, E. Yin. Grounded entity-landmark adaptive pre-training for vision-and-language navigation. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, Paris, France, pp. 12009–12019, 2023. DOI: 10.1109/ICCV51070.2023.01106.

[35] D. Batra, A. Gokaslan, A. Kembhavi, O. Maksymets, R. Mottaghi, M. Savva, A. Toshev, E. Wijmans. ObjectNav revisited: On evaluation of embodied agents navigating to objects, [Online], Available: https://arxiv.org/abs/2006.13171, 2020.

[36] M. Deitke, W. Han, A. Herrasti, A. Kembhavi, E. Kolve, R. Mottaghi, J. Salvador, D. Schwenk, E. VanderBilt, M. Wallingford, L. Weihs, M. Yatskar, A. Farhadi. RoboTHOR: An open simulation-to-real embodied AI platform. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, USA, pp. 3161–3171, 2020. DOI: 10.1109/CVPR42600.2020.00323.

[37] W. Cai, S. Huang, G. Cheng, Y. Long, P. Gao, C. Sun, H. Dong. Bridging zero-shot object navigation and foundation models through pixel-guided navigation skill. In *Proceedings of IEEE International Conference on Robotics and Automation*, Yokohama, Japan, pp. 5228–5234, 2024. DOI: 10.1109/ICRA57147.2024.10610499.

[38] N. Yokoyama, S. Ha, D. Batra, J. Wang, B. Bucher.

VLFM: Vision-language frontier maps for zero-shot semantic navigation. In *Proceedings of IEEE International Conference on Robotics and Automation*, Yokohama, Japan, pp. 42–48, 2024. DOI: 10.1109/ICRA57147.2024.10610712.

[39] X. Zhu, Z. Li, X. Wang, X. Jiang, P. Sun, X. Wang, Y. Xiao, N. Yuan. Multi-modal knowledge graph construction and application: A survey. *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 2, pp. 715–735, 2024. DOI: 10.1109/TKDE.2022.3224228.

[40] S. Wang, J. Yue, J. Liu, Q. Tian, M. Wang. Large-scale few-shot learning via multi-modal knowledge discovery. In *Proceedings of the 16th European Conference on Computer Vision*, Glasgow, UK, pp. 718–734, 2020. DOI: 10.1007/978-3-030-58607-2_42.

[41] X. Li, Z. Wang, J. Yang, Y. Wang, S. Jiang. KERM: Knowledge enhanced reasoning for vision-and-language navigation. In *Proceedings of IEEE/ CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, Canada, pp. 2583–2592, 2023. DOI: 10.1109/CVPR52729.2023.00254.

[42] C. Lin, Y. Jiang, J. Cai, L. Qu, G. Haffari, Z. Yuan. Multimodal transformer with variable-length memory for vision-and-language navigation. In *Proceedings of the 17th European Conference on Computer Vision*, Tel Aviv, Israel, pp. 380–397, 2022. DOI: 10.1007/978-3-031-20059-5_22.

[43] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, L. Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, USA, pp. 6077–6086, 2018. DOI: 10.1109/CVPR.2018.00636.

[44] G. Grisetti, C. Stachniss, W. Burgard. Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 34–46, 2007. DOI: 10.1109/TRO.2006.889486.

[45] S. Kohlbrecher, O. von Stryk, J. Meyer, U. Klingauf. A flexible and scalable slam system with full 3D motion estimation. In *Proceedings of IEEE International Symposium on Safety, Security, and Rescue Robotics*, Kyoto, Japan, pp. 155–160, 2011. DOI: 10.1109/SSRR.2011.6106777.

[46] G. Chen, C. Fan, J. Sun, J. Xia. Mean square exponential stability analysis for Itô stochastic systems with aperiodic sampling and multiple time-delays. *IEEE Transactions on Automatic Control*, vol. 67, no. 5, pp. 2473–2480, 2022. DOI: 10.1109/TAC.2021.3074848.

[47] G. Chen, J. Xia, J. H. Park, H. Shen, G. Zhuang. Robust sampled-data control for switched complex dynamical networks with actuators saturation. *IEEE Transactions on Cybernetics*, vol. 52, no. 10, pp. 10909–10923, 2022. DOI: 10.1109/TCYB.2021.3069813.

[48] G. Chen, G. Du, J. Xia, X. Xie, J. H. Park. Controller synthesis of aperiodic sampled-data networked control system with application to interleaved flyback module integrated converter. *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 70, no. 11, pp. 4570–4580, 2023. DOI: 10.1109/TCSI.2023.3295940.

[49] G. Chen, J. Xia, J. H. Park, H. Shen, G. Zhuang. Sampled-data synchronization of stochastic Markovian jump neural networks with time-varying delay. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 8, pp. 3829–3841, 2022. DOI: 10.1109/TNNLS.2021.3054615.

[50] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng. ROS: An open-source robot operating system. In *Proceedings of ICRA Workshop on Open Source Software*, Kobe, Japan, Article number 5, 2009.

[51] P. Anderson, A. Chang, D. S. Chaplot, A. Dosovitskiy, S. Gupta, V. Koltun, J. Kosecka, J. Malik, R. Mottaghi, M. Savva, A. R. Zamir: On evaluation of embodied navigation agents, [Online], Available: https://arxiv.org/abs/1807.06757, 2018.

[52] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, D. Batra. Habitat: A platform for embodied AI research. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, Seoul, Republic of Korea, pp. 9338–9346, 2019. DOI: 10.1109/ICCV.2019.00943.

[53] D. An, H. Wang, W. Wang, Z. Wang, Y. Huang, K. He, L. Wang. ETPNav: Evolving topological planning for vision-language navigation in continuous environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. DOI: 10.1109/TPAMI.2024.3386695.

Hui Yuan received the M. Sc. degree in control engineering from Xiangtan University, China in 2021. He is currently a Ph. D. degree candidate in control science and engineering at Beijing University of Technology, China.

His research interests include multimodal learning and embodied AI.

E-mail: hui.yuan@cripac.ia.ac.cn

ORCID iD: 0000-0002-9099-7259

Yan Huang received the Ph. D. degree in pattern recognition and intelligent system from University of Chinese Academy of Sciences, China in 2017. He is currently an associate professor with the Institute of Automation, Chinese Academy of Sciences (CASIA), China. He has published more than 100 papers in the leading international journals and conferences such as *IEEE Transactions on Pattern Analysis and Machine Intelligence*, NeurIPS, CVPR, and ICCV. He has obtained awards such as the Presidential Special Award of CAS, Excellent Doctoral Thesis of both CAS and CAAI, NVIDIA Pioneering Research Award, Baidu Fellowship, CVPR 2014 Workshop Best Paper Award, ICPR 2014 Best Student Paper Award, and RACV 2016 Best Poster Award. He has served as a co-chair of ICCV 2019 Workshop on Cross-Modal Learning in Real World and CVPR 2020 Workshop on Multimodal Learning.

His research interests include computer vision and cross-modal data analysis.

E-mail: yhuang@nlpr.ia.ac.cn

ORCID iD: 0000-0002-8239-7229

Naigong Yu received the B. Eng. degree in information processing display and recognition from the Harbin Institute of Technology, China in 1989, the M. Eng. degree in control science and engineering from Shanghai Jiao Tong University, China in 1996, and the Ph. D. degree in pattern recognition and intelligent systems from the Beijing University of Technology, China in 2005. He worked as a visiting scholar with the University of Alberta, Canada in 2011. He is currently a professor with the Faculty of Information Technology, Beijing University of Technology, China.

His research interests include computational intelligence, intelligent systems, and robotics.

E-mail: yunaigong@bjut.edu.cn (Corresponding author)
ORCID iD: 0000-0002-8452-4623

Dongbo Zhang received the M. Sc. degree in computer application technology and the Ph. D. degree in control science and engineering from Hunan University, China in 2001 and 2007, respectively. He has been a professor with the College of Automation and Electronics Information, Xiangtan University, China since 2006.

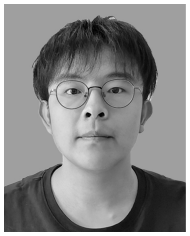
His research interests include pattern recognition, image processing, machine learning, and machine intelligence.

E-mail: zhadonbo@163.com
ORCID iD: 0000-0003-3776-3426

Zetao Du received the B. Eng. degree in computer science from Beijing Forest University, China in 2022. He is currently a master student in computer science from Shanghaitech University, China.

His research interests include multimodal learning and robotics.

E-mail: duzt2022@shanghaitech.edu.cn
ORCID iD: 0009-0002-3105-2092



Ziqi Liu is currently a bachelor student in electronic information engineering at Beijing University of Technology, China.

His research interests include robotic control and hardware system integration.

E-mail: liuziqi0502@emails.bjut.edu.cn
ORCID iD: 0009-0001-2976-144X

Kun Zhang received the M. Eng. degree in bridge and tunnel engineering from the Southwest Jiaotong University (SWJTU), China in 2015, and is currently working as an assistant engineer at the Institute of Automation, Chinese Academy of Sciences(CASIA), China.

His research interests include object detection and multiple object tracking (MOT).

E-mail: kun.zhang@cripac.ia.ac.cn
ORCID iD: 0009-0006-8953-4149

Machine Intelligence Research

Citation: H. Yuan, Y. Huang, N. Yu, D. Zhang, Z. Du, Z. Liu, K. Zhang. Multimodal pretrained knowledge for real–world object navigation. *Machine Intelligence Research*, vol.22, no.4, pp.713–729, 2025. https://doi.org/10.1007/s11633–024–1537–x