

Research Article

Fast Transfer Navigation for Autonomous Robots

Chen Wang , **Xudong Li**, **Xiaolin Tao**, **Kai Ling**, **Quhui Liu**, and **Gan Tao**

School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Correspondence should be addressed to Chen Wang; wangchen100@163.com

Received 19 August 2021; Accepted 23 October 2021; Published 2 December 2021

Academic Editor: L. Fortuna

Copyright © 2021 Chen Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Navigation technology enables indoor robots to arrive at their destinations safely. Generally, the varieties of the interior environment contribute to the difficulty of robotic navigation and hurt their performance. This paper proposes a transfer navigation algorithm and improves its generalization by leveraging deep reinforcement learning and a self-attention module. To simulate the unfurnished indoor environment, we build the virtual indoor navigation (VIN) environment to compare our model and its competitors. In the VIN environment, our method outperforms other algorithms by adapting to an unseen indoor environment. The code of the proposed model and the virtual indoor navigation environment will be released.

1. Introduction

Autonomous navigation is the key capability of an intelligent robot, enabling it to arrive at the destination on the optimal route. It plays an important role in many fields, i.e., mobile robotics [1], unmanned aerial vehicles [2], human-machine interaction [3], etc.

In recent years, the lack of skilled labor promotes the development of social robots for home renovation. Therefore, this paper focuses on the indoor navigation of these robots, especially the efficiency of robotic navigation. The methods for autonomous navigation can be classified as the model-based approaches [4–6] and the model-free approaches [7].

Model-based methods [4–6] explicitly implement robotic location and mapping construction for robotic navigation. Cui et al. [4] modeled the path of the vehicle as pieces of lines and proposed to simultaneously estimate the positions of the global positioning system receiver and the parameters of the line. Yoon and Raychowdhury [5] designed NeuroSLAM to mimic place cells and head direction cells in a rodent brain for robotic location. Engel et al. [6] implemented large-scale direct monocular SLAM by operating on image intensities both for tracking and mapping. The application of radar sensors for situational

awareness leads to the high cost of hardware. In addition, iterative location and mapping inevitably cause the accumulated error, which hurts the navigation performance of intelligent robots. Given the positions of a robot and a destination, algorithms [8] can plan the optimal path between them, and then dynamic control [9, 10] is used to implement robotic motion and navigation.

Model-free methods adopt deep reinforcement learning (DRL) [11] to model the intelligent agent, where the interaction between the agent and environment is formulated as a Markov decision process (MDP).

In our previous work [12], we proposed a tutor-student network for transfer navigation. Specifically, the tutor module must be pretrained in a maze to guide the student module navigate in an unseen maze. However, this two-stage training process inevitably causes high computational costs. In this paper, we employ an attention mechanism to enhance the feature representation in the tutor module and facilitate training for fast transfer navigation. Besides, a comprehensive navigation platform, virtual indoor navigation (VIN), based on ViZDoom [13], is built to simulate the indoor scenes of rough houses. To our knowledge, it is the first 3D virtual environment for robotic navigation in rough houses. In conclusion, our contribution could be formatted in two ways:

- (1) We proposed an attention-based tutor-student model (ATS) for fast transfer navigation. We incorporate a self-attention module into the tutor-student network for enhancing the feature representation and facilitating the training of robots.
- (2) We built the first 3D rough indoor environment for robotic navigation.
- (3) We evaluated the proposed module in VIN and validated its effectiveness on fast transfer navigation.

2. Related Works

2.1. Model-Free Navigation. With its rapid development, deep learning is blended with reinforcement learning to achieve model-free navigation. A Siamese actor-critic network [14] was proposed for the goal-driven visual navigation by introducing the features of a goal. In addition to the goal, Mirowski et al. [15] leveraged multimodal sensory inputs for depth prediction and loop closure classification tasks. Mirowski et al. [16] employed a dual pathway structure to incorporate the locale-specific knowledge with general policies for navigating in multicities. Wang et al. [17] proposed a reinforced cross-modal matching (RCM) approach that enforces cross-modal grounding via reinforcement learning and introduced a self-supervised imitation learning (SIL) method to encourage an agent imitate its own past, good decisions for improving its generalization. Mousavian et al. [18] exploited an advanced detector concurrently with semantic segmentation to develop the transfer ability of agents. Li et al. [19] employed role-playing learning formulated with DRL for robotic navigation in a socially concomitant manner. Zeng et al. [12] presented a tutor-student framework which consists of a “tutor” module and a “student” module. The “tutor” module can facilitate the training of the “student” module with its general features and knowledge.

2.2. Visual Attention Mechanism. Attention mechanism [20–22] has been used frequently in various fields of computer vision, which makes more efficient use of regions of interest and releases distraction. Mnih et al. [20] adopted a recurrent network to adaptively extract features from a sequence of regions at high resolution. Google [21] proposed transformer, an attention-based network architecture, whose self-attention module replaces the encoder and decoder in traditional translation methods. It can greatly improve the efficiency of model inference and outperform other existing methods. Hu et al. [22] designed a squeeze and excitation block to focus on those significant features. Except channel-wise attention, Yu et al. [23] introduced conditional pixel-wise attention to improve the performance of naive Lite-HRNet by means of element-wise weighting operation.

3. Methods

3.1. Formulation. Robotic navigation is formulated as MDP, which models an agent with deep reinforcement learning to interact with the environment as shown in Figure 1.

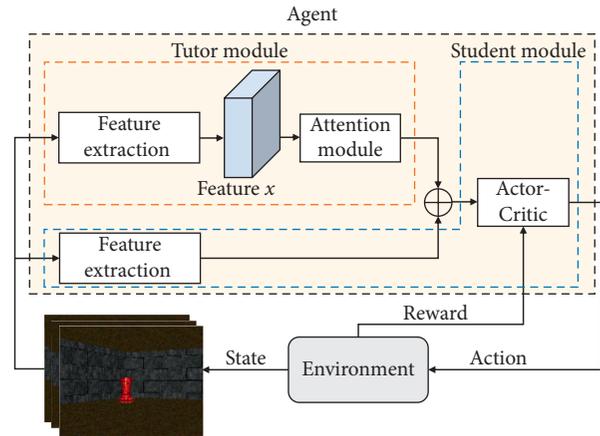


FIGURE 1: Pipeline of the attention-based agent.

Specifically, the agent implements an action after observing the current state of the interactive environment and receives a reward simultaneously. In visual navigation, the state is an RGB image observed by the agent, and the reward implies whether the agent arrives at its destination or not. It aims at maximizing the accumulated rewards with the lowest time cost by learning an optimal policy. Considering the distribution of all potential actions, the accumulated reward R is formulated as the mathematical expectation of all possible rewards according to the implemented actions.

$$R(s_t; \theta) = \mathbb{E} \left(\sum_{a_t} R_t \pi(a_t | s_t; \theta) \right), \quad (1)$$

where θ represents the learnable parameters of policy π . It can be optimized iteratively to maximize the respected accumulated rewards R by the policy gradient algorithm.

$$\theta_k = \theta_{k-1} + \alpha \cdot \mathbb{E} [R_t \nabla_{\theta} \log \pi(a_t | s_t; \theta_{k-1})], \quad (2)$$

where $\pi(a_t | s_t; \theta_{k-1})$ is the probability of selected action a_t under state s_t following the policy π and α is the learning rate.

Advantage Actor-Critic (A2C) [24], a typical policy gradient method, is used to learn this optimal policy for robotic navigation. The framework of the A2C network contains two head blocks, i.e., actor block and critic block. The critic block is used to estimate the accumulated reward, and the actor block aims to predict the optimal action for navigation.

3.2. Attention-Based Tutor-Student Net. We proposed an attention-based tutor-student net to build the agent which is composed of a tutor module and student module. The tutor module trained beforehand in other mazes can extract general features for navigating in an unseen maze. The student module extracts locale-specific features and feeds them into the actor-critic network with general features (Figure 1). Training the tutor module beforehand inevitably increases the temporal cost. To mitigate this problem, we design a pixel-wise attention module in the tutor module to enhance the representation of general features.

There is much similar and abundant information in a series of individual frames. Therefore, every four RGB images are converted into gray images and stacked as a four-channel input tensor in data preprocessing. Feature maps are extracted from this input tensor by feature extraction modules which consist of four convolution layers (see Figure 2).

In the tutor module, we take advantage of the attention mechanism to bridge the dependency within those features and enhance the feature representations. Since the in-plane information plays an important role in the performance of an agent, we emphasize on bridging the in-plane relationship. Specifically, the $C \times H \times W$ feature maps from the feature extraction block are separated into $H \cdot W$ pixel-wise embeddings along the channel axial, as shown in Figure 3. Each embedding X is projected to a query vector Q , a key vector K , and a value vector V through mapping matrices W_Q , W_K , and W_V , respectively. For the i th embedding X_i , the distribution A_i of query Q_i measures the similarity between Q_i and all key vectors.

$$A_i = \text{soft max} \left(\frac{Q \cdot K^T}{\sqrt{d}} \right), \quad (3)$$

where d is the dimension of the key.

After obtaining the attention matrix A , we can generate a feature representation Z by weighting the value vector V adaptively according to the attention matrix A . Essentially, the attention block aggregates similar features globally, which can effectively alleviate the local dependency in convolutional operations.

4. Virtual Environment

Currently, there is no platform which contains various unfurnished rooms for robotic navigation. To verify the effectiveness of the proposed algorithm, a virtual unfurnished indoor environment is built based on Doom [13]. Specifically, to construct a building environment, a floor plan dataset is collected online with crawling technology, which consists of 4 parts: single-bedroom set, two-bedroom set, three-bedroom set, and four-bedroom set. Each set includes 20 plans with various shapes and scales. Figure 4 shows four randomly selected examples from each part. For each floor plan, a 3D navigation environment can be built using a generation tool. Finally, we obtain an indoor navigation platform which includes 80 different 3D scenes.

In the 3D navigation environment, an agent can observe the unfurnished room by using a first-person camera, as shown in Figure 5. The ceiling and floor are all earth yellow, and the wall is covered with the texture of brick patterns. In addition, we set a red cylinder object to represent the destination. To achieve the goal, the agent aims to arrive at the destination by moving forward, turning left, or turning right.

Except the first-person view image, RGBD image and labeling image are able to be captured in our platform. In addition, an agent can also obtain its location and orientation as auxiliary information.

5. Experiments

The navigation task of a decoration robot is to be able to reach a specified target location on an optimal route from any location in a single decoration environment to a single scene. In this paper, we test the proposed self-attention-based algorithm in an interior decoration environment built on ViZDoom. Specifically, we evaluate our model in four different house types, i.e., single-bedroom setting, two-bedroom setting, three-bedroom setting, and four-bedroom setting. Such experiments are effective in testing our model's ability to navigate across scenes.

5.1. Experiment Configuration. At the beginning of each navigation task, the intelligent body randomly initializes its position. This can effectively avoid the overfitting of the smart body to a single navigation route and improve the generalization performance of the model. In each decoration scene, the agent is trained 100 episodes, and each episode is trained 50000 steps. Considering that the input sequential images contain a lot of redundant information, we use three sequential frames as a state tensor with the image size of 640×480 . Our model is implemented by TensorFlow [25] implementation, using Adam optimizer for parameter optimization and training it on an Nvidia GeForce GTX 1080 Ti GPU.

5.2. Experiment Results. To validate the effectiveness of our proposed model, it is fairly compared with A2C and tutor-student models in the same experiment configuration. The tutor-student model is implemented in a two-stage manner. Therefore, we illustrate the experiment results in the following two sections.

5.2.1. Training from Scratch. ATS, tutor-student, and A2C models are trained from scratch in four mazes with different styles, i.e., one-bedroom maze, two-bedroom maze, three-bedroom maze, and four-bedroom maze. Their reward curves during the training process are shown in Figure 6. In all mazes, the reward curves of ATS not only raise and converge much earlier than the other two models but also receive higher rewards, especially in two-bedroom (see Figure 6(b)), three-bedroom (see Figure 6(c)), and four-bedroom mazes (see Figure 6(d)). Except the single-bedroom maze (see Figure 6(a)), the tutor-student model performs similarly as A2C. The advantage of the ATS model benefits from the self-attention module which can effectively facilitate the agent's training process.

5.2.2. Transfer Navigation. To evaluate the performance of the three models in a transfer navigation task, we also select one maze for testing from each part in VIN. Similar to the tutor-student network, the tutor module of ATS is trained in the two-bedroom maze and three-bedroom maze and fine-tuned in the one-bedroom maze and four-bedroom maze for transfer navigation. To illustrate their advantage in robotic

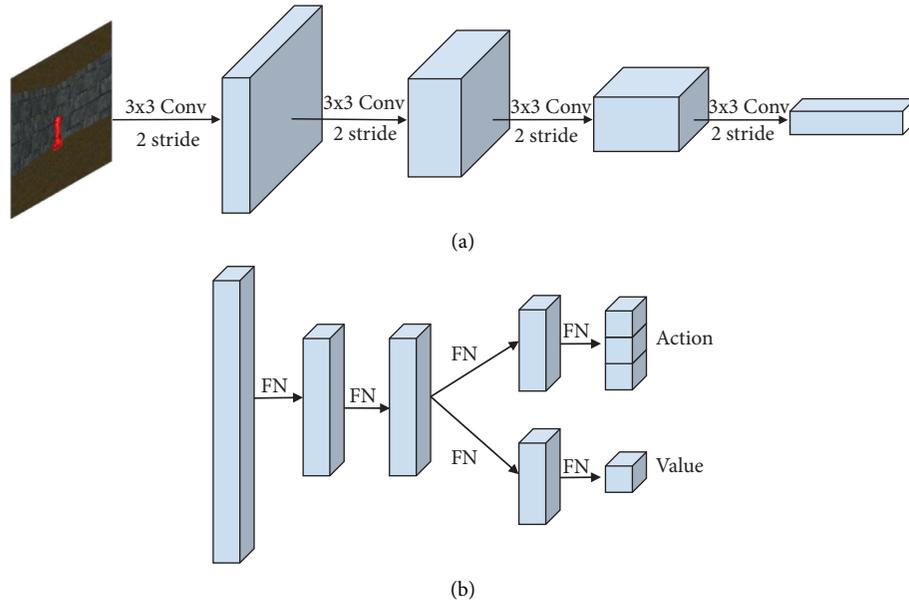


FIGURE 2: Pipeline of the attention-based tutor-student network. (a) Feature extraction module. (b) Actor-critic module.

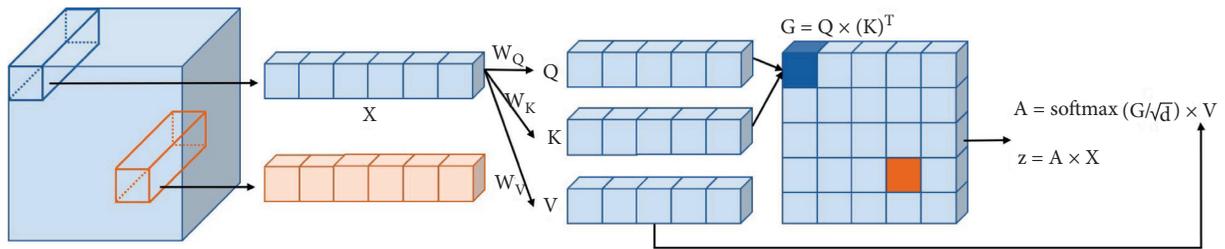


FIGURE 3: Illustration of the attention block.

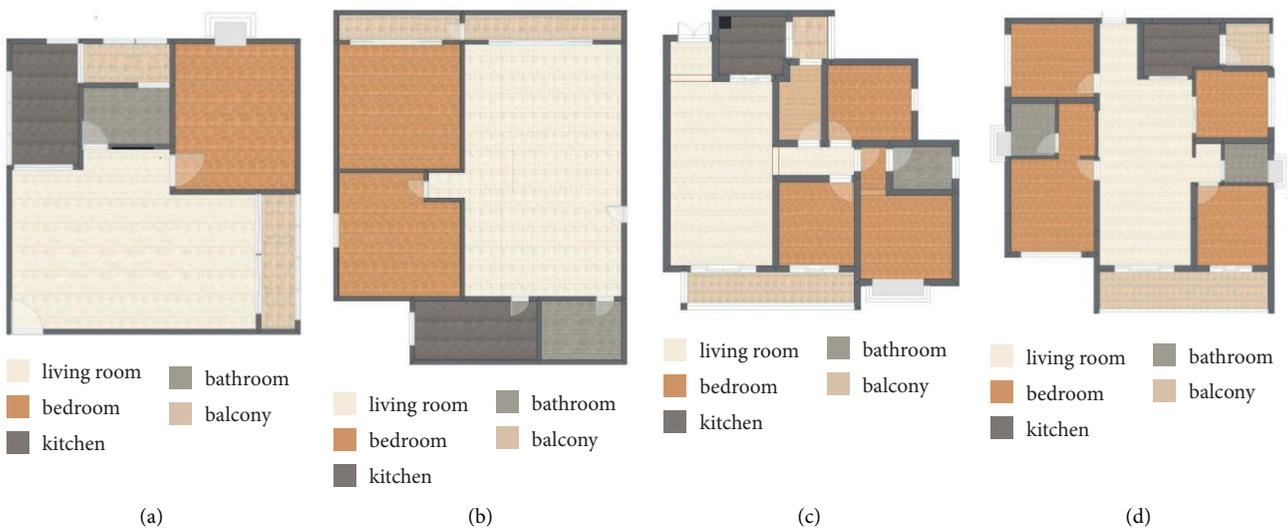


FIGURE 4: 2D floor plans for unfurnished indoor scenes: (a) single-bedroom set, (b) two-bedroom set, (c) three-bedroom set, and (d) four-bedroom set.

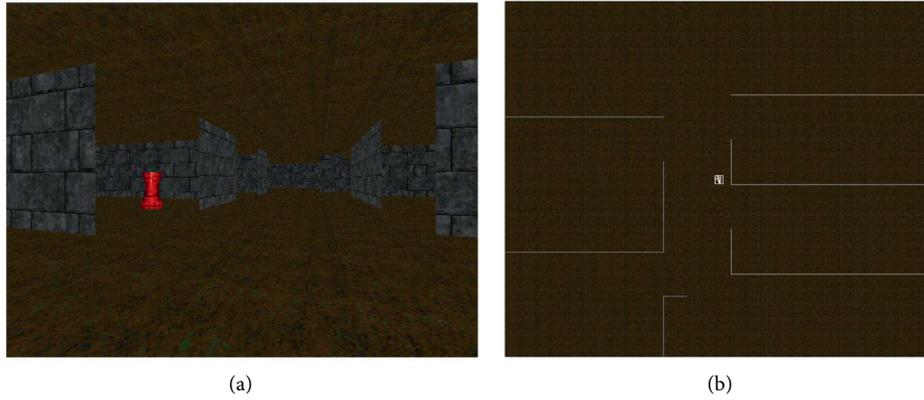


FIGURE 5: Illustration of a virtual indoor navigation environment. (a) First-person view of the agent in the environment. (b) 2D floor plan of the environment.

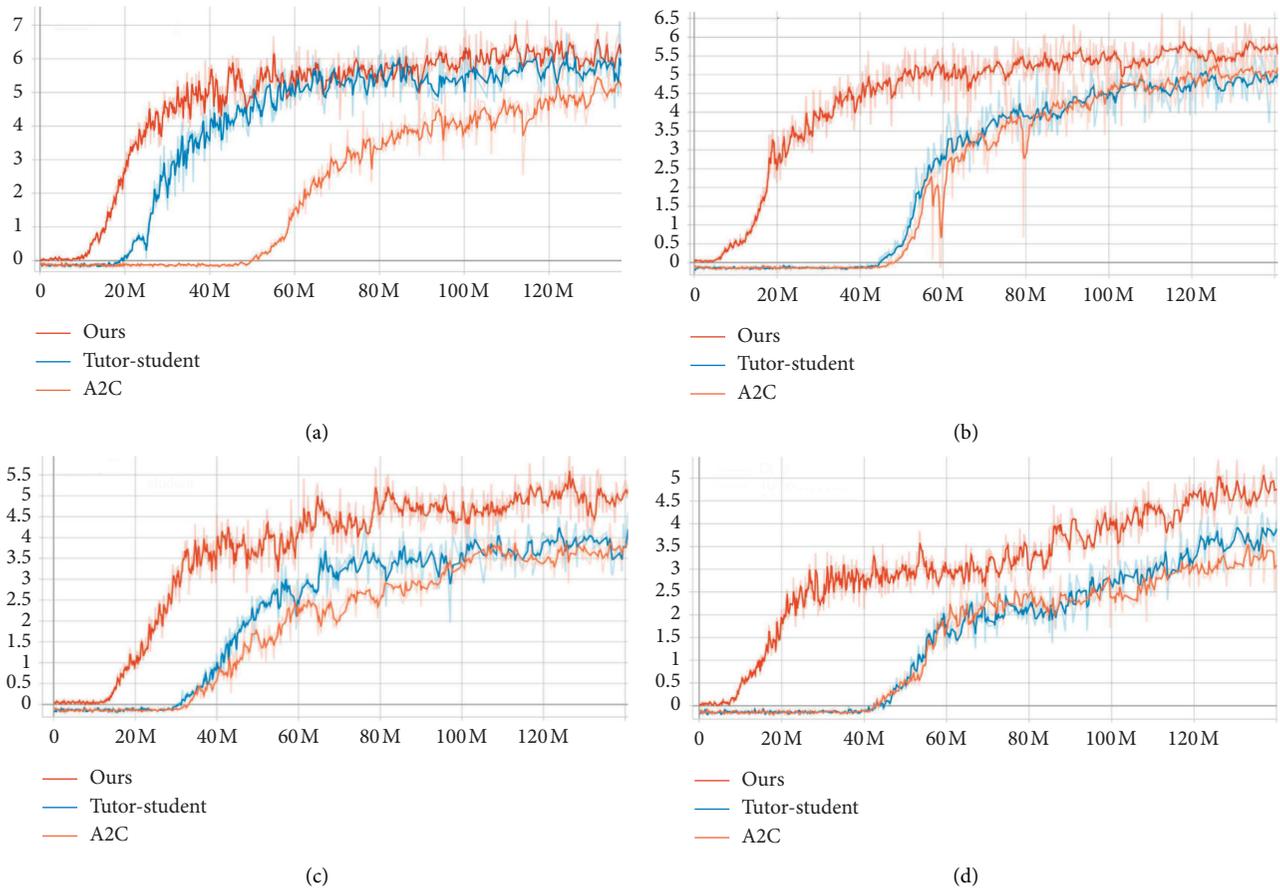


FIGURE 6: Average reward curves of ours, tutor-student, and A2C model training with different bedroom settings. (a) One-bedroom maze. (b) Two-bedroom maze. (c) Three-bedroom maze. (d) Four-bedroom maze.

navigation, A2C is trained from scratch in one-bedroom and four-bedroom mazes as baseline.

Figure 7 shows that ATS outperforms the tutor-student network and A2C model in the transfer navigation task. Specifically, in all mazes, the average reward curves of vanilla A2C start to raise at almost 40M steps and converge at 140M steps, while the tutor-student model improves its navigation ability and arrives at the destination earlier than A2C, which

is obvious in transfer navigation in Figure 7(a). Because the structure of the one-bedroom maze is relatively easier than that of the four-bedroom maze, the average reward curves of the ATS and tutor-student network converge towards similar average rewards in Figures 7(a) and 7(c). However, in the four-bedroom maze with a complex structure (see Figures 7(b) and 7(d)), the average reward curves of ATS raise almost at 10M steps and converge to higher average

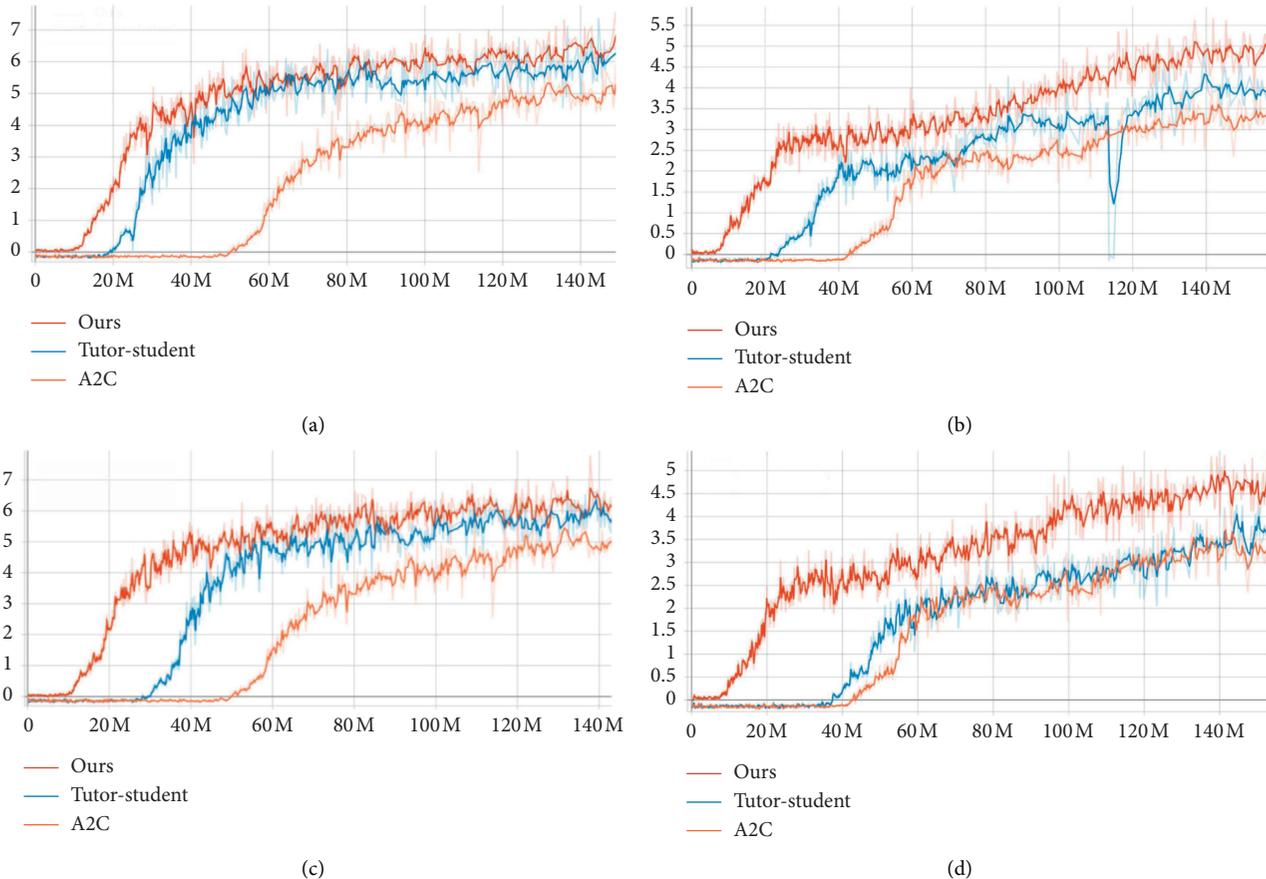


FIGURE 7: Average reward curves of ours, tutor-student, and A2C models in transfer navigation. (a) Transfer navigation from the two-bedroom maze to the one-bedroom maze. (b) Transfer navigation from the two-bedroom maze to the four-bedroom maze. (c) Transfer navigation from the three-bedroom maze to the one-bedroom maze. (d) Transfer navigation from the three-bedroom maze to the four-bedroom maze.

rewards than the tutor-student network and the A2C model, which benefits from the attention module.

6. Conclusion

Indoor navigation plays an important role in robot movement. The state-of-the-art tutor-student model achieves transfer navigation in two steps, which greatly increases the temporal cost and decreases its efficiency. Our paper proposes an ATS model to facilitate its training process for fast transfer navigation. Especially, a self-attention network is used to reweight the extracted features and focus on the task-aware features in the tutor module. To evaluate our model in unfurnished indoor navigation, we build VIN, a virtual navigation platform based on the Doom engine. The experiment results illustrate that the self-attention module effectively speeds the training process up and benefits to the fast transfer navigation of the proposed ATS model.

Data Availability

The training data used to support the findings of this study have been deposited in the GitHub repository (<https://github.com/WangChen100/vizdoomgymmaze>).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (U1813202 and 61773093), National Key R&D Program of China (2018YFE0203900), Research Programs of Sichuan Science and Technology Department (20ZDYF0028), and Important Science and Technology Innovation Projects in Chengdu (2018-YF08-00039-GX).

References

- [1] T. H. Chung, G. A. Hollinger, and V. Isler, "Search and pursuit-evasion in mobile robotics," *Autonomous Robots*, vol. 31, no. 4, Article ID 299, 2011.
- [2] G. Conte and P. Doherty, "Vision-based unmanned aerial vehicle navigation using geo-referenced information," *EURASIP Journal on Applied Signal Processing*, vol. 2009, pp. 1–18, 2009.
- [3] C.-Y. Ma, J. Lu, Z. Wu et al., "Self-monitoring navigation agent via auxiliary progress estimation," in *Proceedings of the*

- International Conference on Learning Representations*, New Orleans, LA, USA, May 2019.
- [4] Y. Cui, S. Sam Ge, and S. Ge, "Autonomous vehicle positioning with GPS in urban canyon environments," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 1, pp. 15–25, 2003.
 - [5] J.-H. Yoon and A. Raychowdhury, "Neuroslam: a 65-nm 7.25-to-8.79-tops/w mixed-signal oscillator-based SLAM accelerator for edge robotics," *IEEE Journal of Solid-State Circuits*, vol. 56, no. 1, pp. 66–78, 2021.
 - [6] J. Engel, T. Schöps, and D. Cremers, *LSD-SLAM: Large-Scale Direct Monocular SLAM*, Springer, Berlin, Germany, 2014.
 - [7] R. Jiang, S. S. Ge, N. Teja Tangirala, and T. H. Lee, "Interactive navigation of mobile robots based on human's emotion," in *Proceedings of the International Conference on Social Robotics*, Kansas City, MO, USA, November 2016.
 - [8] R. Caponetto, L. Fortuna, S. Graziani, and M. G. Xibilia, "Genetic algorithms and applications in system engineering: a survey," *Transactions of the Institute of Measurement and Control*, vol. 15, no. 3, pp. 143–156, 1993.
 - [9] M. Bucolo, A. Buscarino, C. Famoso, L. Fortuna, and M. Frasca, "Control of imperfect dynamical systems," *Nonlinear Dynamics*, vol. 98, no. 4, pp. 2989–2999, 2019.
 - [10] P. Arena, A. Basile, M. Bucolo, and L. Fortuna, "An object oriented segmentation on analog CNN chip," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 50, no. 7, pp. 837–846, 2003.
 - [11] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
 - [12] F. Zeng, C. Wang, and S. S. Ge, "Tutor-guided interior navigation with deep reinforcement learning," *IEEE Transactions on Cognitive and Developmental Systems*, p. 1, 2020.
 - [13] M. Wydmuch, M. Kempka, and W. Jaskowski, "Vizdoom competitions: playing doom from pixels," *IEEE Transactions on Games*, vol. 11, no. 3, pp. 248–259, 2019.
 - [14] Y. Zhu, R. Mottaghi, E. Kolve et al., "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3357–3364, Singapore, May 2017.
 - [15] P. Mirowski, R. Pascanu, F. Viola et al., "Learning to navigate in complex environments," in *Proceedings of the International Conference on Learning Representations*, Toulon, France, April 2017.
 - [16] P. Mirowski, M. K. Grimes, M. Malinowski et al., "Learning to navigate in cities without a map," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 2424–2435, Montréal, Canada, December 2018.
 - [17] X. Wang, Q. Huang, A. Celikyilmaz et al., "Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6629–6638, Long Beach, CA, USA, June 2019.
 - [18] A. Mousavian, A. Toshev, M. Fišer, J. Kosecka, A. Wahid, and J. Davidson, "Visual representations for semantic target driven navigation," in *Proceedings of the International Conference on Robotics and Automation*, pp. 8846–8852, Montreal, Canada, May 2019.
 - [19] M. Li, R. Jiang, S. S. Ge, T. Lee, and H. Lee, "Role playing learning for socially concomitant mobile robot navigation," *CAA Transactions on Intelligence Technology*, vol. 3, no. 1, pp. 49–58, 2018.
 - [20] V. Mnih, N. Heess, A. Graves, and K. kavukcuoglu, "Recurrent models of visual attention," *Advances in Neural Information Processing Systems*, vol. 27, pp. 2204–2212, 2014.
 - [21] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017.
 - [22] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020.
 - [23] C. Yu, B. Xiao, C. Gao et al., "Lite-hrnet: a lightweight high-resolution network," 2021, <https://arxiv.org/abs/2104.06403>.
 - [24] V. Mnih, A. Puigdomènech Badia, M. Mirza et al., "Asynchronous methods for deep reinforcement learning," *Proceedings of International Conference on Machine Learning*, vol. 48, pp. 1928–1937, 2016.
 - [25] M. Abadi, A. Agarwal, B. Paul et al., "Tensorflow: large-scale machine learning on heterogeneous distributed systems," 2016, <https://arxiv.org/abs/1603.04467>.