Research and application of path planning algorithm based on deep learning

Xulong Jia

Shanxi Agricultural University, Software Academy, Jinzhong City, China, 030600

harryjxl@163.com

Abstract. This paper mainly introduces the research and application of path-planning algorithms based on deep learning. Firstly, it introduces the background significance of path planning, the current status of domestic and international research and the content and methods, and then analyzes the application methods of deep learning in path planning, including deep reinforcement learning, convolutional neural network and recurrent neural network, and the comparison between different algorithms. Then, the implementation of a deep learning-based path-planning algorithm is introduced in detail, including system architecture and design, data preprocessing and model training, path-planning implementation and optimization, etc. In the Experiments and Results Analysis section, the analysis and comparison of experimental results, experimental conclusions and expansion directions are summarized. Finally, in the conclusion and outlook section, the research contributions and application values of this paper are described, and the research deficiencies and future expansion directions of deep learning in path planning are proposed. In addition, because of the continuous development of deep learning technology, there are many other possibilities for applying deep learning methods to build more optimal path-planning algorithms in the future.

Keywords: path planning, deep learning, convolutional neural network, recurrent neural network, model training.

1. Introduction

Path planning is one of the important problems in robotics and autonomous driving, and it is gradually becoming a frontier research direction in artificial intelligence. Path planning needs to consider different constraints to make paths safer, more efficient and more accurate. However, traditional path-planning algorithms and rule-based methods may not be able to handle complex environments and perform poorly when they encounter uncertainty. Therefore, the application of deep learning in path planning has received a lot of attention and is considered to address the limitations of traditional path-planning algorithms. Deep learning has been successfully applied to image recognition, natural language processing, etc. As for path planning, deep learning can generate better path-planning solutions by training on the data. At the same time, deep learning can also continuously optimize the path planning results by adaptive means to achieve better results.

In recent years, the application of deep learning in the field of path planning has gained a lot of attention. Researchers at home and abroad have explored various ways to use deep learning to solve this problem. For example, reinforcement learning can be used to train a robot to complete a path for a

© 2023 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

specific task in an environment. Convolutional neural networks and recurrent neural networks have also been widely used in path planning, and there have been some successful applications. In addition, some researchers have proposed new methods to improve the effectiveness of path planning, such as using genetic algorithms or particle swarm optimization methods for search. However, there are still many challenges to overcome. First, deep learning models usually require a large amount of data to get good performance. Therefore, how to obtain enough data efficiently becomes a key issue.

The research of this paper is about the research and application of deep learning-based path-planning algorithms. In this paper, the research will be carried out through the basics of deep learning, the application methods of deep learning in path planning, the comparison of different algorithms in path planning, the implementation of path planning algorithms based on deep learning, and the experiments and result analysis. The research methods in this paper mainly include a literature review, theoretical analysis, algorithm design, data pre-processing, experiments and result analysis.

2. Deep learning in path planning

2.1. Deep learning basics

In path planning, deep reinforcement learning has a clear advantage to learn the optimal path selection strategy by exploring-utilizing policies. For example, in autonomous driving, vehicles need to choose the optimal path based on real-time road conditions in order to reach their destinations. Therefore, training vehicles to learn optimal path-planning strategies by deep reinforcement learning has become an important research direction. Deep reinforcement learning differs from traditional rule-based path planning algorithms in that it does not require complex rules to be defined or costly manual feature extraction before learning. Instead, it only requires a model that interacts with the environment to learn the optimal path-planning strategy by observing the current state and reward signals.

2.2. Deep learning application methods in path planning

2.2.1. Deep reinforcement learning. In the application of path planning, the combination of reinforcement learning and deep learning has obvious advantages. Reinforcement learning is able to obtain the optimal strategy for a variety of complex environments through continuous exploration and learning, while deep learning is able to perform effective data modelling and feature extraction for complex, high-dimensional state spaces, thus improving the effectiveness and learning speed of reinforcement learning. Therefore, deep reinforcement learning can be applied to a wide range of path-planning scenarios, both in traditional intelligent vehicles and in emerging industries, such as drones and self-driving vehicles.

2.2.2. Convolutional neural network. Convolutional Neural Networks (CNN) can process image data more efficiently by using techniques such as local connectivity, weight sharing and pooling [1]. In path planning, by feeding trajectory images into CNNs, spatial structure information such as features, textures and shapes in images can be extracted, and finally, a solution for path planning can be obtained. Specifically, a convolutional neural network can decompose the input image into several small sub-regions, each of which does convolutional operations with a convolutional kernel to obtain an output feature map. These feature maps contain image feature information of different scales, orientations and combinations, which can effectively represent the local features and global structure of the original image.

2.2.3. Recurrent neural network. Recurrent Neural Networks Recurrent Neural Networks (RNN) are a type of neural network widely used to process sequential data [2], which has a recurrent structure that can combine the input of the current moment and the state of the previous moment to jointly calculate the output result. In path planning, recurrent neural networks can be used to represent the current state of the vehicle and thus compute the best next path in the current situation.

Specifically, recurrent neural networks enable iterative updating of input signals and states by building a time series model. In path planning, the recurrent neural network can derive the direction and speed of vehicle travel by simulating the real-time state of the vehicle. Therefore, when the vehicle is in a specific environment, the recurrent neural network can be used to predict the best path based on the surrounding road information. In the implementation, some attention mechanisms can also be added to focus more accurately on those inputs that have a greater impact on path planning.

2.3. Comparison of different algorithms in path planning

In the field of path planning, a variety of algorithms have emerged, two of the more traditional ones are Dijkstra's algorithm[3] and A* algorithm[4]. Each of these algorithms has its own advantages and disadvantages, and the different algorithms in path planning are briefly described below. Dijkstra's algorithm: Dijkstra's algorithm is a typical single-source shortest path algorithm [5], which is used to calculate the shortest distance from the starting point to each of the other vertices. It mainly starts from the starting point and keeps searching outward until the search ends at the end. The algorithm principle: Suppose a weighted directed graph G is divided into two sets A, and B, where set A contains the visited vertices and their shortest path values, and set B contains the unvisited vertices. Dijkstra's algorithm is a blind search algorithm, so the search efficiency is low. O(n2), therefore, the pathfinding efficiency is relatively low in large maps. The flow chart of the algorithm is shown in Figure 1.

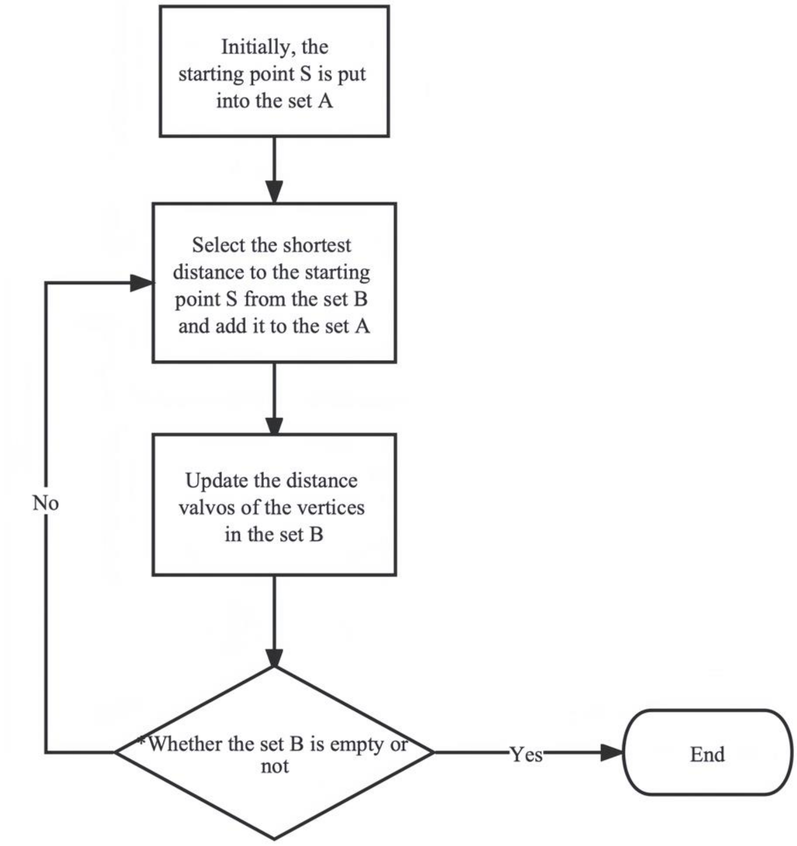
Initially, the starting point S is put into the set A
Select the shortest distance to the starting point S from the set B and add it to the set A
Update the distance valvos of the vertices in the set B
No
*Whether the set B is empty or not
Yes
End

Figure 1. The basic flow of Dijkstra's algorithm [6].

The A* algorithm was published by Hart et al [7] in 1968 and is a commonly used algorithm for pathfinding and graph traversal. It incorporates Dijkstra's algorithm and heuristic functions, and the search weight function is as in equation (1),

$$F(n) = g(n) + h(n) \tag{1}$$

In this function, F(n) is the evaluation function of the A* algorithm for each node, which contains two parts of information: g(n) is the actual cost from the starting point to the current node n, which is the distance moved from the starting point to the current node; the distance moved between two adjacent points is 1, and the further the current point is from the starting point, the larger this value is. h(n) is the evaluation value of the distance from the current node n to the endpoint, which is an estimate of the moving distance from the current node to the endpoint [8]. That is the heuristic function [9]. It is based on Dijkstra's algorithm and combined with the estimation function to optimize the process of finding the shortest path [10]. The A* algorithm not only improves the efficiency of the path search using the heuristic function but also ensures that it does not extend more states than any other optimal algorithm given the same information.

3. Deep learning-based path planning algorithm implementation

3.1. System architecture and design

The system architecture of the deep learning-based path-planning algorithm consists of three parts: data preprocessing, model training, and path planning. Data preprocessing is responsible for transforming raw data into a data format that can be processed by the neural network; model training refers to the use of deep learning methods to train the path planning model; and path planning refers to the use of the trained model to calculate the best path.

Finally, the scalability and reusability of the path-planning algorithm need to be considered. With the continuous development and changes in the field of intelligent transportation, path-planning algorithms need to be continuously optimized and updated to cope with different scenarios and user needs.

3.2. Data preprocessing and model training

In the process of data pre-processing, in addition to transforming the raw data into a format that can be processed by the neural network, some data cleaning and processing is required. These include but are not limited to removing noisy data, filling in missing data, etc.

During model training, it is usually necessary to randomize the training data to avoid overfitting the model to a specific dataset and thus improve its generalization ability. It is also necessary to divide the dataset into a training set and a validation set for training and evaluating the model.

The ultimate goal is to obtain a model that can compute the path correctly. This requires supervision of the model during the training process to evaluate and optimize its performance. Also, tuning and optimizing the structure and hyperparameters of the model can help improve its performance.

3.3. Path planning implementation and optimization

In path planning, the network model outputs an optimal path once the actual start and end points are entered. The shortest path algorithm can be used to search for the best path. Then, the passing points need to be transformed into the actual driving route and the path needs to be adjusted according to the real-time traffic conditions. Factors such as peak traffic flow and traffic jam conditions usually need to be taken into account to choose the best path solution.

In addition, with the growth of the urban population and the continuous congestion of urban traffic, as well as the emergence of new technologies, path-planning algorithms need to be constantly updated and improved. For example, real-time traffic information, weather information, road construction and other factors can be incorporated into path planning to improve the accuracy and practicality of path planning. Also, more advanced deep learning models and algorithms, such as neural networks and deep reinforcement learning, can be introduced to accurately predict and plan paths and perform multidimensional path optimization.

4. Experiments and results analysis

4.1. Analysis and comparison of experimental results

In path planning experiments, we can consider the map as a weighted directed graph. We have conducted path planning experiments with the A* algorithm and the Dijkstra algorithm in a more complex map, and the experimental results show that the A* algorithm has a shorter path length compared with the Dijkstra algorithm. This is because the A* algorithm can optimize the search according to the heuristic function. The experimental results are shown in Figure 2,

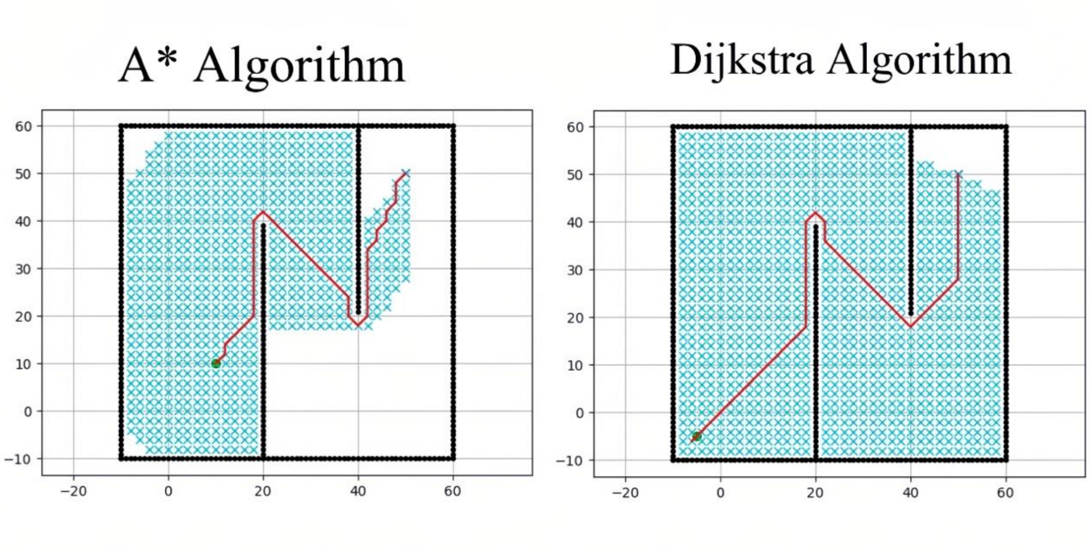

Figure 2. Path planning results for both algorithms [11].

4.2. Experimental conclusions and extension directions

A* algorithm and the Dijkstra algorithm are the most commonly used algorithms in solving path-planning problems. Experimental results show that the A* algorithm has advantages in complex maps to determine the shortest path faster, while the Dijkstra algorithm is suitable for simple maps and situations where search speed is not strictly required. Both algorithms require tuning and optimization of the heuristic functions to obtain better results.

Extension directions:

Firstly, the heuristic functions can be studied in more depth and more accurate heuristic functions can be developed.

Secondly. to study how to optimize path planning algorithms in multidimensional cases to cope with more realistic problems in practical applications.

Thirdly. to study how to implement dynamic path planning, i.e., to calculate the shortest path in real-time in the case of moving objects.

5. Conclusion

The research results of this paper have important practical significance and application value. It provides a new idea and method for realizing more efficient and reliable deep learning-based path planning algorithms, which can play an important role in future applications such as autonomous driving and robotics. Compared with traditional path planning algorithms, the deep learning-based approach can better adapt to complex road conditions and routes, achieving higher efficiency and accuracy. One of the biggest advantages of deep learning-based path planning methods is that they can make full use of massive data and extract features of geospatial data, thus automatically adapting to paths in different environments, such as different regions and road sections, to achieve faster and more efficient path planning.

For the shortcomings of this paper and how to improve it are as follows. First, the diversity of data sets: getting enough data is one of the main problems faced by deep learning models. So, for this problem, more data enhancement techniques can be used to extend the dataset and ensure the diversity of the dataset. Second, the improvement of the experimental setup: some experimental results are mentioned in this paper, but the experimental setup is not described in detail. To better evaluate the performance of the model, more experimental details, such as parameter settings, training and testing datasets, etc., can be provided. Finally, more relevant literature is cited: this paper mentions some knowledge theories about deep learning, and more relevant literature can be cited so that readers can better understand the advantages, disadvantages and application scenarios of these models.

Future research directions can achieve better path-planning results by combining the advantages of traditional models and deep learning models. In this approach, the deep learning model can be responsible for dealing with complex environmental conditions, while the traditional model can effectively deal with known and determinable environmental conditions, thus maximizing the advantages of both approaches.

References

[1] Qi Yanjun, Kong Yueping, Wang Jiajing et al. A gait recognition method combining LSTM and CNN[J]. Journal of Xi'an University of Electronic Science and Technology, 2021,48(05): 78-85. DOI:10.19665/j.issn1001-2400.2021.05.010.

[2] Wang, H. D. Multivariate chaotic time series modelling and prediction based on neural networks [D]. Shenyang University of Aeronautics and Astronautics, 2012.

[3] Gong H, Ni C, Wang P et al. A smooth path planning method based on Dijkstra's algorithm [J/OL]. Journal of Beijing University of Aeronautics and Astronautics:1-10[2023-05-13]. https://doi.org/10.13700/j.bh.1001-5965.2022.0377.

[4] Zhang, Hong-Mei, Ming-Long, et al. Safe Path Planning of Mobile Robot Based on Improved A* Algorithm in Complex Terrains[J]. Algorithms, 2018.

[5] Zhou M, Gao N. Research on Optimal Path based on Dijkstra Algorithms[C]// 2019.

[6] Lu Y, Gao YP, Long JT. A~* algorithm in path planning for mobile robots [J]. Journal of Hubei Normal University (Natural Science Edition), 2022, 42(02):59-65.

[7] Liu, Yunxiang, Du, J., Zhang, Qing. Performance comparison of A~* algorithm and Dijkstra algorithm based on path optimization[J]. Modern Electronics Technology, 2017,40(13):181-183+186.DOI:10.16652/j.issn.1004-373x.2017.13.048.

[8] Hart P E, Nilsson N J, Raphael B. A Formal Basis for the Heuristic Determination of Minimum Cost Paths[J]. IEEE Transactions on Systems Science & Cybernetics, 1972, 4(2): 28-29.

[9] Cui W, Zhu Fazheng. A review of research on path planning algorithms for robot navigation [J/OL]. Computer Engineering and Applications:1-13 [2023-05-13].

[10] Gao Yuyue, Chen Peng,Wang Yan et al. Research on UAV trajectory planning for realistic modelling of earthworks [J]. Information Technology in Civil Engineering, 2022,14(02):1-9.DOI:10.16670/j.cnki.cn11-5823/tu.2022.02.01.

[11] Sakai A, Ingram D, Dinius J, et al. PythonRobotics: a Python code collection of robotics algorithms[J]. 2018.