Optimizing Visual SLAM for Robust and Scalable Mobile Robot Navigation

Kaiyu Xu
International School of Information Science and Engineering, Dalian University of Technology, Digital Media, Dalian, Liaoning Province,116620, China

Keywords: Vision SLAM, Robustness, Scalability, ORB, PPO.

Abstract: The application of vision Simultaneous Localization and Map Building (SLAM) to robot navigation is an important field. But the main challenges faced in this field are how to improve robustness as well as scalability. This thesis aims to optimize the robustness and scalability of vision SLAM in robot navigation. In this paper, it mprove robustness and scalability from both deep learning and image processing. In deep learning, this paper uses Convolutional Neural Network (CNN) algorithm for feature matching and constructs Proximal Policy Optimization (PPO) algorithm model for training. In image processing, this paper mainly uses Oriented FAST and Rotated BRIEF (ORB) technique for feature extraction and loopback detection. The result is that all four experiments are well optimized for robustness and scalability. This allows the robot to operate in complex environments without interference from other external factors, such as lighting conditions and dynamic environments. It also allows the robot to adapt to larger environments and to take on the computational load required to handle larger environments. Ultimately, it is hoped that these methods can be applied to real-life robot navigation.

1 INTRODUCTION

Navigation systems are very famous and robot navigation has a significant place in it. The use of vision Simultaneous Localization and Map Building (SLAM) is one of the important methods to achieve robot navigation. However, traditional robot navigation algorithms mainly rely on pre-constructed environment maps and fixed path planning methods, which will face many problems in practical applications. Firstly, traditional algorithms are environmentally dependent. These algorithms usually need to run in a known environment and lack adaptability to changes in the environment. If the environment changes, maps need to be re-mapped, increasing maintenance costs. Second, traditional algorithms perform poorly when dealing with dynamic environments, making it difficult to respond to moving obstacles or people in real time, affecting the reliability and safety of navigation. For example, classical obstacle avoidance algorithms are slow to respond in dynamic environments and difficult to avoid obstacles effectively (Ghosh, Sanyal and Mukherjee, 2024). In order to solve the above problems, SLAM technology has been developed. SLAM allows a robot to simultaneously build a map of its environment and determine its own position in an unknown or dynamic environment. This allows the robot to navigate autonomously in unknown environments without having to obtain a map of the environment in advance, making it more adaptable. Secondly, SLAM is able to update the environment map in real time, reflecting the environmental changes in time, so that the robot can operate stably even in dynamic environments. Moreover, by fusing multiple sensor data, SLAM can effectively reduce the accumulation of errors and improve the accuracy of robot positioning and navigation. Among them, visual SLAM has excellent performance. Vision SLAM enables mobile robots to simultaneously perform localization and environment mapping through visual information without a map. In contrast to LiDAR, vision SLAM relies on monocular, binocular or RGB-D cameras as the primary sensors (Mur-Artal and Tardós, 2017). The use of cameras as sensors reduces the cost significantly and is obviously much easier to acquire compared to radar. With the development of deep learning and computer vision and big data models, Oriented FAST and Rotated BRIEF (ORB)-SLAM algorithms have emerged

Xu, K.
Optimizing Visual SLAM for Robust and Scalable Mobile Robot Navigation.
DOI: 10.5220/0014307300004718

In *Proceedings of the 2nd International Conference on Engineering Management, Information Technology and Intelligence (EMITI 2025)*, pages 43-48
ISBN: 978-989-758-792-4

(Mur-Artal, Montiel and Tardos, 2015). It is a vision SLAM method based on monocular camera, which can greatly improve the efficiency of mobile robots, and lays a solid foundation for the subsequent development and application of vision SLAM.

Cadena et al. in 'Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age' state that SLAM technology has made significant progress over the last 30 years, enabling robots to able to navigate autonomously in unknown environments, and highlights the robustness and scalability of SLAM in long-term operation (Cadena, Carlone, Carrillo, Latif, Scaramuzza, Neira, Reid and Leonard, 2016). Secondly, Zaffar et al. in 'Sensors, SLAM and Long-term Autonomy: a Review' discuss the various sensors used in SLAM systems and evaluate their performance in long-term autonomous operation, further illustrating the advantages of SLAM technology in dynamic environments (Zaffar, Ehsan, Stolkin and McDonald-Maier, 2018). In Hybrid Navigation Method for Multiple Robots Facing Dynamic Obstacles, the authors propose improved reinforcement learning algorithms to enhance the obstacle avoidance ability of robots in dynamic environments, aiming to solve the navigation problems caused by the accumulation of errors in the traditional methods (Wang, Liu and Li, 2021). Although vision SLAM has been widely used for mobile robot navigation, it still faces some challenges in practical applications. First, the robustness problem is particularly prominent. Under complex conditions such as light variations, dynamic environments, and occlusions, traditional vision SLAM systems tend to lose tracking ability or generate inaccurate maps. For example, low-light conditions or significant light changes can make it difficult to extract visual features, which in turn affects positioning accuracy. Therefore, optimizing the robustness of vision SLAM bears the brunt. In addition, there is a need to optimize the scalability of the vision SLAM. This is because the scalability of existing vision SLAM systems is limited as the size of the environment increases. Building maps for large-scale environments increases the computational load, especially on resource-limited devices, making it difficult to process or store large-scale environmental data in real-time.

The aim of this study is to propose a new framework for optimized visual SLAM with the aim of improving the robustness and scalability of SLAM systems. Deep learning and image processing will be combined, Matlab and python will be used to improve the robustness and scalability of visual SLAM.

2 METHOD

2.1 Data Sources

The data sources are mainly graphical. It is used for Convolutional Neural Network (CNN) feature extraction as well as ORB feature matching. The data source is mainly image based. CNN feature extraction as well as ORB feature matching uses a set of gate images as samples. The difference is that in CNN feature extraction, ResNet50 is used as the feature extraction network.

In Proximal Policy Optimization (PPO), instead of using an external real dataset, a simulation environment is constructed to generate the 'data' for training, i.e., the generation of environmental data. Firstly, a simple SLAM robot navigation environment is defined, and secondly, an observation space is defined, which is defined as a grey scale image of shape (64, 64, 1), i.e., a single-channel image of 64×64 pixels. Finally, the action space is defined and the action space is defined as discrete 3 actions representing forward, left turn and right turn respectively.

Next is the relevant dataset used for ORB loopback detection. Since loopback detection is used to check whether the robot has returned to the previously passed position, it need to process many frames of the image. However, it is difficult to obtain a large number of images of the same scene quickly, so the study thought of acquiring a large number of images of a fixed place by time-lapse photography. Time-lapse photography refers to the use of a camera to record changes in the same scene over time, such as recording the environment of a street from sunrise to sunset. The recorded video is then captured evenly over 50 frames. In this way the study get a large number of images for loopback detection.

2.2 Method

2.2.1 CNN

CNN is a deep learning model specifically designed for processing grid-structured data (e.g., images). CNN is one of the core technologies for computer vision tasks by performing feature extraction and classification through Convolutional Layers, Pooling Layers, and Fully Connected Layers (LeCun, Boser, Denker, Henderson, Howard and Jackel, 1989).

Firstly, CNN can perform local connectivity and weight sharing. CNN uses convolutional operations to extract features from local regions, and significantly reduces model parameters by sharing

weights, thus improving learning efficiency and generalization ability. Secondly, by stacking multiple layers of convolutional and pooling layers, CNNs are able to extract higher semantic features layer by layer from low-level features, which demonstrates a good capability of hierarchical modelling (Albawi, Mohammed and Al-Zawi, 2017). CNN can also introduce a certain degree of translation invariance through operations such as maximal pooling, which makes the model robust to slight deviations in the input data (Li, Zhang, Liu and Yang, 2021).

2.2.2 PPO

Besides CNN, the study can also use PPO to improve robustness. Proximal Policy Optimization (PPO) is a reinforcement learning algorithm. It belongs to the category of Policy Gradient Methods (PGMs), which are used to train an agent to perform optimal decision making in an environment (Schulman, Wolski, Dhariwal, Radford and Klimov, 2017). PPO is widely used in robot control, game AI, and autonomous driving by improving the PGMs and improving the sample utilization rate while ensuring the training stability.

2.2.3 ORB

ORB is an efficient feature extraction and matching algorithm for real-time applications (Rublee, Rabaud, Konolige and Bradski, 2011). It combines FAST (feature detection) and BRIEF (feature description) and is optimized for rotational invariance, making it particularly suitable for robot vision tasks such as SLAM.

2.2.4 Experimental Steps

First is CNN feature extraction. This paper loaded ResNet50 as a feature extraction network, which is a deep CNN with a depth of 50 layers capable of extracting high-level image features. In the SLAM task, using a pre-trained CNN network can extract more representative visual features, thus improving the accuracy of loopback detection. The two test images are then read and resized. These two images should be keyframes captured at some point in time during the SLAM process. And resize the images to be the size it needs for ResNet50. If the sizes don't match, it may cause the next feature extraction to fail. After that CNN features are extracted. The last fully connected layer of ResNet50 (fc1000) is selected as the feature output. Then calculate the cosine similarity of the two images and output the similarity result. Finally, the study set a similarity threshold for judging the results of loopback detection. If the similarity is greater than the similarity threshold, it means that the features of the two images are close and may be loopback keyframes. If the similarity is less than the similarity threshold, it means that the two images have large differences and do not belong to the loopback detection matching range.

This paper can use the PPO algorithm to train a robot to navigate in a visual SLAM environment.

First, this paper creates a simple SLAM robot navigation environment. This environment is divided into the robot's observation space and action space. The environment observed by the robot is a 64*64 grey scale image, which is the SLAM map. The robot's action space is divided into forward, left turn and backward. Consider the need to reset the environment when a new turn begins. So the study need to reset the environment. Then determine the robot's reward mechanism and termination conditions. The reward mechanism is divided into rewards and minor penalties. The termination condition can be set to terminate the action of the robot after performing 100 rounds. After that define the PPO policy network and train the PPO agent. Then there is a loop to train the robot for 100 rounds.

In addition to applying deep learning to improve robustness, the study can also improve it through simple image processing.

This paper use the ORB algorithm for feature matching to optimize robustness.

First the SLAM keyframes are read, that is, the two images are read. Then the images are converted into grey scale maps. This is because ORB feature extraction algorithms usually perform better on greyscale images. The greyscale image reduces the computational complexity while retaining the most critical visual information, making feature detection more stable. After that ORB is used for feature point detection. After that ORB key points are used to extract the feature descriptors. ORB feature descriptors are a type of binary descriptor (BRIEF) that can effectively represent the local image information of the feature points. ORB feature extraction is capable of generating a unique descriptor for each key point, which will be used to match the similar points in the subsequent steps. Feature matching is then performed. The core goal of feature matching is to find the same visual features between two frames, and these matching points can help the robot understand its relative position in the environment. The matching points are then counted and if there are too few matching points, the feature matching fails. Generally three is the minimum acceptable number of matching points. After that the

coordinates of the matching points are obtained and the results are visualized.

In addition to feature matching, the study can also use ORB to perform loopback detection on a series of images.

Loopback detection is a key aspect in vision SLAM, which is mainly used to detect whether the robot has returned to the previously passed location. Since SLAM systems accumulate errors over a long period of time, loopback detection can improve the consistency and accuracy of the map by identifying the same scene and correcting the position estimation errors. In applications such as robot navigation and autonomous driving, loopback detection plays a role in global optimization, enabling SLAM systems to construct more accurate and robust maps.

Firstly 50 images are read and these images are labelled with sequence from 1 to 50. Then it is converted into grey scale image and key point detection is performed and then the corresponding ORB features are extracted. After that loopback detection is performed, so-called loopback detection is to perform multiple feature matching. Greater than the set matching threshold means that the matching is successful, and the two images that are matched successfully are called matched pairs, also known as loopback frame pairs. Finally all the loopback pairs are output. So the essence of loopback detection by ORB is feature matching on multiple pairs of images.

3 ANALYSIS OF RESULTS

3.1 The Experiment About CNN Feature Extraction

Firstly, experiments about feature extraction with CNN algorithm and thus improving robustness are carried out by Matlab. Figure 1 shows two different views of a door. And the similarity threshold is set to 0.8. The final result is obtained with a similarity of 0.8324. Since this similarity is greater than 0.8, it may be a loopback keyframe.

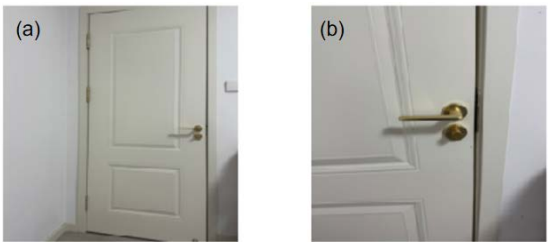

Figure 1: Two different views of a door (Photo credit : Original).

Figure 1(a) shows the full view of the door, while Figure 1(b) shows the door handle part of the door, the two figures have a high degree of similarity, it is a loopback keyframe.

3.2 The Experiment About PPO

Next are experiments to train the PPO model to improve robustness and scalability. This experiment was conducted on python. After 100 rounds of training, the model is ready to be applied to robot navigation on visual SLAM.

Figure 2: Training Reward Trend (Photo credit: Original).

Figure 2 shows a line graph on the number of training rounds and cumulative rewards. According to the Figure2, the study find that as the number of rounds increases, the cumulative reward also tends to rise, and the two are positively correlated.

3.3 The Experiment About ORB Figures Matching

The first is an experiment on feature matching using ORB, which is also performed on Matlab. The final number of matched points obtained is 54. The beginnings and ends of the yellow lines in the diagrams connect the red dots and green markers of the two diagrams, respectively, and the yellow lines represent places where the same door in both diagrams agrees. The study can find that the yellow lines in both figures are mainly densely concentrated near the door handles, which indicates that the features of the door handles in both figures are highly matched. This indicates that the door handles in the two figures are highly matching features.

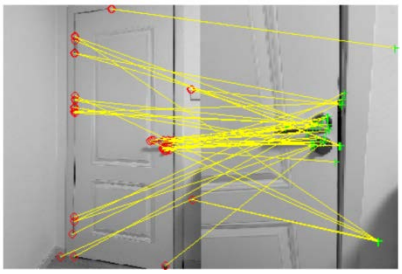

Figure 3: Feature matching (Photo credit: Original).

Figure 3 shows feature matching for the same door from different viewpoints. The yellow lines are mainly clustered near the door handles, so the door handles are highly similar features.

3.4 The Experiment About ORB Loopback Detection

Finally, there is an experiment on loopback detection using ORB. Again, this was performed on Matlab. Loopback detection is essentially performing feature extraction on a series of images multiple times. Since time-lapse photography of the same scene is used, many pairs of loopback frames are generated in these 50 images, and the table below shows only a portion of the pairs of loopback frames. Table 1 shows some of the loopback frame pairs.

Table 1: Partial loopback frame pairs.

Loopback frames	
1	2
1	3
1	5
2	3
2	4
3	4
3	5

4 CONCLUSION

In this paper, it optimize the robustness and scalability of visual SLAM for robot navigation by looking at both deep learning (CNN,PPO) and image processing techniques (ORB)

CNN can extract the similarity between two images through feature extraction, improving the visual perception ability during SLAM process. In addition, CNN can optimize loop detection and improve map accuracy. However, this feature extraction experiment also has defects: because the similarity threshold is an important basis about whether two images are loopback keyframes or not, but the similarity threshold in this experiment is set by us artificially, so how to set a reasonable similarity threshold is the difficult problem.

The paper provided a PPO model for improving robustness and trained it. It can optimize the decision-making strategy of robots in the SLAM process, improve path planning and exploration capabilities. However, it has not been applied to robots, so relevant practices are the focus of future research.

By using the ORB algorithm for keyframe matching and combining it with loop detection, real-time performance and accuracy can be improved, ensuring stable mapping and localization. But there is a flaw in the feature matching experiments about ORBs: the two places connected at the beginning and end of some of the yellow lines are not the same place in the same door in both diagrams, so this means that not all of the yellow lines are accurate. In practical applications, CNN can be combined for feature extraction, ORB for loop detection, PPO for decision optimization, and more efficient robot navigation systems can be constructed.

For the future, the paper can focus on applying these methods to real-life robot navigation. For example, applying the PPO model trained in this paper to real robot navigation. The significance of the research in this paper is to optimize the robustness and scalability of visual SLAM for robot navigation, which is finally reflected in real robot navigation applications.

REFERENCES

Albawi, S., Mohammed, T. A., Al-Zawi, S., 2017. Understanding of a convolutional neural network. In 2017 International Conference on Engineering and Technology (ICET), 1-6.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., Leonard, J. J., 2016. Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age. IEEE Transactions on Robotics, 32(6), 1309 – 1332.

Ghosh, S., Sanyal, S., Mukherjee, A., 2024. Obstacle Avoidance and Path Planning Methods for Autonomous Navigation of Mobile Robot. Sensors, 24(11), 3573.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Jackel, L. D., 1989. Backpropagation applied to handwritten zip code recognition. Neural Computation, 1(4), 541 – 551.

Li, Y., Zhang, J., Liu, W., Yang, J., 2021. A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Transactions on Neural Networks

and Learning Systems. DOI: 10.1109/TNNLS.2021.3075429.

Mur-Artal, R., Montiel, J. M. M., Tardos, J. D., 2015. ORB-SLAM: A versatile and accurate monocular SLAM system. IEEE Transactions on Robotics, 31(5), 1147 – 1163.

Mur-Artal, R., Tardós, J. D., 2017. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras. IEEE Transactions on Robotics, 33(5), 1255 – 1262.

Rublee, E., Rabaud, V., Konolige, K., Bradski, G., 2011. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2564 – 2571.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal Policy Optimization Algorithms. arXiv preprint, arXiv:1707.06347.

Wang, Y., Liu, Y., Li, X., 2021. Hybrid Navigation Method for Multiple Robots Facing Dynamic Obstacles. Tsinghua Science and Technology, 26(5), 674 – 691.

Zaffar, M., Ehsan, S., Stolkin, R., McDonald-Maier, K., 2018. Sensors, SLAM and Long-term Autonomy: A Review. arXiv preprint, arXiv:1807.01605..