SUGISAKA MASANORI

Journal of Robotics, Networking and Artificial Life Vol. 10(4); March (2024), pp. 323–327
ISSN (Online): 2352-6386; ISSN (Print): 2405-9021
https://alife-robotics.org/jrnal.html

Research Article

Research on Improving Accuracy of Dynamic Visual SLAM Detection

Yufei Liu, Kazuo Ishii

Department of Human Intelligence Systems, Kyushu Institute of Technology, 2-4 Hibikino, Wakamatsu-ku, Kitakyushu, 808-0196, Japan

ARTICLE INFO

Article History

Received 04 December 2023
Accepted 17 April 2024

Keywords

Visual SLAM
Deep learning
Feature detection
Position estimation
YOLOv5

ABSTRACT

Currently, the application of SLAM systems in the field of navigation is quite active. However, the detection accuracy of classical SLAM systems when applied in dynamic environments is not high. This is because some dynamic objects in dynamic environments may occlude the environmental features that should have been extracted by the SLAM system. In order to solve this problem, my research conducts analysis on the detection process in dynamic environments and designs a solution: to identify and remove dynamic objects in dynamic environments to improve detection accuracy. In this research, an object detection algorithm YOLOv5 is first used to detect, identify, and remove dynamic objects in the environment extracted by the SLAM system. Then, the remaining static environmental features are passed into visual SLAM for conventional SLAM environment calculation and localization work. Finally, the modified integrated algorithm is validated and analyzed on the TUM dataset for feasibility. The results indicate that this approach successfully removes dynamic objects and effectively improves the robustness of visual SLAM applications in dynamic environments.

© 2022 *The Author*. Published by Sugisaka Masanori at ALife Robotics Corporation Ltd.
This is an open access article distributed under the CC BY-NC 4.0 license
(http://creativecommons.org/licenses/by-nc/4.0/).

1. Introduction

Recently, the robotics industry is developing rapidly and gradually being applied in various fields, such as factory handling, mine inspection, or delivery scenarios in daily life. Visual SLAM (Simultaneous Localization and Mapping) is one of the important technologies in this regard. It enables autonomous localization and map building for robots, vehicles, or other mobile devices in environments without GPS signals. Visual SLAM primarily relies on sensors such as cameras to acquire environmental information. Compared to additional sensors like GPS or LiDAR, visual sensors are more common and cost-effective. Visual SLAM is applicable to various environments, including indoor, outdoor, structured, or unstructured environments, and can generate high-precision maps accordingly. This also makes it a research hotspot in the field of mobile robotics. For developers, there is a rich variety of open-source libraries and algorithms for visual SLAM technology, allowing for customization and extension according to specific requirements to meet the needs of different application scenarios. The YOLO (You Only Look Once) series is a popular real-time object detection algorithm. It directly predicts the entire image using a single neural network model, without the need for complex sliding windows or region proposal generation. Therefore, it has high real-time performance and can complete object detection tasks in a short time. Additionally, the YOLO algorithm is designed to be simple, with a small number of parameters and a high degree of model lightweight, making it suitable for deployment and application on embedded devices and mobile platforms, enabling real-time object detection tasks. Combining the two can satisfy real-time and efficient detection requirements.

Corresponding author E-mail: liu.yufei124@mail.kyutech.jp, ishii@brain.kyutech.ac.jp URL: www.kyutech.ac.jp

2. Problem Description

2.1. *Overview of visual SLAM technology*

Because this article is based on the visual SLAM algorithms, it first introduces the relevant research trends of visual SLAM. At present, some mainstream visual slam technologies include VINS Mono, RTABMap, PTAM, LSD-SLAM, DSO, and ORBSLAM series [1]. The entire working process of visual SLAM can roughly include the following parts. First, environmental information is captured in the form of images through a camera and fed into the system, forming a video stream in chronological order. Then, the system extracts feature from each image in the video stream and matches features between adjacent frames, calculating the camera's motion through minimizing pixel intensity values. The methods for feature matching mainly include optical flow and direct methods. Optical flow identifies image feature points and estimates camera motion through triangulation or epipolar geometry. Direct methods utilize pixel blocks or extract image corners directly, calculating motion estimation based on grayscale values.

After motion estimation, the data undergoes noise filtering to achieve an optimal pose estimation. Subsequently, the global map is estimated using maximum a posteriori probability. In 2020, Campos et al. [2], introduced ORB-SLAM3, which builds upon the research conducted by Artal et al., and has emerged as a prominent feature-based SLAM system in the field.

2.2. *Impact of Dynamic Objects on Visual SLAM*

ORB-SLAM3 is a famous algorithm in visual SLAM, primarily detecting the environment through orb feature extraction. Its structure consists of two main parts: the front-end and the back-end. The front-end mainly processes the environmental image data captured by the sensor through feature extraction and matching. It solves the epipolar geometry relationship between corresponding feature pixels to estimate camera translation and rotation parameters. On the other hand, the back-end involves nonlinear optimization, mapping estimation, and loop closure detection. Unlike previous methods using Kalman filtering, nonlinear optimization uses bundle adjustment (BA) to simultaneously optimize the six degrees of freedom of camera pose parameters and landmark pose in space.

However, the detection accuracy of ORB-SLAM3 in dynamic environments is currently not high. This is because the BA method mentioned earlier in the back-end performs poorly in dynamic environments. When optimizing camera poses using the BA method, the detection accuracy depends on whether the feature points extracted from the environment detection images are static features. In practice, in dynamic scenes, it is worth discussing how feature-based SLAM algorithms and direct methods SLAM algorithms distinguish between dynamic features and static features. Mismatching dynamic features as static features may cause misalignment, and misalignment in the front-end will affect the accuracy of pose estimation in the back-end. Ultimately, this will result in a significant discrepancy between the system's estimated pose and the actual environment.

3. Elimination of the dynamic points

3.1. *Dynamic SLAM based on geometric methods*

There are two main methods to reduce the impact of dynamic features in visual SLAM in dynamic scenes based on geometric approaches. One method, based on traditional geometry, is proposed by Sun et al. [3] detect moving objects by comparing differences between consecutive frames; however, this technique suffers from limited real-time performance. Wang et al. [4] , utilizes Epipolar Geometry to filter matching feature points between adjacent frames and combines depth information obtained from RGB-D cameras to identify independent dynamic objects in the scene through clustering. However, this method requires the use of transformation matrices for pose estimation between adjacent frames, leading to a significant decrease in detection accuracy when multiple dynamic objects need to be detected in the environment. Another method proposed by Lin et al. [5] combines image depth information with visual ranging to detect the positions of moving objects in the scene. However, due to the uncertainty associated with depth information and the accumulation of errors when computing transformation matrices between consecutive frames, this method also compromises accuracy.

The fundamental principle of these methods mentioned above is the calculation of dynamic features. Features based on dynamic objects deviate from the standard constraints observed in static scenes through triangulation, fundamental matrix estimation, epipolar line determination, and reprojection error analysis. During pose estimation, these dynamic features are treated as outliers. The correctness of feature matching depends on whether the extracted feature points violate the aforementioned constraints, thereby appropriately excluding dynamic points. However, it should be noted that the accuracy of this method largely depends on the proportion of static feature points present in the given scene.

3.2. *Dynamic SLAM based on deep learning*

Next, we will introduce another direction to solve this problem, which is to use deep learning to improve the detection accuracy of visual slam systems in dynamic environments. There are already some methods that use deep learning to remove dynamic features. Berta Bescos et al. [6] proposed the DynaSLAM algorithm, which leverages Mask R-CNN (Region-based Convolutional Neural Network) to optimize ORB-SLAM2. By combining deep learning with geometry, DynaSLAM effectively filters out dynamic feature points in the scene. While this algorithm has demonstrated impressive performance on the TUM dataset, its reliance on Mask R-CNN for pixel segmentation hampers real-time detection efficiency, limiting its applicability in real-world environments. Another approach called DDL-SLAM (Dynamic Deep Learning SLAM) [7] employs DUNet (Deformable Unity Networking) and semantic masks obtained through multi-view geometry to detect dynamic objects and restore the obscured background using an image restoration strategy. However, due to pixel-level mask calculation, this method also falls short of achieving real-time performance. In contrast to Mask R-CNN, YOLOv5 (You Only Look Once Version 5) [8]. offers a more efficient object detection model that achieves detection speeds ranging from 45-155 frames per second (fps), surpassing Mask R-CNN's maximum speed of 5 fps by a factor of 9-30 times. Integrating YOLOv5's object detection results into dynamic visual SLAM algorithms could partially compensate for the low efficiency associated with Mask R-CNN usage.

This study also follows the aforementioned research approach by integrating deep learning algorithms into the object detection thread of the ORB-SLAM3 algorithm. YOLOv5 object detection algorithm was selected for dynamic object recognition in dynamic environments. Additionally, a new module was added to the tracking thread of ORB-SLAM, which removes dynamic features that may affect recognition while sensors extract environmental information. Consequently, the backend can perform motion estimation without interference from dynamic objects, thereby improving the detection accuracy of ORB-SLAM3 in dynamic environments. The integrated algorithm flowchart is shown in Fig.1.

4. Experimental verification and data analysis

4.1. *Experimental construction and data set*

My experiment involved testing the optimized visual SLAM system proposed in this paper using video sequences from the TUM dataset. The experimental results were analyzed to evaluate the localization accuracy of the SLAM system. The experiments were conducted on the Ubuntu 20.04 operating system, with a 12th Gen Intel(R) Core (TM) i9-12900H 2.50GHz CPU, an NVIDIA GeForce RTX 3060 GPU with 12GB of VRAM, and the PyTorch deep learning framework. The algorithm's performance was tested on the fr3_walking_xyz, fr3_walking_half, and fr3_walking_static dataset sequences. The fr3_walking_xyz dataset depicts two individuals walking and conversing in a fixed scene, with both the camera and people in motion, representing a high-dynamic scene. The fr3_walking_half dataset builds upon this by having the camera move along a semi-circular trajectory in the air. The fr3_walking_static dataset, on the other hand, features relatively stationary objects, representing a low-dynamic scene.

4.2. *Visual SLAM front-end feature extraction effect after integrating YOLOv5*

After completing the algorithm integration, I compared the modified algorithm with the original one. The specific visualization results of the detection operation are shown in Fig. 2. From the figure, it is evident that integrating the object detection algorithm into the frontend of visual SLAM has achieved real-time and accurate detection of dynamic objects. Additionally, after removing some dynamic feature points, the system can detect more useful static feature points, which can better assist backend computation and improve detection accuracy.

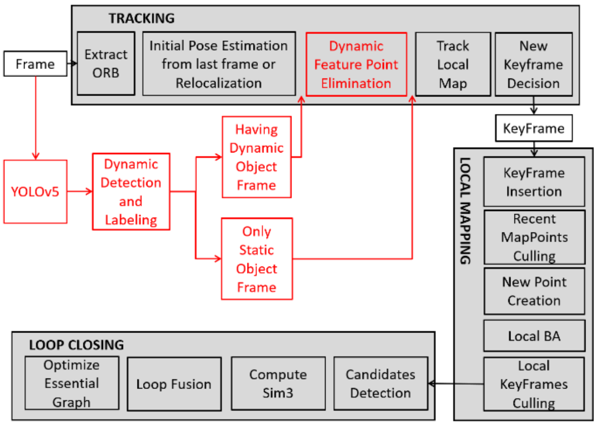


Fig.1 Algorithm framework

4.3. *Experimental data indicator analysis*

The evaluation methods of SLAM systems can be summarized into two common parameters: Absolute Trajectory Error (ATE) and Relative Pose Error (RPE). ATE assesses the difference between the true trajectory and the estimated trajectory, while RPE calculates the pose differences within the same time interval, typically used for odometer error estimation. Afterward, both parameters need to be calculated for Root Mean Square Error (RMSE) to determine the overall error value. It can be observed from the definitions of these parameters that a lower RMSE value indicates a smaller error, implying a closer approximation to the true situation, thus reflecting the algorithm's stronger robustness and stability [9].

Following experiments conducted on different datasets, evaluations of various processes were computed. The comparison between the original algorithm and the integrated algorithm in terms of parameters is illustrated in Table 1 and Table 2.

Table.1 ATE performance comparison

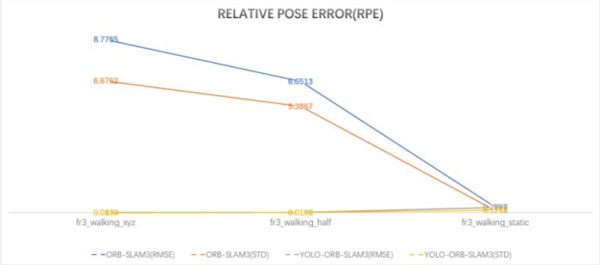
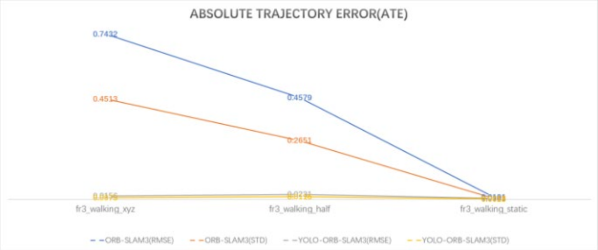
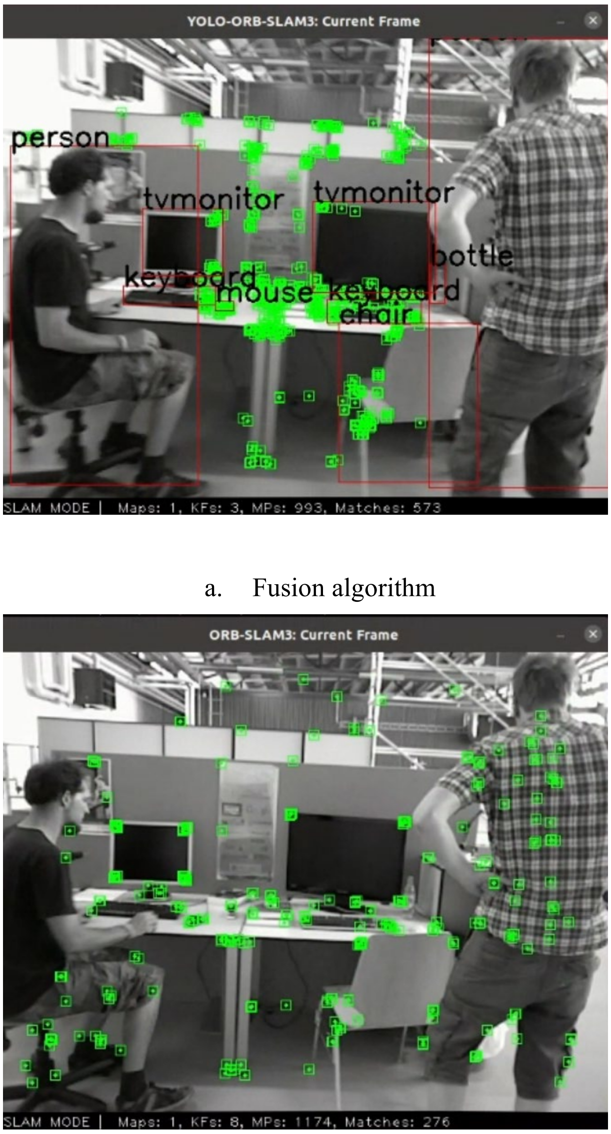


Table.2 RPE performance comparison



In the tables, the "xyz" and "half" datasets represent the parameter results in dynamic environments, while the "static" dataset represents the parameter results in static environments. It can be distinctly observed from the parameter comparison in the tables that the integrated algorithm proposed in this study not only maintains the detection accuracy of the original algorithm in static environments but also significantly improves the detection accuracy of visual SLAM systems in dynamic environments.



SLAM MODE | Maps: 1, KFs: 3, MPs: 993, Matches: 573

a. Fusion algorithm

SLAM MODE | Maps: 1, KFs: 8, MPs: 1174, Matches: 276

b. Original algorithm

Fig.2. Comparison of front-end feature extraction results before and after integrating YOLOv5

References

1. Zhaopeng G, Liu H, University P, et al. A survey of monocular simultaneous localization and mapping[J]. CAAI Transactions on Intelligent Systems, 2015.
2. Campos C, Elvira R, Rodríguez J J G, et al. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam [J]. IEEE Transactions on Robotics, 2021, 37(6): 1874-1890.
3. Sun Y, Liu M, Meng M Q H. Improving RGB-D SLAM in dynamic environments: A motion removal approach[J]. Robotics and Autonomous Systems, 2017, 89: 110-122.
4. Wang R, Wan W, Wang Y, et al. A new RGB-D SLAM method with moving object detection for dynamic indoor scenes[J]. Remote Sensing, 2019, 11(10):1143.
5. Lin S F, Huang S H. Moving object detection from a moving stereo camera via depth information and visual odometry[C]//2018 IEEE International Conference on Applied System Invention (ICASI). IEEE, 2018:437-440.
6. Bescos B, Fácil J M, Civera J, et al. DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes[J]. IEEE Robotics and Automation Letters, 2018, 3(4) : 4076-4083.
7. Ai Y, Rui T, Lu M, et al. DDL-SLAM: A robust RGB-D SLAM in dynamic environments combined with deep learning[J]. IEEE Access, 2020, 8: 162335-162342.
8. Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016:779-788.
9. Chang Z, Wu H, Sun Y, et al. RGB-D Visual SLAM Based on Yolov4-Tiny in Indoor Dynamic Environment[J]. Micromachines, 2022, 13(2):230.

Authors Introduction

Ms. Yufei Liu

She graduated from Zhejiang Normal University in China with a bachelor's degree in 2017. Now she is studying as a master's student at Kyushu Institute of Technology in Japan.

Dr. Kazuo Ishii

He received his PhD from the University of Tokyo, Japan, in 1996. In 2011, he joined Kyushu Institute of Technology and is currently a professor in the Department of Human Intelligent Systems. His research interests include information communications and marine robotics. He is a member of IEEE.