

THIẾT KẾ SLAM ĐỊA HÌNH 3D CHO ROBOT DI ĐỘNG DI CHUYỂN TRONG MÔI TRƯỜNG KÍN

DESIGNING 3D TERRAIN SLAM FOR MOBILE ROBOTS TO MOVE IN A CLOSED ENVIRONMENT

Trần Trọng Tiến¹, Trần Bá Hiến^{2,*}, Đoàn Ngọc Mạnh²,
Sái Thanh Huyền³, Đoàn Quang Khởi³, Trương Thị Bích Liên⁴

TÓM TẮT

Bài báo trình bày về xây dựng bản đồ địa hình 3D cho robot tự hành hoạt động trong môi trường trong nhà dựa trên hệ điều hành lập trình cho robot (Robot Operating System - ROS). Phần cứng là một robot Omni 4 bánh với nền tảng máy tính nhúng hiệu suất cao Jetson-Tx2, camera 3D và một cảm biến Lidar để thu thập dữ liệu từ môi trường bên ngoài. Kết hợp với việc chạy mô phỏng robot trong môi trường trong nhà sử dụng Gazebo và thử nghiệm trên Rviz cho thấy sự tiềm năng, hiệu quả của hướng nghiên cứu sử dụng hệ điều hành robot ROS trong việc lập bản đồ môi trường cho robot tự hành.

Từ khóa: RTAB_Map, SLAM2D, SLAM3D, ROS, Gazebo, Rviz, SLAM, Omni robot.

ABSTRACT

This paper presents the construction of 3D terrain maps for self-propelled robots operating in indoor environments based on a programming operating system for robots (Robot Operating System - ROS). Hardware is a 4-wheel Omni robot with Jetson-Tx2 high performance embedded computer platform, 3D camera and a Lidar sensor to collect data from the external environment. Combined with running robot simulations in an indoor environment using Gazebo and testing on Rviz, it shows the potential and effectiveness of the research direction using ROS robot operating system in mapping the environment for autonomous robots.

Keywords: RTAB_Map, SLAM2D, SLAM3D, ROS, Gazebo, Rviz, SLAM, Omni robot.

¹Lớp Điện tử Truyền thông 01 - K15, Khoa Điện tử, Trường Đại học Công nghiệp Hà Nội

²Lớp Điện tử Truyền thông 04 - K13, Khoa Điện tử, Trường Đại học Công nghiệp Hà Nội

³Lớp Điện tử Truyền thông 02 - K13, Khoa Điện tử, Trường Đại học Công nghiệp Hà Nội

⁴Khoa Điện tử, Trường Đại học Công nghiệp Hà Nội

*Email: hienbn3333@gmail.com

1. GIỚI THIỆU

Ngày nay, robot di động được sử dụng rộng rãi trong các hoạt động, tự động di chuyển trong các môi trường không cố định và không cần sự giám sát của con người. Hoạt động tự trị của robot trong môi trường chưa được biết đến đòi hỏi robot phải tự nhận biết được môi trường xung

quanh, xây dựng bản đồ, định vị và lập kế hoạch đường đi và tránh các vật cản tĩnh và động trong quá trình di chuyển.

Xây dựng bản đồ và định vị đồng thời hay còn gọi là SLAM (Simultaneous Localization and Mapping) là vấn đề tính toán của việc xây dựng hoặc cập nhật bản đồ của một môi trường không xác định đối thời theo dõi vị trí của các tác nhân bên trong bản đồ đó. Những năm gần đây, SLAM đã là chủ đề của nghiên cứu kỹ thuật được quan tâm. Nhưng với những cải tiến lớn về tốc độ xử lý của máy tính và sự sẵn có của các cảm biến giá rẻ như máy ảnh và laser, SLAM hiện được sử dụng cho các ứng dụng thực tế trong một số lĩnh vực ngày càng tăng. Phương pháp này thu thập dữ liệu từ các cảm biến để tái tạo môi trường hoạt động thông qua việc đưa thông tin môi trường vào trong một bản đồ 2D hoặc 3D. Cảm biến được sử dụng trong SLAM được chia thành hai loại: cảm biến ngoại vi (thu nhận dữ liệu từ môi trường bên ngoài) và cảm biến nội vi (xác định sự thay đổi vị trí, hướng, gia tốc,...). Đã có nhiều công trình thực hiện SLAM 2D là phương pháp tạo ra bản đồ 2D và phát hiện các chướng ngại vật xung quanh trong môi trường không xác định và cũng đã được các tác giả xây dựng và thực thi trên nền ROS. Để tăng độ chính xác, SLAM thường kết hợp các dữ liệu từ nhiều cảm biến qua các phương pháp xác suất như Markov, Kalman,... Tuy nhiên việc sử dụng nhiều cảm biến sẽ làm tăng độ phức tạp, chi phí và thời gian xử lý của hệ thống. Đặc biệt là robot tự hành hiện nay không chỉ giới hạn trong việc di chuyển, mà nó còn được tích hợp các cơ cấu chấp hành, cánh tay máy robot...vv để thực hiện các nhiệm vụ cụ thể nào đó trong môi trường không gian hoạt động, do đó SLAM không chỉ ý nghĩa trong việc xây dựng bản đồ 2D môi trường hoạt động phục vụ điều hướng cho robot, mà còn cần thiết có các bản đồ 3D (SLAM3D) nhằm phục vụ các bài toán tương tác khác của robot trong môi trường hoạt động đó.

Hiện nay với sự phát triển của lĩnh vực thị giác máy tính nên các hệ thống SLAM thường sử dụng camera để thu thập dữ liệu từ môi trường bên ngoài của những tác nhân gần xung quanh và kết hợp với Lidar để xác định vị trí của các tác nhân xa. Cùng với xu hướng sử dụng hệ điều hành robot - ROS (Robot Operating System) thì phương pháp SLAM cũng được phát triển hiệu quả. Các phương pháp SLAM sử dụng cảm biến trên nền tảng ROS phổ biến hiện

nay như Visual SLAM. Một số phương pháp của Visual SLAM như maplab, ORB-SLAM2, DVO-SLAM, MCPTAM, RTAB-Map, RGBDSLAMv2.... Trong các phương pháp của Visual SLAM thì RTAB-Map tương đối toàn diện khi có thể cung cấp bản đồ dạng lưới 2D (Occupancy Grid) như cách sử dụng cảm biến thông thường hay bản đồ 3D (Octomap). RTAB-Map được phân phối dưới dạng một ROS package có khả năng xử lý thời gian thực, tối ưu hóa việc định vị và tạo bản đồ thực tế.

2. CƠ SỞ LÝ THUYẾT

2.1. Bản đồ hóa và định vị đồng thời SLAM cho Robot

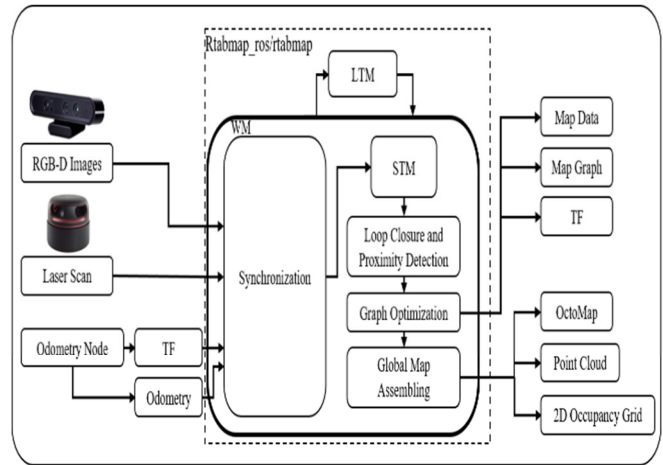
Một Robot có thể tự động di chuyển trong một môi trường không xác định thì việc xác định cho Robot biết môi trường xung quanh, vị trí của Robot trong môi trường đó là rất cần thiết.

Xây dựng bản đồ và định vị đồng thời hay còn gọi là SLAM (Simultaneous Localization and Mapping) có nghĩa là khám phá và lập bản đồ môi trường không xác định trong khi ước tính tư thế của chính robot bằng cách sử dụng các cảm biến được gắn trên robot. Đây là công nghệ quan trọng để điều hướng như lái xe tự động. Những năm gần đây, SLAM đã là chủ đề của nghiên cứu kỹ thuật được quan tâm. Nhưng với những cải tiến lớn về tốc độ xử lý của máy tính và sự sẵn có của các cảm biến giá rẻ như máy ảnh và laser, SLAM hiện được sử dụng cho các ứng dụng thực tế trong một số lĩnh vực ngày càng tăng. Phương pháp này thu thập dữ liệu từ các cảm biến để tái tạo môi trường hoạt động thông qua việc đưa thông tin môi trường vào trong một bản đồ 2D hoặc 3D. Cảm biến được sử dụng trong SLAM được chia thành hai loại: cảm biến ngoại vi (thu nhận dữ liệu từ môi trường bên ngoài) và cảm biến nội vi (xác định sự thay đổi vị trí, hướng, gia tốc,...). Bộ mã hóa và đơn vị đo lường quán tính (IMU) thường được sử dụng để ước lượng tư thế của robot. Bộ encoder tính toán tư thế gắn đúng của robot với tính toán chết để đo lường quay của bánh xe. Quá trình này đi kèm với khá nhiều lỗi ước lượng và thông tin quán tính được đo bởi cảm biến quán tính bù cho sai số của tư thế được tính toán. Để tăng độ chính xác, SLAM thường kết hợp các dữ liệu từ nhiều cảm biến qua các phương pháp xác suất như Markov, Kalman,... [4, 5]. Tuy nhiên việc sử dụng nhiều cảm biến sẽ làm tăng độ phức tạp, chi phí và thời gian xử lý của hệ thống.

2.2. Cấu trúc RTAB_Map [12, 14]

RTAB-Map là một phương pháp SLAM dựa trên đồ thị RGB-D sử dụng bộ đóng vòng lặp. Bộ đóng vòng lặp sử dụng phương pháp tiếp cận từ nhiều điểm để xác định khả năng một hình ảnh mới đến từ một vị trí trước đó hay vị trí mới. RTAB-Map phát hiện lỗi sai bằng cách sử dụng phương pháp GoodFeaturesToTrack (GFTT) theo mặc định, giúp giảm bớt việc điều chỉnh tham số, cho phép các tính năng được phát hiện đồng nhất trên các kích thước hình ảnh và cường độ ánh sáng khác nhau. Ngoài ra, RTAB-Map hỗ trợ tất cả các loại tính năng có sẵn trong OpenCV, chẳng hạn như SIFT, ORB hoặc BRIEF. Một trình tối ưu hóa đồ thị giảm thiểu các lỗi trong bản đồ khi các ràng buộc mới được

thêm vào và một phương pháp quản lý bộ nhớ hiệu quả được sử dụng để thực hiện các ràng buộc thời gian thực trong các môi trường lớn.



Hình 1. Sơ đồ khối rtab_map

Hình 1 mô tả sơ đồ khối rtabmap-ros. Cấu trúc của rtabmap_ros là một biểu đồ với các nút liên kết với nhau. Sau khi đồng bộ hóa các dữ liệu đầu vào, modul bộ nhớ ngắn hạn (STM) tạo một nút để ghi nhớ tư thế Odometry, dữ liệu cảm biến và các thông tin bổ sung cho các modul tiếp theo.

Các đầu vào yêu cầu là: TF để xác định vị trí của các cảm biến liên quan đến chân đế robot, Odometry, hình ảnh từ camera và đầu quét laser từ Lidar. Tất cả các thông báo từ đầu vào này sau đó được đồng bộ hóa và chuyển đến thuật toán đồ thị (SLAM). Kết quả đầu ra là dữ liệu bản đồ chứa trong Map Data được bổ sung mới nhất với các dữ liệu cảm biến nén và biểu đồ, Map Graph không có bất kỳ dữ liệu nào, hiệu chỉnh Odometry được xuất bản trên TF, một tùy chọn lưới chiếm dụng 3D (OctoMap), Point Cloud và một lưới chiếm dụng 2D (2D Occupancy Grid).

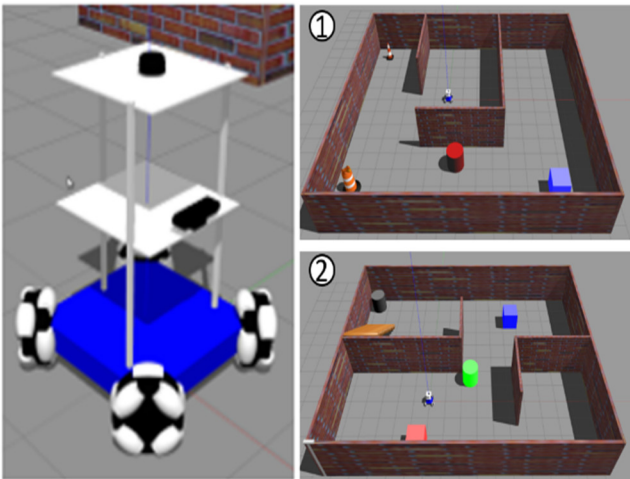
Phương pháp RtabMap chạy trên module quản lý đồ thị. Nó sử dụng để giới hạn kích thước của biểu đồ để có thể đạt được SLAM trực tuyến lâu dài trong môi trường lớn. Nếu không có bộ quản lý bộ nhớ, khi bản đồ phát triển, thời gian xử lý của các module như đóng vòng lặp, phát hiện vùng lân cận, tối ưu hóa đồ thị và lắp ráp bản đồ toàn cầu cuối cùng có thể vượt qua thời gian ràng buộc với thời gian thực, tức là thời gian xử lý trở nên quá lớn. Về cơ bản, bộ nhớ của RtabMap được chia thành bộ nhớ làm việc (WM) và bộ nhớ dài hạn (LTM), khi một nút chuyển sang LTM, nó không còn khả dụng cho các modul bên trong WM. Khi thời gian cập nhật RtabMap vượt quá ngưỡng thời gian cho phép, một số nút trong WM sẽ chuyển sang LTM để giới hạn kích thước của WM và giảm thời gian cập nhật. STM dùng để xác định xem nút nào cần chuyển sang LTM.

3. KẾT QUẢ MÔ PHỎNG VÀ THỰC NGHIỆM

Mô hình Robot được xây dựng bằng gói URDF. Bánh xe omni được thiết kế với một số con lăn được bố trí xung quanh chu vi bánh xe. RPlidar được đặt trên đỉnh của Robot

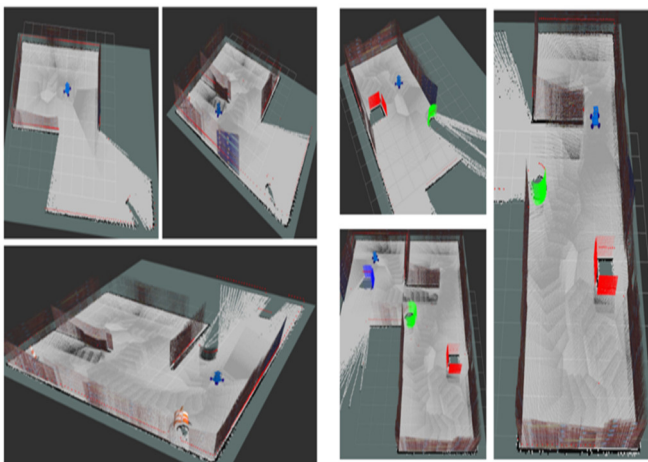
thực hiện nhiệm vụ SLAM để xây dựng bản đồ và quét vật cản tầm cao, khoảng cách xa. Astra Camera được đặt ở tầng trung, giúp quét vật cản tầm trung và tầm thấp phía trước Robot [1-3, 13]. Hình 2, với các tham số mô phỏng và thực nghiệm giống nhau:

- Tốc độ lớn nhất theo phương x và y: 1,5m/s
- Tốc độ góc lớn nhất: 0,5rad/s
- Bán kính robot: 0,25m
- Bán kính bánh xe: 0,07m
- Tham số cho cảm biến lidar: Phạm vi quét lớn nhất: 0,2 ÷ 10m, Độ phân giải: 1°, Góc quét: 360°.



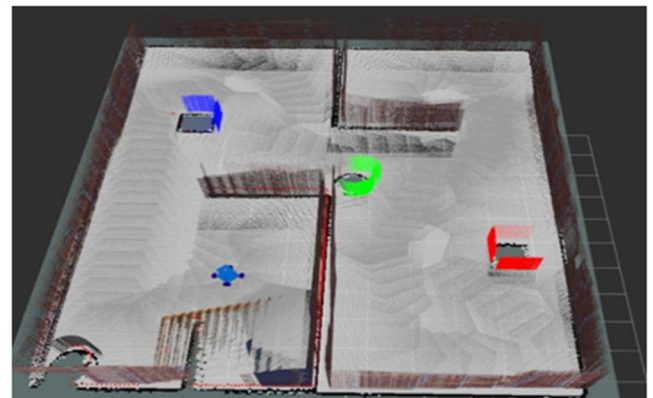
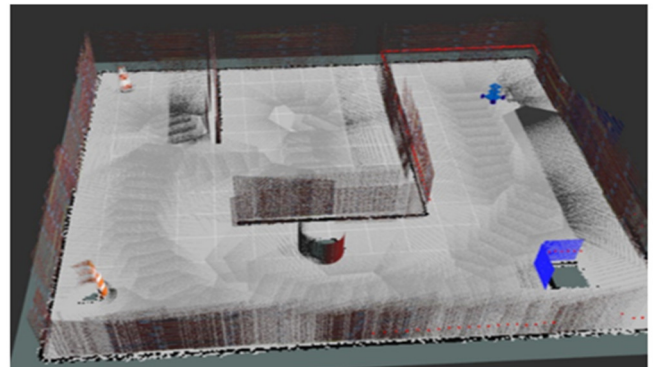
Hình 2. Mô hình 3D robot Omni và môi trường mô phỏng

Việc chạy mô phỏng thử nghiệm SLAM được thực hiện trên Rviz, là công cụ trực quan của ROS. Mục đích chính là hiển thị các thông báo thu được ở chế độ 3D, cho phép người dùng xác minh trực quan dữ liệu. Thông qua phần mềm Rviz, người dùng có thể giám sát được môi trường xung quanh của Robot theo thời gian. Hình 3 là kết quả thu được khi bắt khởi chạy kỹ thuật SLAM trong hệ thống nhận biết robot. Các đám mây điểm ảnh được quét từ camera được dựng lên với độ cao và màu sắc tương đồng với các vật thể được tạo ra trong môi trường trong nhà.



Hình 3. Quá trình SLAM

Môi trường thực nghiệm gồm các vách tường ngăn và các vật cản được sắp xếp ở các vị trí ngẫu nhiên trên bản đồ. Robot di chuyển xung quanh phòng, sử dụng camera để thu lại hình ảnh trong quá trình di chuyển, từ đó tái tạo lại bản đồ của môi trường xung quanh. Bản đồ 3D ở hình 4 là các pointcloud hay còn gọi là đám mây điểm được thu thập trong quá trình robot chuyển động. Dữ liệu đám mây này biểu thị hình ảnh của một vật dưới dạng nhiều điểm trong không gian tọa độ 3 chiều. Do kích thước của robot nhỏ, nên thị trường hoạt động của camera còn thấp nhưng vẫn thu được ảnh ở khoảng cách xa. Kết quả cho thấy robot có khả năng tái tạo lại bản đồ một cách hiệu quả.



Hình 4. Kết quả quá trình SLAM3D

4. KẾT LUẬN

Bài báo trình bày về lập bản đồ hóa (SLAM 3D) cho robot tự hành hoạt động trong các môi trường trong nhà dựa trên hệ điều hành lập trình cho robot ROS. Các kết quả cho thấy, robot có khả năng thu thập dữ liệu từ môi trường xung quanh, xây dựng bản đồ 3D và định vị vị trí của robot trên bản đồ của môi trường trong nhà. Các kết quả này là nền tảng cho các bước điều hướng, lập quỹ đạo chuyển động cho robot phục vụ các bài toán cụ thể như robot tự hành hoạt động trong các nhà máy, ứng dụng trong công việc vận chuyển hàng hóa trong nhà, Logictis... Đặc biệt bên cạnh đó, bản đồ 3D có ý nghĩa với các robot tự hành có tích hợp các cơ cấu chấp hành, cánh tay máy robot... ngoài việc di chuyển tự trị còn có thể thực hiện các nhiệm vụ tương tác cụ thể trong môi trường không gian hoạt động.

TÀI LIỆU THAM KHẢO

- [1]. Nguyễn Thị Duyên, Ngô Mạnh Tiến, Hà Thị Kim Duyên, Bùi Quang Tuấn, Trần Bá Hiến, Phạm Ngọc Minh, 2021. *Lập quỹ đạo cục bộ dựa trên bản đồ định vị đồng thời SLAM cho Robot tự hành trong nhà kính nông nghiệp trên nền tảng hệ điều hành ROS*. Hội thảo quốc gia lần thứ XXIV: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông - Thái Nguyên.
- [2]. Ngô Mạnh Tiến, Nguyễn Mạnh Cường, Hà Thị Kim Duyên, Bùi Quang Tuấn, Trần Bá Hiến, Nguyễn Minh Đông, Đỗ Quang Hiệp, 2021. *Xây dựng hệ thống bản đồ hóa SLAM và ứng dụng điều hướng cho Robot đa hướng sử dụng bánh xe Mecanum dựa trên hệ điều hành ROS*. Hội thảo quốc gia lần thứ XXIV: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông - Thái Nguyên.
- [3]. Nguyễn Thị Duyên, Ngô Mạnh Tiến, Hà Thị Kim Duyên, Bùi Quang Tuấn, Trần Bá Hiến, Nguyễn Minh Đông, Đỗ Quang Hiệp, 2021. *Xây dựng hệ điều hướng trên bản đồ, định vị SLAM cho Robot tự hành trong nhà kính nông nghiệp dựa trên hệ điều hành ROS*. Hội nghị - Triển lãm quốc tế lần thứ 4 về Điều khiển và Tự động hoá (VCCA).
- [4]. L. Xie, C. Scheifele, W. Xu, K. A. Stol, 2015. *Heavy-duty omni-directional Mecanum-wheeled Robot for autonomous navigation: System development and simulation realization*. IEEE International Conference on Mechatronics (ICM), pp. 256-261.
- [5]. M. Galli, R. Barber, S. Garrido, L. Moreno, 2017. *Path planning using Matlab-ROS integration applied to mobile Robots*. IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), pp. 98-103.
- [6]. M. Quigley, et al., 2009. *ROS: an open-source Robot Operating System*. ICRA workshop on open source software, vol. 3, p. 5.
- [7]. I. Zamora, N. G. Lopez, V. M. Vilches, A. H. Cordero, 2016. *Extending the openai gym for Robotics: a toolkit for reinforcement learning using ros and gazebo*. arXiv preprint arXiv:1608.05742.
- [8]. S. Park, G. Lee, 2017. *Mapping and localization of cooperative Robots by ROS and SLAM in unknown working area*. 56 th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), pp. 858-861.
- [9]. B. M. da Silva, R. S. Xavier, L. M. Gonçalves, 2019. *Mapping and Navigation for Indoor Robots under ROS: An Experimental Analysis*. Creative Commons CC BY license.
- [10]. Q. Lin, et al, 2017. *Indoor mapping using gmapping on embedded system*. IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 2444-2449.
- [11]. L. Zhi, M. Xuesong, 2018. *Navigation and Control System of Mobile Robot Based on ROS*. 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 368-372.
- [12]. Trần Bá Hiến, Ngô Mạnh Tiến, Hà Thị Kim Duyên, Ngô Mạnh Long, 2021. *Ứng dụng RTAB-Map xây dựng bản đồ 3D cho Robot đa hướng bốn bánh dựa trên hệ điều hành ROS*. Hội nghị Quốc gia lần thứ XXIV về Điện tử, Truyền thông và Công nghệ Thông tin (REV - ECIT 2021).
- [13]. Trần Bá Hiến, Bùi Thị Thu Hà, Hà Thị Kim Duyên, Trương Thị Bích Liên, TS. Ngô Mạnh Tiến, 2022. *Xây dựng hệ thống điều khiển cho Robot tự hành di chuyển trong nhà ứng dụng hệ điều hành ROS*. Tạp chí Khoa học và Công nghệ, trường đại học Công nghiệp Hà Nội.
- [14]. Labbé, Mathieu, Francois Michaud. *RTAB-Map as an Open-Source Lidar and Visual SLAM Library for Large-Scale and Long-Term Online Operation*. Interdisciplinary Institute of Technological Innovation (3IT).