2026 IEEE/SICE International Symposium on System Integration (SII)
January 11-14, 2026. Cancún, México

Robust Localization using Map Selection from Multiple Maps Based on the Relative Pose Difference of Map and Interoceptive Sensor

Takumi Suzuki[1], Yuki Funabora, Shinji Doki, Kae Doki

Abstract— **Autonomous mobile robots are expected to demonstrate a high degree of adaptability, enabling effective operation across diverse environments. Robust localization is a key requirement for achieving such autonomous mobility. Typically, localization methods using sensors such as cameras and LiDAR construct a map in advance from features of the driving environment. The robot then estimates its pose by matching the currently observed features to the pre-prepared map. However, environmental changes create discrepancies between the current environment and the pre-prepared map, leading to localization failure. This paper presents a map selection method for robust localization, which selects a map that reflects the current environment from multiple pre-prepared maps constructed under different environmental conditions. Although existing methods are limited to a single sensor, the proposed method can be applied to different types of sensors in a unified manner by handling sensor information at the pose information layer. To achieve this, the method utilizes the relative pose difference between the interoceptive sensor and the map, which is less susceptible to environmental changes, thereby enabling appropriate map selection under varying environmental conditions. The experiment was conducted using a robot equipped with a stereo camera in an environment with four conditions. The results showed maximum localization errors of 0.10–0.24 m and mean errors of 0.03–0.04 m, demonstrating robust localization through the selection of an appropriate map that reflects the current conditions.**

I. INTRODUCTION

Robots are expected to move autonomously in diverse environments, including the automated transportation of goods, the automated security and cleaning of office buildings, the automated inspection of social infrastructure, and the operation of autonomous vehicles. Because robots rely on pose information for fundamental autonomous mobile technologies, such as map construction, path planning, and obstacle avoidance, localization to estimate the robot's pose based on sensor observations is crucial. Various localization methods using sensors such as LiDAR and cameras have been proposed [1]–[5]. However, localization accuracy decreases with sensor environmental dependencies and environmental changes.

As examples of sensor environmental dependencies, LiDAR localization may be unreliable in open or tunnel-like environments with sparse geometric features, and camera-based localization may fail in low-light scenes where visual features cannot be extracted. Therefore, robust localization using a single sensor in diverse environments is challenging. To address this issue, researchers have enhanced the robustness of localization by employing multiple sensors with different characteristics to complement each other's weaknesses [6]–[13].

Environmental changes comprise time of day, door state (open/closed), and lighting. Sensors such as LiDAR and cameras generally construct maps by extracting features from sensor data collected in the environment in advance. Then, by matching the current conditions and the pre-prepared map, localization is performed. Environmental changes that alter environmental features can cause mismatches between the current conditions and the pre-prepared map, potentially leading to localization failure. To address this issue, researchers have enhanced the robustness of localization by constructing multiple maps under different conditions and selecting the appropriate map that reflects the current conditions from multiple pre-prepared maps [14]–[19]. For example, Labbe et al. [14] proposed a method that selects the map that yields a sufficient number of matching inlier features with the current image. Similarly, Biber et al. [17], [18] proposed a method that selects a map by computing the likelihood of the current LiDAR data under the distribution model generated from each map, and choosing the one with the highest likelihood. In this way, existing methods using multiple maps are limited to a single sensor and therefore cannot adapt to diverse environments due to sensor environmental dependencies. To address this limitation, we are planning a localization system that is robust to sensor environmental dependencies and environmental changes by combining multiple sensor methods and multiple map methods. To this end, a map selection method that is independent of sensor type is required.

In this paper, as a first step toward our planned system, we propose a map selection method that ensures applicability to any sensor by converting sensor-specific data into pose information and handling it uniformly. By using the pose information of the map and the interoceptive sensor that does not rely on any environmental information, the method that is independent of sensor type aims to select the appropriate map that reflects the current conditions from multiple pre-prepared maps. In our experiments, we evaluate whether the proposed method can select the appropriate map and achieve robust localization under environmental changes.

II. METHOD

The system overview is shown in Fig.1. For efficient and accurate map selection, the method classifies multiple maps into two groups: valid maps M_{pos}, considered as candidates, and invalid maps M_{neg}, excluded from selection.

*This work was supported by Mitsubishi Electric Corporation

[1]Department of Information and Communication Engineering, Graduate School of Engineering, Nagoya University Fro-cho Chikusa-ku Nagoya Aichi Japan suzuki.takumi.v4@s.mail.nagoya-u.ac.jp

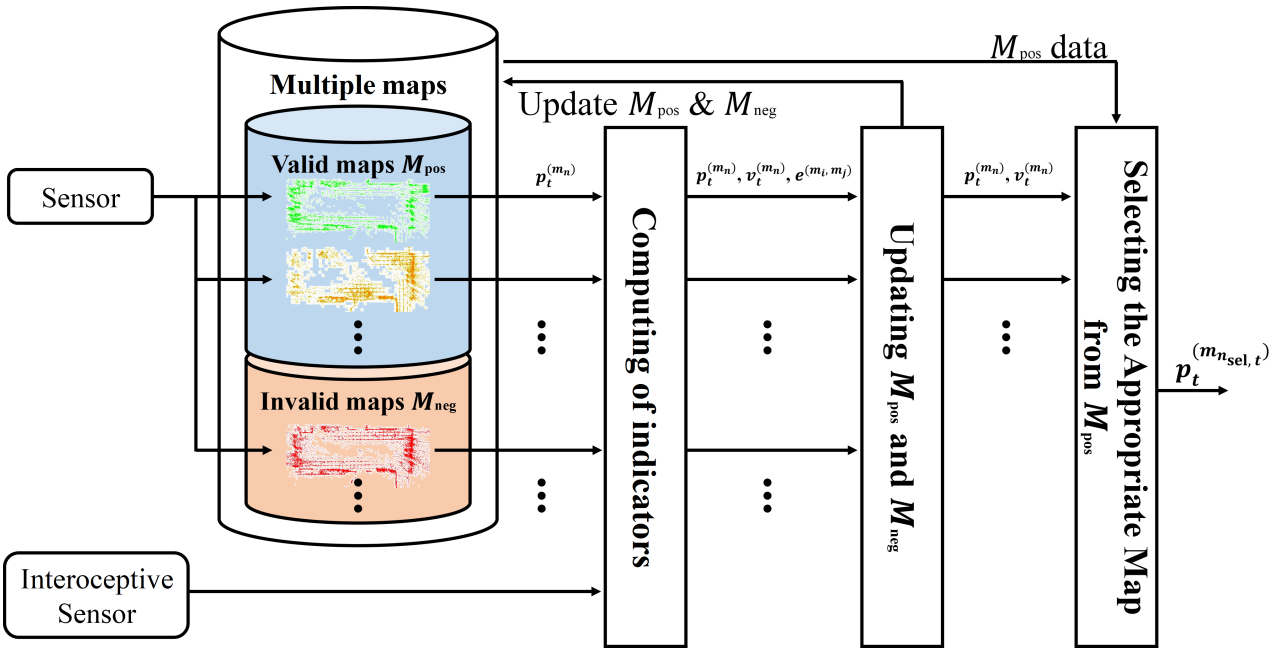


Fig. 1. System for selecting the map that best reflects the current conditions from multiple pre-prepared maps by evaluating their difference from an interoceptive sensor. The system classifies the maps into valid maps (M_{pos}) and invalid maps (M_{neg}) to enable effective and appropriate selection. It consists of three main components: (i) Computation of $v_t^{(m_n)}$ and $e^{(m_i,m_j)}$, which calculates the values used in the other two components; (ii) Updating M_{pos} and M_{neg}, which updates the classification of multiple maps based on these values; and (iii) Selecting the Appropriate Map from M_{pos}, which selects the most suitable map from M_{pos}.

And, this system consists of three modules: (i) computing of indicators, (ii) updating M_{pos} and M_{neg}, and (iii) selecting the appropriate map from M_{pos}.

In (ii) updating M_{pos} and M_{neg}, to build a map selection method that ensures applicability to any sensor, we handle sensor data at the pose information layer. We focused on the relative pose difference between the map and the interoceptive sensor. An initial investigation confirmed that, although the value can detect the occurrence of the localization error of map m_n, simply applying a threshold or using the map with the smallest value may not always select the appropriate map [20]. This is because the relative pose of the map becomes similar to that of the interoceptive sensor, even though the map's localization is incorrect. To address this, we detect a map with a localization error in advance and remove the maps from the selection candidates, thereby enabling more accurate map selection. A map with a localization error is detected in two steps: (i) when the absolute pose difference between maps is detected, one of the maps is assumed to have a localization error; and (ii) based on the relative pose difference, we then determine which map contains the error.

Sections II-A–II-C respectively describe (i) computing indicators, (ii) updating M_{pos} and M_{neg}, and (iii) selecting the appropriate map from M_{pos}.

A. Computing of Indicators

This section describes the computation of $v_t^{(m_n)}$, the relative pose difference of map m_n and $e^{(m_i,m_j)}$, the absolute pose difference between map m_i and map m_j at step t.

First, we compute $v_t^{(m_n)}$. The absolute pose $\mathbf{p}_t^{(m_n)}$ of map m_n at step t can be obtained from sensor data using an existing localization method. Similarly, the absolute pose $\mathbf{p}_t^{(i)}$ from an interoceptive sensor can be obtained by integrating the interoceptive sensor measurements over time. We obtain the mean relative pose $\delta\mathbf{p}_t^{(m_n)}$ of map m_n and $\delta\mathbf{p}_t^{(i)}$ of an interoceptive sensor by comuting the relative pose and applying a time-series mean filter with a window size w according to (1).

$$\delta\mathbf{p}_t^{(s)} = \frac{1}{w}\sum_{k=1}^{w}\left|\,\mathbf{p}_{t-k+1}^{(s)} - \mathbf{p}_{t-k}^{(s)}\,\right|, \quad s \in \{m_n, i\} \tag{1}$$

From $\delta\mathbf{p}_t^{(m_n)}$ and $\delta\mathbf{p}_t^{(i)}$, we obtain the difference between the mean relative pose obtained from an interoceptive sensor and map m_n (translation: $v_{d,t}^{(m_n)}$, rotation: $v_{\theta,t}^{(m_n)}$) according to (2) and (3).

$$v_{d,t}^{(m_n)} = \left|\,\|A\delta\mathbf{p}_t^{(m_n)}\| - \|A\delta\mathbf{p}_t^{(i)}\|\,\right| \tag{2}$$

$$v_{\theta,\,t}^{(m_n)} = \|B(\delta\mathbf{p}_t^{(m_n)} - \delta\mathbf{p}_t^{(i)})\| \tag{3}$$

Here, we define $A = \begin{bmatrix}1&0&0\\0&1&0\\0&0&0\end{bmatrix}$, $B = \begin{bmatrix}0&0&0\\0&0&0\\0&0&1\end{bmatrix}$. Using α, we combine $v_{d,t}^{(m_n)}$ and $v_{\theta,t}^{(m_n)}$ into a single value $v_t^{(m_n)}$ by computing their weighted average.

$$v_t^{(m_n)} = \alpha\, v_{d,\,t}^{(m_n)} + (1-\alpha)\, v_{\theta,\,t}^{(m_n)} \tag{4}$$

Next, we compute $e^{(m_i,m_j)}$, the difference between the absolute poses of map m_i and map m_j, using (5).

$$e^{(m_i,m_j)} = \|A(\mathbf{p}_t^{(m_i)} - \mathbf{p}_t^{(m_j)})\| \tag{5}$$

Algorithm 1 Updating M_{pos}

for $m_{\mathrm{pos1}} \in M_{\mathrm{pos}}$ **do**
  $f[m_{\mathrm{pos1}}] \leftarrow 0$
  for $m_{\mathrm{pos2}} = M_{\mathrm{pos}}$ **do**
    if $e^{(m_{\mathrm{pos1}}, m_{\mathrm{pos2}})} \geq e_{\mathrm{th}}$ **then**
      $f[m_{\mathrm{pos1}}] \leftarrow f[m_{\mathrm{pos1}}] \vee (\ v_t^{(m_{\mathrm{pos1}})} > v_t^{(m_{\mathrm{pos2}})})$
    end if
  end for
end for
for $m_{\mathrm{pos1}} \in M_{\mathrm{pos}}$ **do**
  if $f[m_{\mathrm{pos1}}]$ **then**
    $M_{\mathrm{pos}} \leftarrow \mathrm{Remove}(M_{\mathrm{pos}}, \text{“}m_{\mathrm{pos1}}\text{”})$
    $M_{\mathrm{neg}} \leftarrow \mathrm{Addition}(M_{\mathrm{neg}}, \text{“}m_{\mathrm{pos1}}\text{”})$
  end if
end for

Algorithm 2 Updating M_{neg}

for $m_{\mathrm{neg1}} \in M_{\mathrm{neg}}$ **do**
  $f[m_{\mathrm{neg1}}] \leftarrow 1$
  for $m_{\mathrm{pos1}} = M_{\mathrm{pos}}$ **do**
    $f[m_{\mathrm{neg1}}] \leftarrow f[m_{\mathrm{neg1}}] \wedge \left(e^{(m_{\mathrm{neg1}}, m_{\mathrm{pos1}})} < e_{\mathrm{th}}\right)$
  end for
end for
for $m_{\mathrm{neg1}} \in M_{\mathrm{neg}}$ **do**
  if $f[m_{\mathrm{neg1}}]$ **then**
    $M_{\mathrm{neg}} \leftarrow \mathrm{Remove}(M_{\mathrm{neg}}, \text{“}m_{\mathrm{neg1}}\text{”})$
    $M_{\mathrm{pos}} \leftarrow \mathrm{Addition}(M_{\mathrm{pos}}, \text{“}m_{\mathrm{neg1}}\text{”})$
  end if
end for

B. Updating M_{pos} and M_{neg}

This section describes the updating of M_{pos} and M_{neg}. This is performed after step w, taking into account a window size of the time-series mean filter w. Initially, all multiple maps are included in M_{pos}.

Using Algorithm 1, M_{pos} is updated. We check whether the map m_{pos1} in M_{pos} should be moved to M_{neg}. The decision is made by comparing the all map m_{pos2} in M_{pos}. If there exists a map m_{pos2} whose $e^{(m_{\mathrm{pos1}}, m_{\mathrm{pos2}})}$ is more than e_{th}, then either m_{pos1} or m_{pos2} has a localization error. If $v_t^{(m_{\mathrm{pos1}})}$ is more than $v_t^{(m_{\mathrm{pos2}})}$, the map m_{pos1} is considered incorrectly localized and is moved it from M_{pos} to M_{neg}.

Next, using Algorithm 2, M_{neg} is updated. We check whether the map m_{neg1} in M_{neg} should be moved back to M_{pos}. The decision is made by comparing the all map m_{pos1} in M_{pos}. If all $e^{(m_{\mathrm{neg1}}, m_{\mathrm{pos1}})}$ less than e_{th}, the map m_{neg1} is considered correctly localized and is moved back it from M_{neg} to M_{pos}.

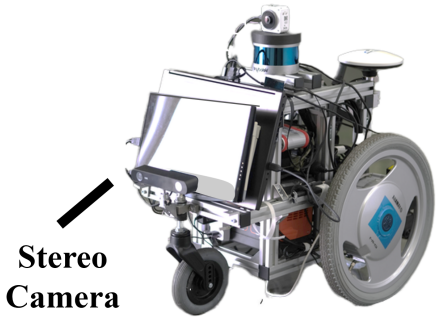


Fig. 2. Robot equipped with a stereo camera, capable of recording stereo camera and wheel odometry data.

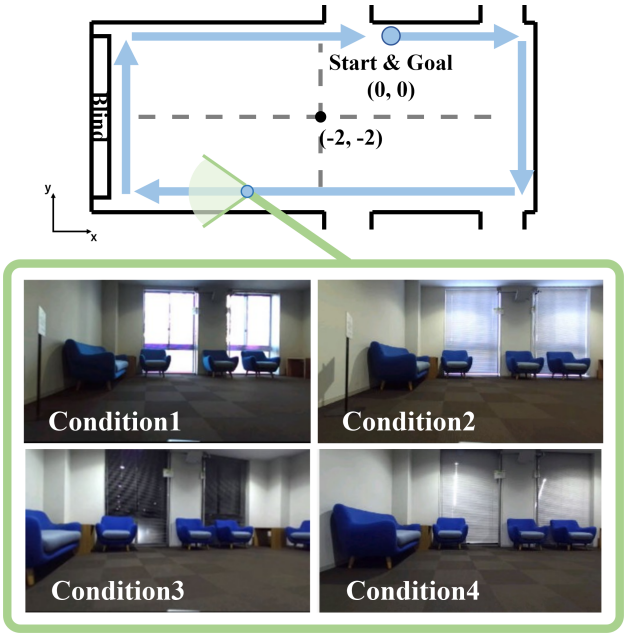


Fig. 3. Environment with two environmental changes: two times of day (mid-day or night) and two blind states (open or closed). condition 1 corresponds to mid-day and open, condition 2 to mid-day and closed, condition 3 to night and open, and condition 4 to night and closed. The route taken through the environment is shown by the light blue arrow.

C. Selecting the Appropriate Map from M_{pos}

From M_{pos}, we select the most appropriate map at time t using (6).

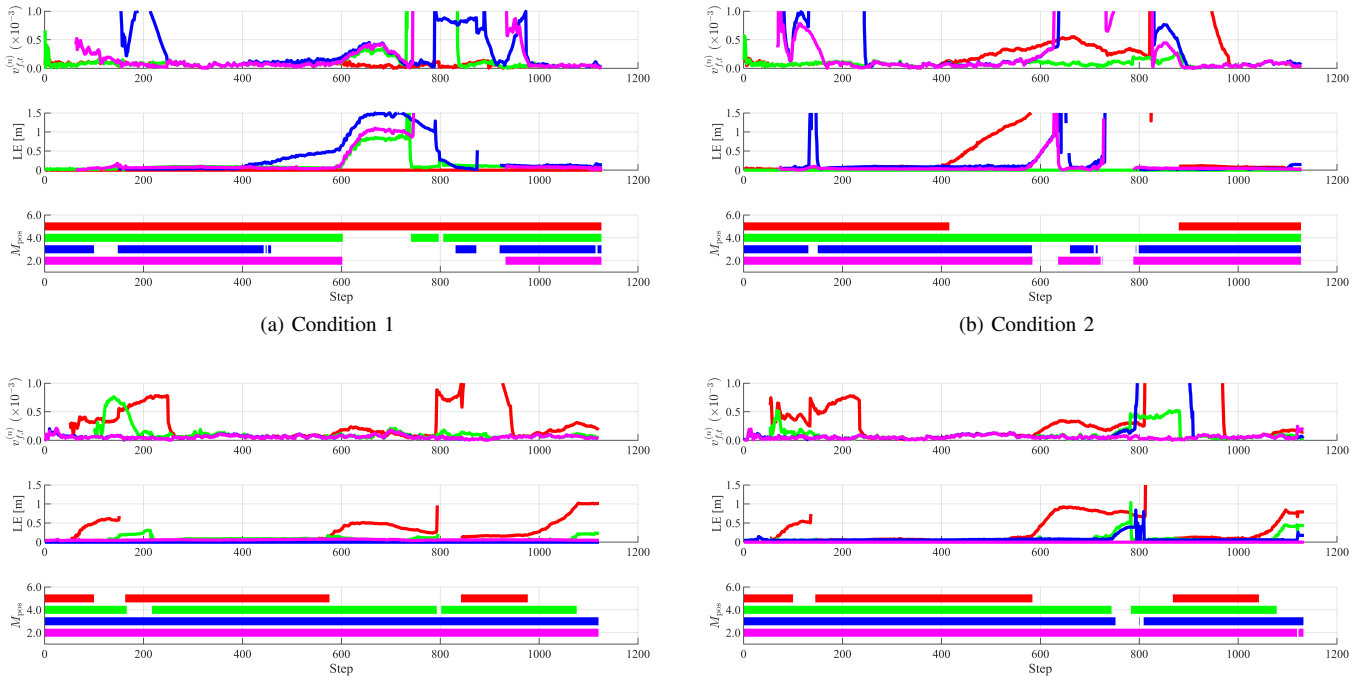
$$m_{n_{\mathrm{sel},\,t}} = \underset{m_n \in M_{\mathrm{pos}}}{\arg\min}\ v_t^{(m_n)} \tag{6}$$

Accordingly, we use the pose $\mathbf{p}_t^{(m_{n_{\mathrm{sel},\,t}})}$ from map $m_{n_{\mathrm{sel},\,t}}$ as the final localization.

III. EXPERIMENT

We used a mobile robot (Fig.2) built on a powered wheelchair unit (YAMAHA Joy Unit X PLUS+) equipped with a stereo camera (Stereolabs ZED2). We use ORB-SLAM2 [21] for camera-based localization. We use wheel odometry as an interoceptive sensor.

In a used environment, we prepared four conditions by combining two times of day (mid-day or Night) and two



(a) Condition 1 (b) Condition 2

(c) Condition 3 (d) Condition 4

Fig. 4. Time-series data of $v_{f,t}^{(n)}$, localization error (LE), and M_{pos}. The red, green, blue, and purple lines indicate maps 1, 2, 3, and 4, respectively. M_{pos} row is plotted only when the corresponding map is included in M_{pos}, and left blank otherwise.

TABLE I
ENVIRONMENTAL CONDITION TABLE

condition	Time of day	Blind
1	Mid-day	Open
2	Mid-day	Close
3	Night	Open
4	Night	Close

blind states (Open or Closed), as shown in Table I and Fig.3. For each of the four conditions, sensor data were collected twice along the light-blue route shown in Fig.3, with one dataset used for map construction and the other for evaluation. First, using the map construction data, we constructed maps 1–4 under conditions 1–4, respectively. Next, using the evaluation data under four conditions, localization was performed by our method, selecting the appropriate map from the four maps.

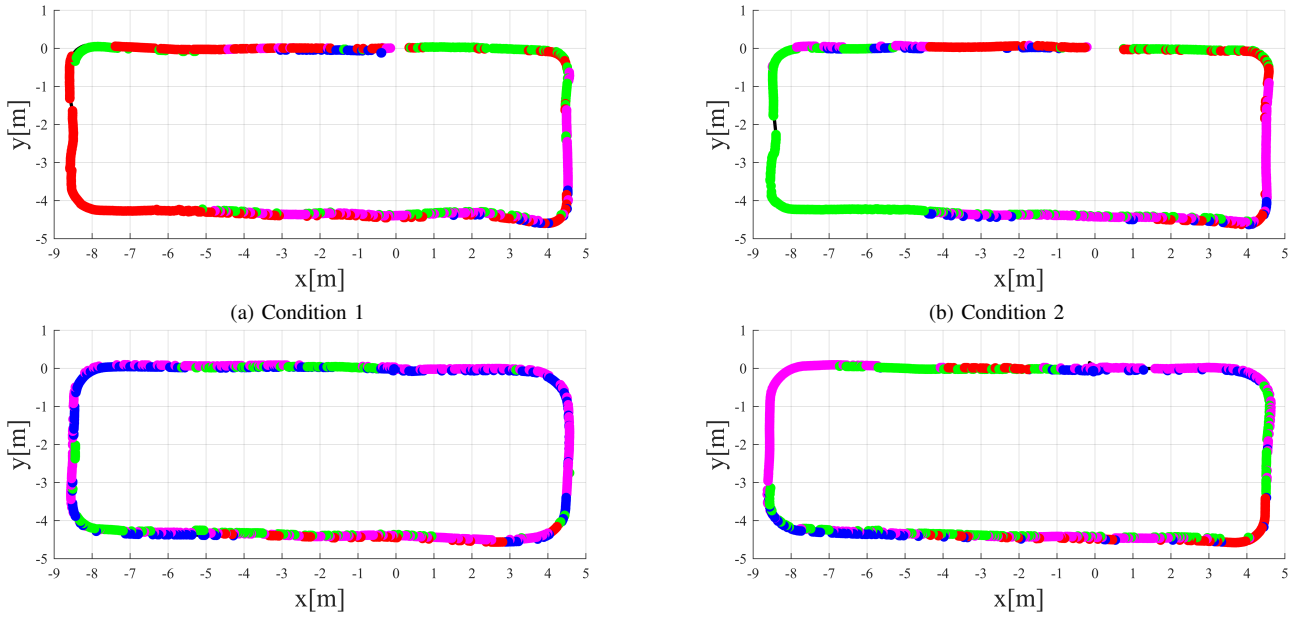
In this experiment, we assume that at least one appropriate map exists. We evaluate whether our method can exclude maps with localization errors from M_{pos} by checking both the time-series data of M_{pos} and the localization errors of each map. We also assess whether the method can achieve robust localization under environmental changes by examining both the localization errors and the map selection results. As ground truth of localization, we use the pose of the map constructed under the same condition of the current environment. The parameters for this experiment were set to $w = 100, \alpha = 0.2, e_{\mathrm{th}} = 0.2$.

A. *Time-Series data of* M_{pos}

Fig.4 shows, from top to bottom, the time-series of $v_t^{(m_n)}$, the localization error (LE), and M_{pos}. The red, green, blue, and purple lines correspond to maps 1, 2, 3, and 4, respectively.

Fig.4(a) shows the result of condition 1. In around step 500, map 3 (blue) exhibits a localization error. By comparing the localization difference of other maps and using $v_t^{(m_n)}$, map 3 (blue) is excluded from M_{pos}. Between steps 550 and 700, map 2 (green) and map 4 (purple) exhibit a localization error and are similarly excluded from M_{pos}. Near step 680, although maps 3 (blue) and 4 (purple) still have localization errors, their $v_t^{(m_n)}$ drop to levels comparable to map 1 (red), which has no error. However, because the errors were already detected and maps 3 (blue) and 4 (purple) were preemptively removed from M_{pos}, the correct map selection remains possible. Moreover, between steps 600 and 800, the appropriate map for condition 1 (map 1 (red)) stays within M_{pos}. Around step 800, the localization errors for maps 2, 3, and 4 approach 0m, and based on $e^{(m_i,m_j)}$, these maps are moved back into M_{pos}. The results for conditions 2, 3, and 4 (Fig.4(b), (c), and (d)) show the same pattern: whenever a map incurs a localization error, it is detected and excluded from M_{pos}.

Here, from both the time-series data of M_{pos} and the localization errors of each map, the system excludes maps with localization errors from M_{pos}, and includes maps with no localization errors in M_{pos}.



(a) Condition 1 (b) Condition 2

(c) Condition 3 (d) Condition 4

Fig. 5. Localization trajectory and map selection results. The trajectory color indicates the selected map and is shown in red, green, blue, or purple to represent maps 1, 2, 3, and 4, respectively.

TABLE II
LOCALIZATION ERROR

Condition	Map	Maximum[m]	Mean±SD[m]
1	selected map	0.14	0.03 ±0.04
	map 2	2.61	0.16 ±0.25
	map 3	1.55	0.38 ±0.49
	map 4	3.82	0.21 ±0.35
2	selected map	0.10	0.03 ±0.03
	map 1	3.38	0.68 ±0.91
	map 3	29.9	0.27 ±1.41
	map 4	9.16	0.09 ±0.38
3	selected map	0.14	0.04 ±0.03
	map 1	1.02	0.26 ±0.26
	map 2	0.31	0.09 ±0.05
	map 4	0.09	0.06 ±0.01
4	selected map	0.26	0.03 ±0.04
	map 1	1.97	0.30 ±0.31
	map 2	1.04	0.08 ±0.11
	map 3	0.84	0.08 ±0.07

B. Localization Result

The localization errors (maximum and mean ± standard deviation) for the selected map and each individual map are shown as Table. II. Note that the map constructed under the same condition are excluded from the evaluation as they are used as ground truth.

We first focus on the maximum localization error. In all conditions, the selected map keeps the error within 0.3 m, confirming that localization never broke down. In contrast, for the individual maps, map 1 (under conditions 2, 3, and 4), map 2 (under conditions 1 and 4), map 3 (under conditions 1 and 2), and map 4 (under conditions 1 and 2) show the maximum localization error exceeding 1.0 m, indicating localization failure.

Next, we focus on the mean localization error. In all conditions, the selected map achieves a mean localization error within 0.03–0.04 m and a standard deviation within 0.03–0.04 m, demonstrating consistently stable localization. In contrast, for the individual maps, map 1 (under conditions 2, 3, and 4), map 2 (condition 1), map 3 (conditions 1 and 2), and map 4 (condition 1) exhibit mean localization errors exceeding 0.1 m and standard deviations exceeding 0.1 m, indicating that consistent and stable localization cannot be achieved.

From both the maximum and mean localization errors, it is evident that only the selected map maintains stable localization without collapse across all conditions, thereby confirming its robustness against environmental changes.

We examine the map selection results. Fig.5 presents the localization trajectories under each condition. The color of the localization trajectory corresponds to the selected map, with each map ' s color defined as shown in Fig.4. In the environment shown in Fig.3, environmental changes due to time of day and blinds occur mainly on the left side of the environment. Therefore, it is expected that environmental changes have a significant impact on localization on the left side. In contrast, on the right side of the environment, the impact of environmental changes on localization is considered negligible. In the following, we discuss the map selection results for these two regions: the left and right sides of the environment.

Fig.5(a) shows the localization trajectory of condition 1. On the left side of the environment, map 1 (red) is mainly selected. Since this region is affected by environmental changes, high-accuracy localization cannot be expected without using map 1, which was constructed under the same conditions as the driving environment. Consequently, the selection of map 1 is reasonable, and it enables stable

localization. In contrast, on the right side of the environment, various maps are selected. As this region is not significantly affected by environmental changes, stable localization can be expected with any of the maps. Therefore, the selection of various maps is reasonable, and stable localization is achieved regardless of which map is chosen. Similar trends were observed for conditions 2 and 4.

Fig.5(c) shows the localization trajectory of condition 3. On the left side of the environment, maps 3 (blue) and 4 (purple) are mainly selected. Although this region is affected by environmental changes, map 3, constructed under different conditions from the environment, still achieves consistently stable localization, as shown in Fig.4(c). Therefore, the selection of maps 3 and 4 is reasonable, and it enables localization performance equivalent to the ground truth. On the right side of the environment, similar trends were observed for conditions 1, 2, and 4.

From the map selection results, it was demonstrated that the proposed method was able to select an appropriate map from multiple pre-prepared maps corresponding to the current conditions, both in regions affected by environmental changes and in regions unaffected by them, thereby achieving robust localization.

IV. CONCLUSIONS

In this paper, we propose a method for selecting multiple maps that applies to any sensor and is less susceptible to environmental changes and sensor environment dependencies. To avoid the effects of environmental changes, we used an interoceptive sensor that does not use environmental information. To address the issues of initial investigation, we introduced a system that classifies multiple maps into M_{pos} and M_{neg}. When updating M_{pos} and M_{neg}, $v_t^{(m_n)}$ and $e^{(m_i,m_j)}$ were used to detect maps with localization errors, which were then excluded from M_{pos} and add to M_{neg}. When selecting the map, map selection is performed from M_{pos}, ensuring that the process remains effective and appropriate. In our experiment, maps with localization errors were appropriately excluded from M_{pos}, and by selecting from M_{pos}, robust localization was achieved.

Since the proposed method assumes that the system maintains maps suitable for the driving environment, future work will focus on enabling it to add maps when insufficient and remove them when unnecessary. In addition, we plan to extend the system to accommodate various sensors, including multiple map-based sensors, multiple non-map-based sensors, and multiple interoceptive sensors, in order to realize a more robust localization system. We will validate the effectiveness of this approach through experiments on large-scale datasets covering diverse environments, including both indoor and outdoor scenes.

REFERENCES

[1] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE transactions on robotics*, Vol. 31, No. 5, pp. 1147–1163, 2015.

[2] Guoquan Huang. Visual-inertial navigation: A concise review. In *2019 international conference on robotics and automation (ICRA)*, pp. 9572–9582. IEEE, 2019.

[3] Lanbo Liu, Zijian Liu, and Benjamin E Barrowes. Through-wall bio-radiolocation with UWB impulse radar: Observation, simulation and signal extraction. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 4, No. 4, pp. 791–798, 2011.

[4] Suining He and S-H Gary Chan. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys & Tutorials*, Vol. 18, No. 1, pp. 466–490, 2015.

[5] Ramsey Faragher and Robert Harle. Location fingerprinting with bluetooth low energy beacons. *IEEE journal on Selected Areas in Communications*, Vol. 33, No. 11, pp. 2418–2428, 2015.

[6] Aaronkumar Ehambram, Raphael Voges, Claus Brenner, and Bernardo Wagner. Interval-based Visual-Inertial LiDAR SLAM with Anchoring Poses. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 7589–7596, 2022.

[7] Chunran Zheng, Qingyan Zhu, Wei Xu, Xiyuan Liu, Qizhi Guo, and Fu Zhang. FAST-LIVO: Fast and Tightly-coupled Sparse-Direct LiDAR-Inertial-Visual Odometry. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4003–4009, 2022.

[8] Danpeng Chen, Shuai Wang, Weijian Xie, Shangjin Zhai, Nan Wang, Hujun Bao, and Guofeng Zhang. VIP-SLAM: An Efficient Tightly-Coupled RGB-D Visual Inertial Planar SLAM. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 5615–5621, 2022.

[9] Yang Song, Mingyang Guan, Wee Peng Tay, Choi Look Law, and Changyun Wen. UWB/LiDAR Fusion For Cooperative Range-Only SLAM. *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6568–6574, 2018.

[10] Ernesto Martín-Gorostiza, Miguel A. García-Garrido, Daniel Pizarro, Patricia Torres, Manuel Ocaña Miguel, and David Salido-Monzú. Infrared and Camera Fusion Sensor for Indoor Positioning. In *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8, 2019.

[11] Xianjia Yu, Paola Torrico Morrn, Sahar Salimpour, Jorge Pena Queralta, and Tomi Westerlund. Loosely Coupled Odometry, UWB Ranging, and Cooperative Spatial Detection for Relative Monte-Carlo Multi-Robot Localization. *arXiv preprint arXiv:2304.06264*, 2023.

[12] Anderson G Pires, Paulo AF Rezeck, Rodrigo A Chaves, Douglas G Macharet, and Luiz Chaimowicz. Cooperative localization and mapping with robotic swarms. *Journal of Intelligent & Robotic Systems*, Vol. 102, No. 2, p. 47, 2021.

[13] Takumi Suzuki, Yuki Funabora, Shinji Doki, Kae Doki, and Mitsuhiro Yamazumi. DSFS: Dynamic Sensor Fusion System for robust localization with diverse sensing information. *Journal of Robotics and Mechatronics*, 2025. in press.

[14] Mathieu Labbé and François Michaud. Multi-session visual SLAM for illumination-invariant re-localization in indoor environments. *Frontiers in Robotics and AI*, Vol. 9, p. 801886, 2022.

[15] Chris Linegar, Winston Churchill, and Paul Newman. Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation. In *2015 IEEE International conference on robotics and automation (ICRA)*, pp. 90–97. IEEE, 2015.

[16] Timothy Morris, Feras Dayoub, Peter Corke, Gordon Wyeth, and Ben Upcroft. Multiple map hypotheses for planning and navigating in non-stationary environments. In *2014 IEEE international conference on robotics and automation (ICRA)*, pp. 2765–2770. IEEE, 2014.

[17] Peter Biber and Tom Duckett. Experimental analysis of sample-based maps for long-term SLAM. *The International Journal of Robotics Research*, Vol. 28, No. 1, pp. 20–33, 2009.

[18] Peter Biber, Tom Duckett, et al. Dynamic maps for long-term operation of mobile service robots. In *Robotics: science and systems*, pp. 17–24, 2005.

[19] Cyrill Stachniss and Wolfram Burgard. Mobile robot mapping and localization in non-static environments. In *aaai*, pp. 1324–1329, 2005.

[20] Taiki Nishikawa, Takumi Suzuki, Yuki Funabora, Shinji Doki, and Kae Doki. Comparison of Relative Posture Between Maps and Odometry for Selection of Maps Suitable for the Environment. In *Proc. 2025 Annual Tokai-Section Joint Conference on Electrical, Electronics, Information, and Related Engineering*, September 2025. F4-4 [in Japanese].

[21] Raúl Mur-Artal and Juan D. Tardós. ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. *IEEE Transactions on Robotics*, Vol. 33, No. 5, pp. 1255–1262, 2017.