360ST-Mapping: An Online Semantics-Guided Topological Mapping Module for Omnidirectional Visual SLAM

Hongji Liu, Huajian Huang, Sai-Kit Yeung and Ming Liu

***Abstract*— Topological map as an abstract representation of the observed environment has the advantage in path planning and navigation. Here we proposed an online topological mapping method, 360ST-Mapping, by making use of omnidirectional vision. The 360° field of view allows the agent to obtain consistent observation and incrementally extract topological environment information. Moreover, we leverage semantic information to guide topological places recognition further improving performance. The topological map possessing semantic information has the potential to support semantics-related advanced tasks. After combining the topological mapping module with the omnidirectional visual SLAM, we conduct extensive experiments in several large-scale indoor scenes to validate the effectiveness.**

I. INTRODUCTION

The mapping module of SLAM (Simultaneous Localization and Mapping) takes responsibility to incrementally build a proper representation of the observed environment. The representation directly reflects how the agents recognize the world. With the development of SLAM [1], [2], [3], accurate geometric maps of the world denoted as point cloud, occupancy grid or truncated signed distance function (TSDF) can be obtained. To facilitate path planning and navigation, the geometric reconstruction is further abstracted into the topological map. As a simplified representation, the topological map is composed of an undirected graph. Each node of the graph represents the location of a place while edges represent the connected or navigable relationship between nodes.

To build a topological map, it is important to correctly identify all the relatively isolated places in the scene. In general, a place with clear boundaries (e.g. wall and doors) can be identified as a node. Minimizing the number of unnecessary topological nodes allows the topological map to do its job while kept as lightweight as possible. As it is tricky for a purely visual system to determine node assignment, how to properly segment the map and generate the nodes is still an open problem. The exiting method, TopoMap [4], introduced an offline approach that decomposes the recovered free space into a larger number of clusters and then iteratively merges the clusters. However, in practice, an online mapping module supporting efficient scene abstraction is in demand.

Hongji Liu is with The Hong Kong University of Science and Technology (Guangzhou), Thrust of Robotics & Autonomous Systems, Ming Liu is with The Hong Kong University of Science and Technology (Guangzhou), Thrust of Robotics & Autonomous Systems and The Hong Kong University of Science and Technology, Department of Electronic and Computer Engineering, Huajian Huang and Sai-Kit Yeung are with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology. {hliucq, hhuangbg}@connect.ust.hk, {saikit, eelium}@ust.hk

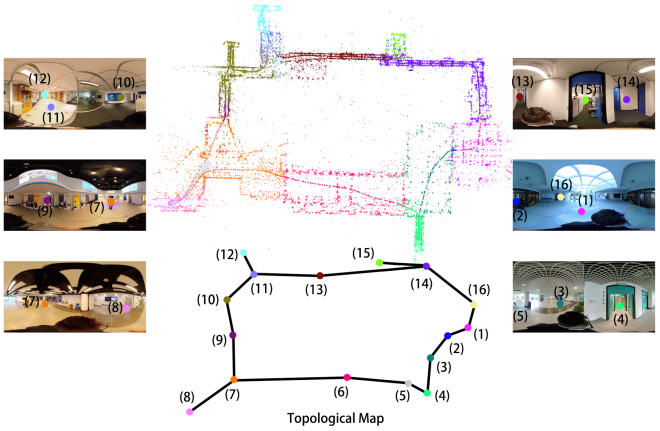


Fig. 1. The mapping results on the academic building. Our proposed method 360ST-Mapping is integrated into an omnidirectional visual SLAM. It can identify individual places (denoted as distinct colors) and generate topological map online.

We note that 360-degree field-of-view has a great consistency of observation [5] which theoretically allows for robust visual landmarks tracking. Hence in this paper, we seek to fill the gap by exploiting the stable landmark co-visibility relationships between omnidirectional vision images.

When the landmarks change dramatically, the agent can be considered having entered a new place such that a new node is created. Consequently, We use the co-visibility relationship as the first step to measure the extent of the scene change. As it does not require prior knowledge, it can be seamlessly carried out online with the operation of the SLAM system. However, merely relying on geometric information is vulnerable in the situation of geometric degradation. Therefore, we also take advantage of an object detector to extract semantic information of the scene. According to the results of object detection, the semantic coefficient denoted as the weighted Hamming distances among the incoming and scene reference frames is calculated to guide topological node assignment. In addition, we will store the extracted semantic information into each node because it can provide a higher-level understanding of scenes. Finally, we choose an indirect SLAM system, OpenVSLAM [6], supporting 360° camera as the system framework to conduct experiments. Although its reconstruction is sparse compared to 360VO [5], OpenVSLAM supports loop-closure which is necessary for topological mapping. We collect large-scale indoor scenes to test our method. The results verify that our methods can extract accurate topological relationships of the scenes, as the Fig 1 shows. Basically, the main contribution of this paper

can be summarized as follows:

1) We proposed a simple yet effective method, 360ST-Mapping, which is able to build topological maps online.
2) We introduced a weighted Hamming distance to measure the semantic difference between images.
3) The method is seamlessly integrated into omnidirectional visual SLAM with a low computational cost.

II. RELATED WORKS

Extracting topological characteristics of the observed environment and establishing a graph to represent the discrete spaces are effective methods for localization, path planning, and navigation [7]. According to different requirements of the specific tasks, the definitions of the topological map are various. Konolige et al. [8] defined a navigation graph on the basis of a global pose graph constructed by SLAM in the form of occupancy grids using the Ray Tracing method to do the next navigation task. Rosa et al. [9] were inspired by the behavior of bees in the construction of each honeycomb then used multiple UAVs to build a honeycomb liked topological map. Dall'Osto et al. [10] treated the tracked points in the "repeat" phase in the "teach and repeat" problem as topological nodes to guide the robot to move according to the taught route. Wen et al. [11] defined the topological node as the different semantic object in the scene, namely a chair is a node and a sofa is another node, which is not in the sense of place. Xue et al. [12] used traversed points to create topological nodes which represent navigable points, the topological edges are built between each pair of nodes if there are no obstacle points between them. But concerning large scale indoor scenes, the common definition is that a topological map is an undirected graph reflecting the relationship of spaces with relatively clear boundaries [13].

To establish a topological map, most vision-based methods are based on visual features and descriptors, either global descriptors or local descriptors. Goedemé et al.[14] used a combination of two different kinds of wide baseline features to help detect loop closure and based on Dempster-Shafer theory to decide whether merge or separate topological nodes. Liu et al.[15] proposed a lightweight adaptive descriptor named FAST to describe the scene and judge the scene changes. Such kinds of methods are always limited by the simplicity of information used so that they are easily influenced by great scene changes such as illumination or the layout of objects. Garcia-Fidalgo et al.[16] used both global description features and local description features to detect loop closures, based on which determined whether to create a new node or not. To enhance robustness, Bayesian inference used in [17] is to find the topological structure while [18] used mutual information graph to segment the topological regions. We first explore the omnidirectional co-visibility of 360° images for topological mapping.

Compared to online methods, most of the offline methods rely on reconstructed models and cannot be integrated into SLAM system. Blochliger et al.[4] extracts discrete approximate free space information from sparse landmarks by using voxel-based Truncated Signed Distance Fields. Oleynikova et al. [19] extracted sparse 3D Topological graph based on ESDF map. He et al.[20] built three-level topological graph(storey-region-volume) from a complete 3D point cloud map of the scene. Rosinol et al.[21] and Ravichandran et al.[22] did great jobs on the multi-hierarchical map which will be a great help for all kinds of robot tasks, this kind of map relied on the pre-constructed large scale mesh map of the environment. The performance of these methods depends on the quality of the basic map to some extent. [23] and [24] used also co-visibility of landmarks as a judgment basis. But what the biggest difference here is their method only begins after collecting all the images of the scene, and distinguishes all independent node clusters based on similarity in the post-processing stage. Our method does not rely on a prior map while can distinguish scenes and build the topological graph during tracking.

Different from pure vision-based methods. Vale and Ribeiro[25] used a set of features acquired from laser and sonars to represent the state of the topological place. Islam et al.[26] classify the topological nodes according to the free space shape feature to build the topological map with various sensors including camera, sonar sensor, etc. Shin et al.[27] used WI-FI fingerprint as a base to build the topological map. Wen et al.[11] based on stereo Visual-Inertial Odometry to build the semi-dense topological map of the scene. The online approach [13] exploits range information from laser and depth cameras to determine the gaps and doors which are the landmarks for new node creation.

III. METHODOLOGY

In terms of large-scale indoor scenes, each node of the corresponding topological map represents a relatively independent place, such as rooms and corridors. To extract a proper topological map, we propose a module 360ST-Mapping taking advantage of omnidirectional co-visibility and semantic information provided by an object detector to accurately identify distinct spaces. 360ST-Mapping module works on the basis of visual SLAM, shown in the Fig. 2.

Right after a new keyframe is created in the 360ST-Mapping Module, the system selects a proper scene reference frame. Then according to the co-visibility ratio and object detection results, the system compares the two keyframes to judge whether the new keyframe belongs to a new scene. If yes, a new topological node will be created. Then the system will refine the whole topological map online by culling redundant nodes. At the same time, the loop closure module of the SLAM system will also cooperate with 360ST-Mapping module to correct the topological relation where the loop closure occurs.

A. Scene Reference Keyframe

Our system decides whether to create a new topology node based on the comparison between a reference keyframe, which we call scene reference keyframe, and the current keyframe. If we simply select the latest keyframe as the scene reference frame, the feature correspondings between consecutive keyframes are always strong. It means that we

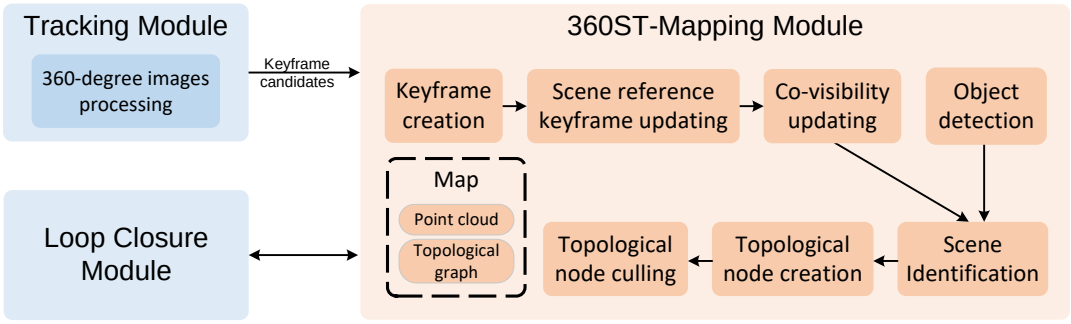


Fig. 2. A schematic diagram of the omnidirectional visual SLAM system after combining our proposed method, 360ST-Mapping.

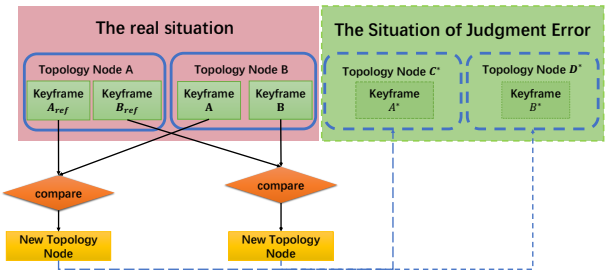


Fig. 3. Scene reference frame selection trap. If the previous frame has been determined to be a new topological node, and the scene reference frame the current frame compared with still belongs to the previous topological node according to the distance selection rule, then two consecutive new topological nodes will be generated.

cannot rely on the variance of consecutive keyframes to determine the assignment of topological nodes. Conversely, we need to select a representative keyframe of each place as the scene reference keyframe. Therefore, We designed a practical scene reference keyframe selection mechanism. Firstly, the scene reference keyframe should have enough translation from the current location. The threshold is set as 1 experimentally. Further, we check whether there has been a new topological node existed between the scene reference keyframe and the current keyframe. If there has been a new topological node, we should adjust the scene reference keyframe to the first keyframe of the latest topological node. This is because sometimes after comparing the current keyframe with the scene reference keyframe, it is found that the current keyframe belongs to a new topological node. However, the following keyframe is still compared with the keyframe of the previous topological node so it is also considered to lead to a new topological node. Fig. 3 depicts the problem scenario. Algorithm 1 describes the selection process in details.

Algorithm 1: represent frame Selection

input : All past keyframes KF_{t-*} in the database and current keyframe KF_t
output: The selected keyframe KF_{I_r}

Acquire all the past keyframes $\{KF_{t-i}, i = 1,2,3...n\}$;
for $i \leftarrow 1$ **to** n **do**
  Judge whether KF_{t-i} *is still valid*;
  if $D(KF_{t-i}, KF_t) > 1$ *and* $D(KF_{t-i}, KF_t) < \min(D(KF_{t-*}, KF_t))$ **then**
    $\min(D(KF_{t-*}, KF_t)) = D(KF_{t-i}, KF_t)$
    $I_r = t - i$
for $i \leftarrow I_r + 1$ **to** t **do**
  Judge whether KF_i *is still valid*;
  if KF_i *leads a new topological node* **then**
    $I_r = i$
return KF_{I_r}

B. Co-visibility

That two frames have a co-visibility relationship indicates they can observe the same landmarks. We define the co-visibility ratio as the ratio of the number of landmarks that can be observed in common between two frames to the sum of the total number of landmarks observed in the two frames, that is, a kind of Intersection of Union. When the co-visibility ratio between the scene reference keyframe and the current keyframe is lower than a specific threshold, it means that the landmarks observed in these two frames have changed significantly. It is reasonable to believe that the current frame is going into a different place from the scene reference frame. Hence, the co-visibility ratio can be used as one of the judge standards for the topological place change. The co-visibility ratio can be calculated by,

$$R_{Co} = \frac{N^l_{KF_c,KF_r}}{N^l_{KF_c,KF_r} + N^l_{KF_r,\overline{KF_c}} + N^l_{KF_c,\overline{KF_r}}} \tag{1}$$

where $N^l_{KF_c,\overline{KF_r}}$ and $N^l_{KF_r,\overline{KF_c}}$ are the number of landmarks observed only in the current keyframe and the scene reference keyframe respectively. $N^l_{KF_c,KF_r}$ is the number of landmarks that can be observed both in the two frames. The probability that the current frame belongs to a different topological node from the scene reference frame according to the co-visibility ratio then can be calculated as,

$$P_{Co} = 1 - R_{Co} \tag{2}$$

The co-visibility ratio can be very low even when the

current frame is in the same topological place as the scene reference frame by moving for a certain distance. Because the landmarks observed in previous frames may not be visible in the current frame. It is a double-edged sword. The advantage is that we can use it to identify almost all the topological places change while the disadvantage is that we will construct a lot of redundant topological nodes. Therefore, we need more judgment standards to make a comprehensive judgment.

C. Object Detection

1) The model: A place has its own function so that the classes of objects inside are usually unique. Hence naturally, we think of using object detection to assist in judging scene changes. Here in our system, we used YOLOv4[28] trained on COCO data set with 80 types of objects as the object detection support because of its high precision and efficiency.

2) Weighted Hamming distance: We express the result of object detection as an 80 dimensional vector (related to the total number of classes of object detection), and the value of each element is the number of objects of that class. To measure semantic variance between two images, we introduce the weighted Hamming distance that balances the effect of object class distribution and the number of objects detected in the scene. First, we calculate the proportion of detected objects in the two frames as weight ω_c. The larger the proportion, the more important the object is. When calculating the object detection vector distance between two frames, if the object appears only in one frame, the distance increases the weight of the object. If it appears or does not appear in both frames, the distance is not increased. In general, the probability of scene change judged from the object detection results P_{Ob} can be formulated as,

$$P_{Ob} = \sum_{c \in \mathbb{C}} \omega_c I_c(KF_c, KF_r), \tag{3}$$

$$\omega_c = \frac{N^c_{KF_c} + N^c_{KF_r}}{\sum_{c \in \mathbb{C}} N^c_{KF_c} + \sum_{c \in \mathbb{C}} N^c_{KF_r}}, \tag{4}$$

$$I_c(KF_c, KF_r) = \begin{cases} 0, & \text{if } \left(N^c_{KF_r} - th\right)\left(N^c_{KF_c} - th\right) > 0 \\ 1, & otherwise \end{cases}. \tag{5}$$

where $N^c_{KF_c}$ and $N^c_{KF_r}$ are the number of objects of class c detected in the current and scene reference keyframe respectively, $\mathbb{C}$ is the set of all the classes. th can take any value between 0 and 1, just to distinguish whether there are objects of class c in the scene.

D. The probability model

After calculating the place change probability based on co-visibility and semantic coefficient respectively, we simply combine the two results to calculate the final topological place change probability,

$$P_{change} = \frac{P_{Co} + P_{Ob}}{2} \tag{6}$$

When the P_{change} between the current frame and the represent frame is over a certain threshold(in our paper, we use 0.5) we will think that the topological scenario has indeed changed so create a new topological node and update the connection relationship between topological nodes.

E. Extreme Case Handling

We need to deal with some algorithm exceptions that may be led by the failure of the base SLAM algorithm during the operation of the system to enhance the robustness of the topological mapping. Tracking lost happens from time to time due to the failure of extracting or matching feature points between keyframes. The omnidirectional visual SLAM is no exception. When tracking is resumed from the lost state, we will create a new topological node if the current frame is farther than a certain distance from the scene reference frame. The reason for such a handling strategy is: if the tracking fails, it indicates that the environment features change greatly, otherwise the tracking will not be lost. When the distance is too large, we deem the agent has exceeded the range of a topological place. And we will not build a connection relationship between the new topological node and the last one, because we cannot determine whether there is a passable connection relationship between the two topological nodes during the loss of tracking.

F. Topological Node Culling

To reduce the redundant topological nodes in the map, we use some post-processing steps to refine the topological places recognition results.

1) Local consistency filtering mechanism: The algorithm generates redundant topological nodes in some frames due to the change of view angle and occlusion of the camera's field of view. However, we find that sometimes these keyframes still maintain the co-visibility relationship with surrounding keyframes. We use this relation to detect the consistency of the local topological map. When a large number of co-visibility relationships which satisfy the co-visibility ratio is over a certain threshold t(we call them vital frames) exist between the current frame and other surrounding frames, we will start the local consistency correction. We use the mode voting result of all the vital frames' topological node ID to determine which topological node the current frame and those frames belonging to the same topological node with the current frame should be re-assigned to.

2) Global topological nodes merge: Some topological nodes consist of just a few keyframes, some topological nodes are very close to their neighboring topological nodes. In such cases, there may be significant changes of viewpoint or the range of the topological place is very small. However, for navigation tasks and human-computer interaction demands, two topological nodes that are too close do not need to exist at the same time and the topological place with too small range is not so meaningful. Therefore, topological nodes that contain too few keyframes or are too close to other topological nodes will be merged with surrounding topological nodes.

3) Global loop closure: Thanks to the loop closure detection function supported by OpenVSLAM [6], we can further refine the topological map. When loop closure is detected, it

means that the current frame and the candidate frame belong to the same node, then the node to which the current frame belongs needs to be merged with the topology node to which the candidate frame belongs, and the adjacent relationship between nodes should be migrated accordingly.

IV. EXPERIMENT

In order to prove our method is fully feasible and efficient, we tested our method in two real scenarios. In the following part, firstly we will introduce our two test scenarios. Then we will show the overall performance of the algorithm, including the established scene topological map and the quantitative indicators mentioned above. Next, we will show the robustness of our algorithm in the challenging scenarios. Finally, we conduct ablation studies to verify the effect of each component.

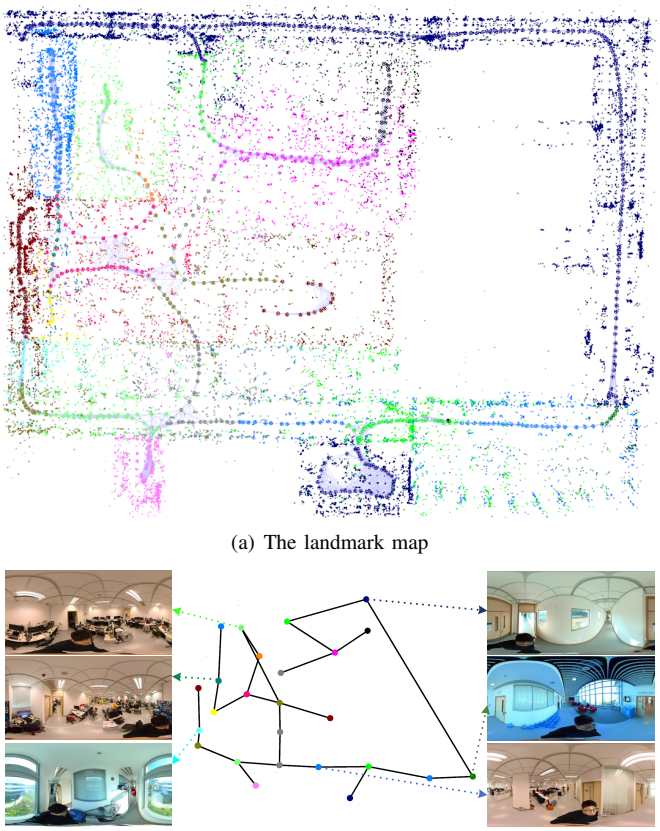
A. Test Scenarios

The first scene is an indoor laboratory. There are meeting rooms, offices, corridors, different work areas, and other topological locations in the laboratory. The objects in the laboratory are placed irregularly and relatively disorderly, and the objects are easy to block each other, which is a challenge to judge the scene change. The second scene is the first floor of the academic building. The scene contains long corridors, multiple office areas, offices, open spaces, and so on. The scale of this scene is large, and there are no representative objects such as doors in many places.

B. Overall Performance

1) Result topological map: We ran our system to reconstruct the topological structure of both scenes. The final result of the landmark map and corresponding topological map of the laboratory are shown in Fig. 4(a) and Fig. 4(b), respectively. The result maps of the academic building are shown in Fig. 1 with the same expression. To clarify, the corridor from the top to the right in the laboratory is considered to be the same topological node, so the location of the topological node will make the structure of the topological map look slightly distorted and puzzling. The colorful points in the landmark map represent the landmarks observed by the keyframes which are represented by spheres. The color of landmarks and keyframes corresponds to the color of the topological node in the topological map. Whenever the agent enters a new region, the algorithm can recognize it is a new node in the sense of topology and establish the connection relationship with adjacent topological nodes. In our academic building, the correspondence between the new topological node identified by the algorithm (as can be seen from the color change of the landmark map) and the new location in the actual scene is shown in Fig. 1.

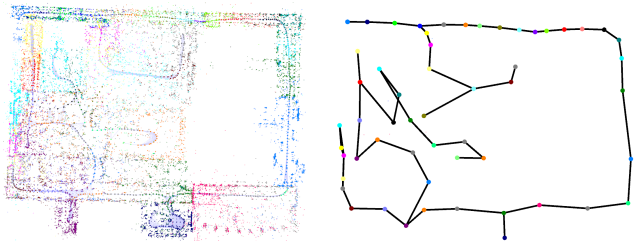
During the process, there are many times the agent returns to a place that it has visited before. The system can accurately identify the same place, namely the same topological node. Fig. 6 gives an example of how the system identified the same place. The red arrow curve marks the test route direction. The red circle marks the location where multiple visits to the same place occurred. The picture in the top right corner is the landmark map with only keyframes. And the bottom right corner is the topological map corresponding to the location. When the agent returns from location B to location A, the frame's color restores from purple to orange, which represents the return to a previous topological place.



(a) The landmark map

(b) The topological map with parts of scene reference keyframes

Fig. 4. The results on the laboratory scene. The topological characteristics are represented by different colors, best viewed in color.



(a) w/o using object detection (b) w/o using object detection

Fig. 5. The results on the laboratory scene are the maps built by 360ST-Mapping without object detection module. It generates lot of redundant topological nodes. The total number of nodes generated is about half of the version without object detection.

2) Quantitative results: We need to clarify several standards which we used to define a real topological node in the test scenes. The following places will be considered as topological node for evaluation.

1) A place that can be accessed through a door.
2) A corridor.

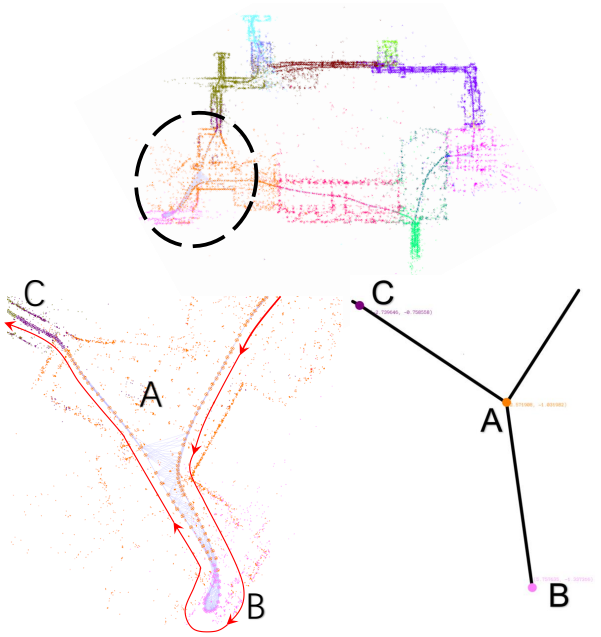


Fig. 6. The system can accurately identify the same topological node when returning to the same place.

3) A place with only one entrance and exit.
4) An intersection connecting above places.

Only the above places should appear in the topological map as topological nodes. We will use TP (true positive, the algorithm takes the place as a topological node, which is right), FP (false positive, the algorithm takes the place as a topological node, which is wrong), FN (false negative, the algorithm thinks the place is not a topological node, which is wrong) and derived accuracy and recall to measure the effectiveness of the algorithm in establishing topological nodes. At the same time, we calculate the total number of topology nodes established by the algorithm, and that should be in the scene. We also calculate the RR(node redundancy rate) to measure the simplification of the topological map. The lower the redundancy, the more refined the topological map.

According to the topological place definition criteria we stated earlier, we quantified the rationality of the topological nodes established in both two scenarios. We test each configuration five times, record the mean and standard deviation of several important index in table I. Our method can achieve high recall and acceptable accuracy in any scene and maintain a low node redundancy rate.

3) Semantic information: The semantic information we assign to each topological node includes the numbers of each kind of object in the scene. We give Fig. 7 and Fig. 8 to show you the object classes distribution among different topological places both in the tested laboratory and academic building scenes.

C. The evaluation of robustness

1) Camera field of view occlusion: In the first test of robustness, we re-run the algorithm in the academic building

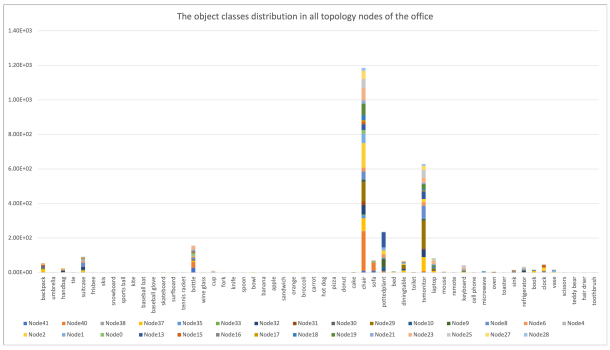


Fig. 7. The object classes distribution of all topological nodes in the laboratory.

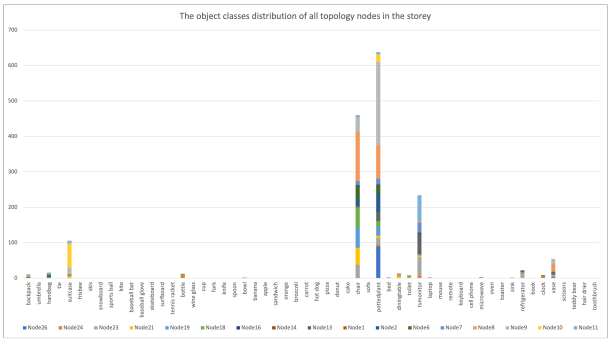


Fig. 8. The object classes distribution of all topological nodes of the academic building.

scene mentioned above and intermittently block half of the vision of our camera during the mapping process (the photo in Fig. 9 shows the camera field of view under interference) to evaluate the mapping performance of the system. The results are shown in Fig. 9. Among the 26 vision changes caused by interference, 7 of them affect the correct judgment of the topological place. The remaining interference has no effect at all. The accuracy and recall rate of establishing topological nodes do not decrease much compared with no interference. Refer to the table I for detailed data. In most locations where algorithms make mistakes, there are few object detection results, which can not be used as a reference. For example, interference 1 and interference 3 marked in the figure are located in the open hall, and interference 10 occurs in the corridor.

2) Huge scene changes: The last experiment we did was to test the recognition ability of the system after significant changes in the scene. We follow the route marked by the arrow in Fig. 10. Our method can correctly identify the same place and extract consistent topological information When the agent revisits the place despite large illumination and appearance variance during mapping processing.

D. Ablation Experiment

Because our system contains many submodules, they will have their own impact on the final effect of establishing the topological map. In order to see the role of these modules more clearly, we did ablation experiments.

TABLE I. Quantitative results of topological map established by various modules. TP, FP, FN represent true positive, false positive, false negative respectively. Pr means precision, Re means recall, NTN means the number of topological nodes built, RTN represents the number of topological nodes in the real scene. RR means redundancy ratio, which is calculated by $(NTN-RTN)/RTN$.

Case	Scene	TP	FP	FN	Pr/σ(%)	Re/σ(%)	NTN	RTN	RR/σ(%)
360ST-Mapping	Laboratory	17	11.8	1	59.4/5.5	94.4/3.9	28.8	18	60/14.4
w/o object detection	Laboratory	17.2	36.8	0.8	32.3/4	95.5/4.6	54	18	200/45.3
w/o co-visibility	Laboratory	16.7	16.8	1.3	50.9/8.6	92.6/4.5	33.5	18	86.1/31.6
w/o merge	Laboratory	17	34.8	1	33/3.2	94.4/6.8	51.8	18	187.8/25
360ST-Mapping	Academic building	12.3	7.5	2.5	62.3/4.5	82.2/3.5	19.8	15	34.9/4
w/o object detection	Academic building	15	32.5	0	31.6/1.2	100/0	47.5	15	216.7/12.7
w/o co-visibility	Academic building	11.8	1	3.2	92.9/8.2	78.7/5.6	12.8	15	-14.7/17
w/o merge	Academic building	13.7	19	1.3	41.9/3	91.1/3.8	32.7	15	117.8/10.2
with interference	Academic building	12.3	9.8	2.8	55.9/5.6	81.7/6.4	22	15	46.7/9.4

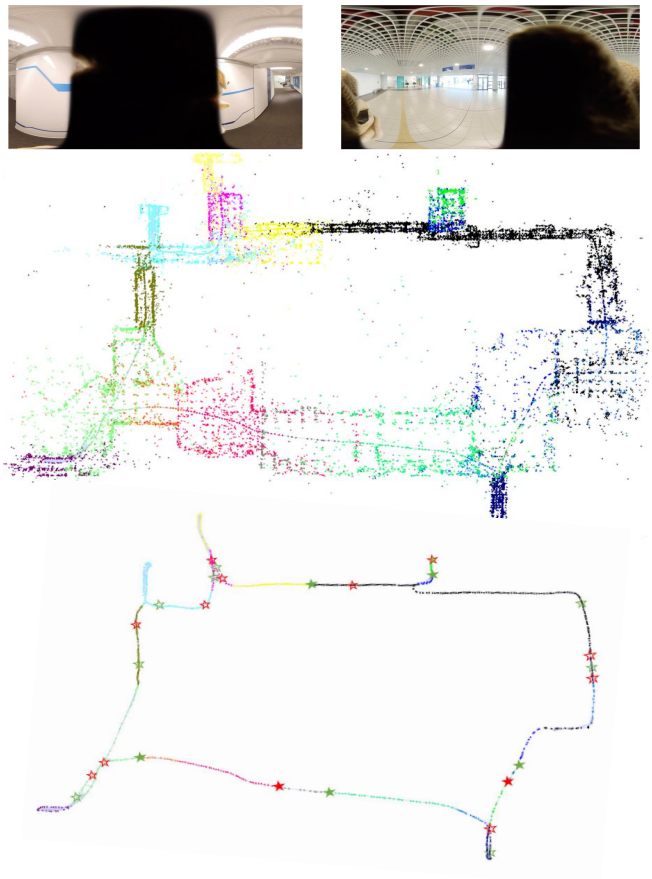


Fig. 9. The mapping result of academic build with temporary disruption. Compared with normal situation, the accuracy and recall rate of establishing topological nodes do not decrease much. The first row shows the 360° images with disruption, while second row is the landmark map. The locations where interference occurs and ends are denoted as green and red star respectively in the corresponding trajectory shown in the last row.

The three variables are whether there is an object detection judgment module, whether there is an online merging module and whether there is a co-visibility judgment module, respectively. If we do not use object detection to assist the topological place recognition process, we can still establish the topological map. The effect is shown in Fig. 5(a) and Fig. 5(b). It is obvious that the topological map built in such way consists of more topological nodes than the "complete" version(Fig. 4(a), Fig. 4(b)). That is exactly one of the reasons why we introduce object detection to the algorithm.

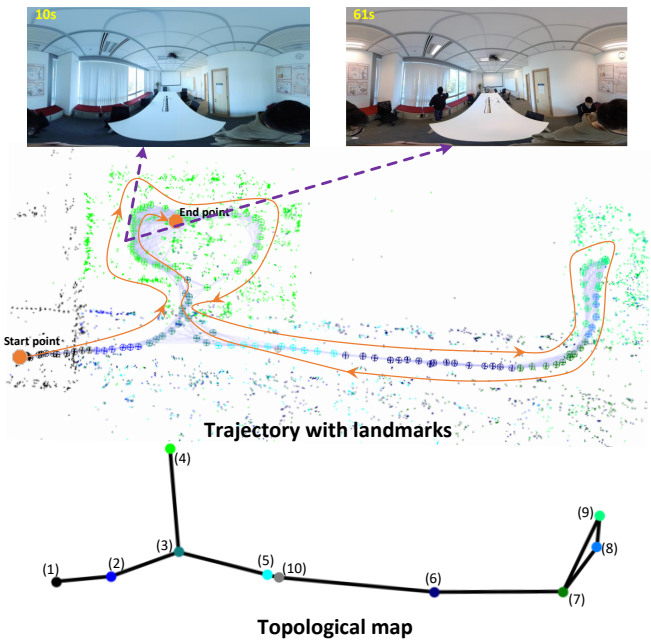


Fig. 10. The system can maintain a consistent topological map when the agent revisit the previous place, even though the previous place undergo obvious change, including illumination and human interaction. The orange line with arrows in the middle row simulate motion flow of the agent.

From table I we can see, when not using object detection, the accuracy of topological node recognition declines seriously and results in a very high node redundancy. The number of nodes established is about twice that with object detection and the accuracy is only half of it. However, its recall rate is slightly higher than that of the version with object detection and has reached 100% in many experiments. it can also be seen that only using co-visibility to judge scene changes has a high recall rate. That is exactly the advantage of the co-visibility relationship in scene change recognition. By using the merging module, the number of nodes can be reduced by 50% without affecting the recall rate of the algorithm, so as to double the accuracy.

When there is no co-visibility judgment module, the performances in the two scenes are different. In the laboratory scenario, the accuracy and recall of topology nodes are lower than that of the complete algorithm. The accuracy is slightly higher than the version without the object detection module.

The node redundancy is also between the version without the object detection module and the complete algorithm. In the academic building, the accuracy is very high, which can reach more than 90%, but the recall rate is slightly lower than the complete algorithm. The node redundancy is a negative value because the number of nodes generated is less than the number of nodes expected in the scene. The reason why the algorithm without co-visibility judgment module performs differently in the two scenes is that the objects in the laboratory scene are messy, cover or interfere with each other seriously, so the object detection module is easy to be disturbed. The building scene is relatively open, the object contour is clear, so it is friendly to the object detection module. However, it is also because the scene is relatively open in the building scene, objects in one place can also be seen in another place, so the recall rate of topology nodes is lower than the complete algorithm.

These results show that it is necessary for us to integrate the co-visibility relationship with the object detection results to conduct the topological node creation task.

V. CONCLUSIONS

In this paper, we present an online mapping approach, 360ST-Mapping, supporting scene topological map reconstruction. We take advantage of robust co-visibility provided by the omnidirectional vision and semantic coefficient measured by weighted Hamming distance to accurately identify the scenes and determine topological node assignment. 360ST-Mapping can be seamlessly integrated into the visual SLAM system without much computational cost. Extensive experiments on large-scale indoor scenarios show that the proposed method can recover a proper topological map and has high robustness of places recognition even though high interference occurs such as dynamic elements and temporary occlusion of perspective.

REFERENCES

[1] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[2] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[3] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[4] F. Blochliger, M. Fehr, M. Dymczyk, T. Schneider, and R. Siegwart, "Topomap: Topological mapping and navigation based on visual slam maps," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 3818–3825.

[5] H. Huang and S.-K. Yeung, "360vo: Visual odometry using a single 360 camera," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[6] S. Sumikura, M. Shibuya, and K. Sakurada, "OpenVSLAM: A Versatile Visual SLAM Framework," in *Proceedings of the 27th ACM International Conference on Multimedia*, ser. MM '19. New York, NY, USA: ACM, 2019, pp. 2292–2295. [Online]. Available: http://doi.acm.org/10.1145/3343031.3350539

[7] D. S. Chaplot, R. Salakhutdinov, A. Gupta, and S. Gupta, "Neural topological slam for visual navigation," 2020.

[8] K. Konolige, E. Marder-Eppstein, and B. Marthi, "Navigation in hybrid metric-topological maps," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 3041–3047.

[9] R. da Rosa, M. Aurelio Wehrmeister, T. Brito, J. L. Lima, and A. I. P. N. Pereira, "Honeycomb map: a bioinspired topological map for indoor search and rescue unmanned aerial vehicles," *Sensors*, vol. 20, no. 3, p. 907, 2020.

[10] D. Dall'Osto, T. Fischer, and M. Milford, "Fast and robust bio-inspired teach and repeat navigation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 500–507.

[11] S. Wen, Y. Zhao, X. Liu, F. Sun, H. Lu, and Z. Wang, "Hybrid semi-dense 3d semantic-topological mapping from stereo visual-inertial odometry slam with loop closure detection," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 16 057–16 066, 2020.

[12] W. Xue, R. Ying, Z. Gong, R. Miao, F. Wen, and P. Liu, "Slam based topological mapping and navigation," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2020, pp. 1336–1341.

[13] C. Gomez, M. Fehr, A. Millane, A. C. Hernandez, J. Nieto, R. Barber, and R. Siegwart, "Hybrid topological and 3d dense mapping through autonomous exploration for large indoor environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 9673–9679.

[14] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool, "Omnidirectional vision based topological navigation," *International Journal of Computer Vision*, vol. 74, no. 3, pp. 219–236, 2007.

[15] M. Liu, D. Scaramuzza, C. Pradalier, R. Siegwart, and Q. Chen, "Scene recognition with omnidirectional vision for topological map using lightweight adaptive descriptors," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 116–121.

[16] E. Garcia-Fidalgo and A. Ortiz, "Hierarchical place recognition for topological mapping," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1061–1074, 2017.

[17] A. Ranganathan and F. Dellaert, "Inference in the space of topological maps: an mcmc-based approach," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 2, 2004, pp. 1518–1523 vol.2.

[18] M. Liu, F. Colas, and R. Siegwart, "Regional topological segmentation based on mutual information graphs," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 3269–3274.

[19] H. Oleynikova, Z. Taylor, R. Siegwart, and J. Nieto, "Sparse 3d topological graphs for micro-aerial vehicle planning," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–9.

[20] Z. He, H. Sun, J. Hou, Y. Ha, and S. Schwertfeger, "Hierarchical topometric representation of 3d robotic maps," *Autonomous Robots*, vol. 45, no. 5, pp. 755–771, 2021.

[21] A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, and L. Carlone, "Kimera: from slam to spatial perception with 3d dynamic scene graphs," 2021.

[22] Z. Ravichandran, L. Peng, N. Hughes, J. D. Griffith, and L. Carlone, "Hierarchical representations and explicit memory: Learning effective navigation policies on 3d scene graphs using graph neural networks," *arXiv preprint arXiv:2108.01176*, 2021.

[23] Z. Zivkovic, B. Bakker, and B. Krose, "Hierarchical map building using visual landmarks and geometric constraints," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 2480–2485.

[24] R. Vazquez-Martin, P. Nunez, A. Bandera, and F. Sandoval, "Spectral clustering for feature-based metric maps partitioning in a hybrid mapping framework," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 4175–4181.

[25] A. Vale and M. I. Ribeiro, "Environment mapping as a topological representation," in *Proceedings of the 11th International Conference on Advanced Robotics-ICAR2003 Universidade de Coimbra, Portugal, June*, 2003, pp. 1–3.

[26] N. Islam, K. Haseeb, A. Almogren, I. U. Din, M. Guizani, and A. Altameem, "A framework for topological based map building: A solution to autonomous robot navigation in smart cities," *Future Generation Computer Systems*, vol. 111, pp. 644–653, 2020.

[27] H. Shin and H. Cha, "Wi-fi fingerprint-based topological map building for indoor user tracking," in *2010 IEEE 16th International Conference on Embedded and Real-Time Computing Systems and Applications*, 2010, pp. 105–113.

[28] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," 2020.