Cooperative Visual Simultaneous Localization and Mapping by Ordering Keyframes Similarity

Jhen-Wei Ruan, Shih-Yu Shih, Meng-Yang Tsai, Yu-Wei Fang, and Cheng-Ming Huang
Electrical Engineering Department, National Taipei University of Technology, Taipei, Taiwan, R.O.C.

***Abstract*— In this paper, multiple moving cameras, which are employed to build the maps individually, are integrated to construct a complete map of the whole large environment. Since multiple cameras may capture the same scenes during performing its own SLAM, the keyframes similarity of each cameras could be evaluated to construct the relation between the map of each camera. The proposed cooperative SLAM algorithm can efficiently build the complete map and reduce the computational time. Through the experiments and analyses on several environments with different size, the amount of feature points, and the size of overlapping area between cameras, the proposed cooperative SLAM with multiple moving cameras can efficiently obtain the information of moving cameras and the environments.**

I. Introduction

With the advance of science and technology nowadays, the demand of simultaneous localization and mapping (SLAM) on a mobile platform is more and more important, whose applications including the automatic guided vehicle in a factory, the service robot in a shopping mall, and the home sweeping, etc. When traditionally building the map of a large environment by only using a single camera [1], the long computational time and the huge data would easily result in the inefficiency of SLAM. Some researchers [2] have proposed the cooperative SLAM with multiple cameras, however, the computational time is still slow such that the results cannot be utilized immediately.

II. Methods and Results

The proposed cooperative SLAM algorithm is based on the ORB-SLAM [1] of a single camera. Figure 1 presents the flowchart of the cooperative SLAM algorithm. Based on the extracted key frames with ORB feature points during the SLAM on each camera, the key frames of all cameras are stored in a key frame database. At every time instant, the newest ten key frames of each camera will be compared with all of the key frames of the other cameras in the key frame database, the pairs of similar key frames are labeled as the candidates of merging maps. The number of the candidates of merging maps in each camera is also ranked as the priority of merging maps. The SLAM map of camera with less candidates of merging maps will be first transferred into the map of other cameras with more candidates. The map of each camera is merged by following the priority of merging maps.

J. W. Ruan, S. Y. Shih, M. Y. Tsai, Y. W. Fang, and C. M. Huang are with the Electrical Engineering Department, National Taipei University of Technology, Taipei, Taiwan, R.O.C. (e-mail: cmhuang@mail.ntut.edu.tw).
This work was supported in part by the Ministry of Science and Technology, Taiwan under the grants: MOST 106-2221-E-027-010.

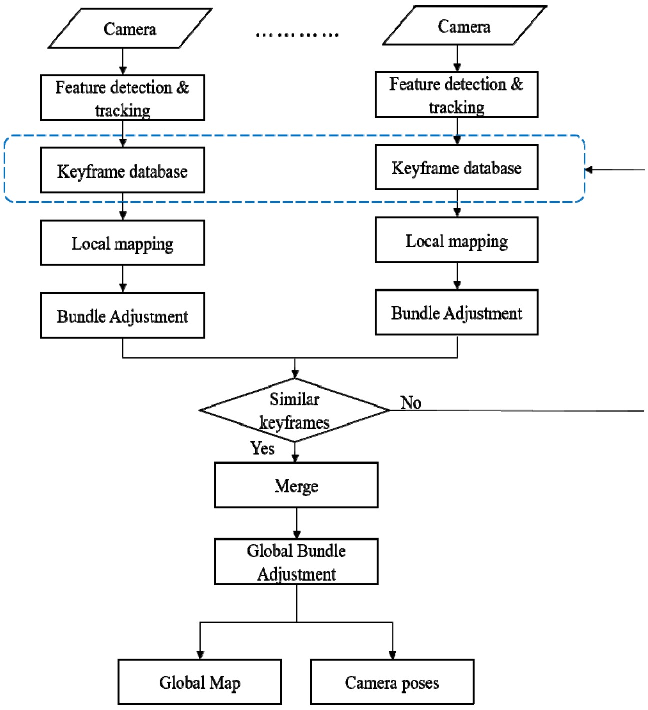

Fig. 1. Flowchart of cooperative SLAM algorithm.

The algorithm for merging the local map of each camera is described as the following steps:

Step 1. Given the keyframe database of all N cameras at current time t, and define K_j^i is the jth keyframe of the ith camera. M_i is the number of keyframes of the ith camera. Initialize the association index $V_{j,h}^{i,k} = 0$, which associates the jth keyframe of the ith camera K_j^i with the hth keyframe of the kth camera K_h^k, for all i, k, j, h.

Step 2. For each camera i, evaluate the ORB feature points similarity $S_{j,h}^{i,k} = ORBMatch(K_j^i, K_h^k)$ of the latest ten keyframes of the ith camera with all of the keframes of the other cameras, *i.e.*, $i = 1,\ldots,N$, $j = t-9,\ldots,t$, $k = 1,\ldots,N$, $h = 1,\ldots,M_k$, $k \neq i$ and $j \neq h$. If the similarity $S_{j,h}^{i,k}$ is larger than a predefined threshold value S_{Thr}, set the association index $V_{j,h}^{i,k}=1$ and record the pair of the similar keyframes in the merging keyframe candidate $D^{i,k}=\left\{\left(K_j^i,K_h^k\right)\mid S_{j,h}^{i,k}>S_{Thr}\right\}$.

Step 3. Generate the merging camera candidate $E=\left\{i \mid N^i>D_{Thr}\right\}$, $i=1,...,N$, where the summation value $N^i=\sum_j\sum_k\sum_h V_{j,h}^{i,k}$.

Step 4. Select the camera p with least similar keyframes, where $p=\arg\min_{i\in E}\left(C_1N^i+C_2M_i/\sum_{i=1}^{N}M_i\right)$, in which C_1 and C_2 are weights. Then, merge the local map of the camera p into the local map of the other cameras based on the merging keyframe candidate $D^{p,k}$, $k=1,...,N$ and $k\neq p$.

Step. 5. Remove the camera p from the merging camera candidate E, *i.e.*, $E = E - p$.

(i) If $E\neq\{\varnothing\}$, goto Step 4.

(ii) If $E=\{\varnothing\}$, then stop and output the merged global map.

Figure 2 presents the merged map of two camera's poses and moving trajectories labeled by blue and red colors. The merged map is transferred from the ROS into the Unity [3] by the JSON (JavaScript Object Notation) to construct the virtual environment for the user experience. The online applications of the merged SLAM map on the path planning and the virtual reality experience are shown in Fig. 3 and 4, respectively.

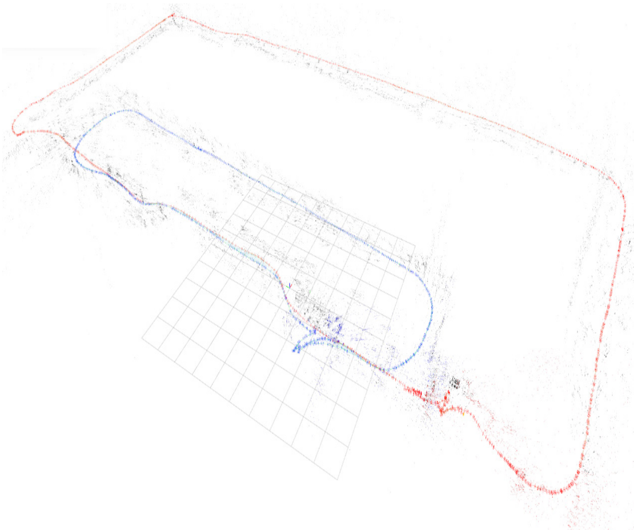

Fig. 2. Merged SLAM map of two cameras.

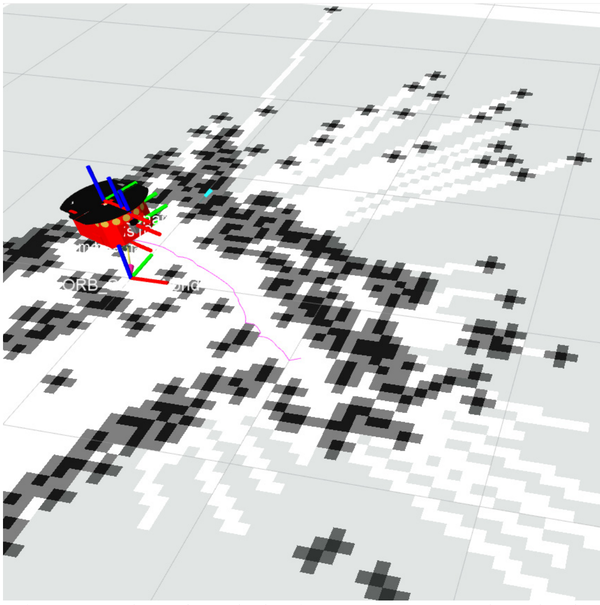

Fig. 3. Merged map for path planning. (Magenta curve: planned path)

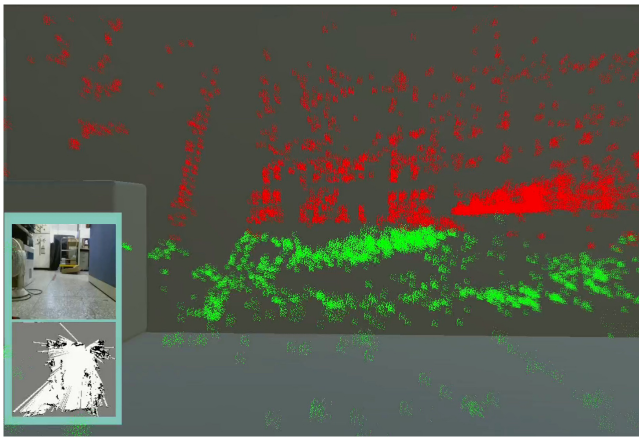

Fig. 4. Merged map for VR experience. The subimages in the left hand side show the current key frame image and the current location on map.

REFERENCES

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

[2] T. M. Bonanni, B. D. Corte, and G. Grisetti, "3-D Map Merging on Pose Graphs," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 1031–1038, Apr. 2017.

[3] "Unity - Game Engine." [Online]. Available: https://unity3d.com/. [Accessed: 09-Jul-2017].