2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

ORBSLAM-Atlas: a robust and accurate multi-map system

Richard Elvira, Juan D. Tardós and J.M.M. Montiel

Abstract— **We propose ORBSLAM-Atlas, a system able to handle an unlimited number of disconnected sub-maps, that includes a robust map merging algorithm able to detect sub-maps with common regions and seamlessly fuse them. The outstanding robustness and accuracy of ORBSLAM are due to its ability to detect wide-baseline matches between keyframes, and to exploit them by means of non-linear optimization, however it only can handle a single map. ORBSLAM-Atlas brings the wide-baseline matching detection and exploitation to the multiple map arena. The result is a SLAM system significantly more general and robust, able to perform multi-session mapping. If tracking is lost during exploration, instead of freezing the map, a new sub-map is launched, and it can be fused with the previous map when common parts are visited. Our criteria to declare the camera lost contrast with previous approaches that simply count the number of tracked points, we propose to discard also inaccurately estimated camera poses due to bad geometrical conditioning. As a result, the map is split into more accurate sub-maps, that are eventually merged in a more accurate global map, thanks to the multi-mapping capabilities.**

We provide extensive experimental validation in the EuRoC datasets, where ORBSLAM-Atlas obtains accurate monocular and stereo results in the difficult sequences where ORBSLAM failed. We also build global maps after multiple sessions in the same room, obtaining the best results to date, between 2 and 3 times more accurate than competing multi-map approaches. We also show the robustness and capability of our system to deal with dynamic scenes, quantitatively in the EuRoC datasets and qualitatively in a densely populated corridor where camera occlusions and tracking losses are frequent.

I. INTRODUCTION

SLAM (Simultaneous Localization and Mapping) algorithms are able to build a map from sensor readings, and simultaneously estimate the sensor localization within the map. Cameras are particularly interesting sensors because of the unique combination of geometry and semantics they provide. In this case, the algorithms are dubbed V-SLAM (Visual SLAM), in this work we focus on the purely visual monocular and stereo sensors. We focus on keyframe and feature point SLAM methods because of their relocalization and place recognition performance, displayed in their capability to build up to city block size maps robustly.

More specifically we build on top of the reference system ORBSLAM [1], [2], [3]. If compared with visual odometry methods [4], [5], [6], [7], [8], ORBSLAM can perform far more accurately especially if the same area is revisited. The ORBSLAM accuracy comes from non-linear bundle adjustment (BA) in which the observations of the same map point come from widely separated keyframes. On the one hand, ORBSLAM is able to robustly detect matches between keyframes even if they are widely separated in time, even in the extreme case of loop closure. ORBSLAM is able to make the most of these abundant high parallax re-observations by an intertwining of elementary mapping stages: ORB matching, DBoW2 place recognition, pose graph optimization, local BA, global BA, and map management. The map management includes creation, deletion, and merging of map points and keyframes. However, it can only handle a single map, which provokes a total failure in exploratory trajectories if tracking is lost, and prevents multi-session mapping.

We propose the ORBSLAM-Atlas system, a generalization of ORBSLAM to the multiple map case. Our main contributions are:

- A multi-map representation that we call *atlas*, that handle an unlimited number of sub-maps. The atlas has a unique DBoWs database of keyframes for all the sub-maps, which allows efficient multi-map place recognition.
- Algorithms for all the multi-mapping operations: new map creation, relocalization in multiple maps, and map merging. We have devised how to interweave the elementary mapping stages to perform the multi-mapping operations robustly, accurately and efficiently. Among all the components of the system, it is relevant the map merging procedure that produces a seamless fusion of two maps with a common region. After the merge, the two merging maps are totally replaced by the new merged map. We propose the creation of a new map after tracking loss. It prevents the failure in exploratory trajectories in which relocalization cannot recover the camera tracking losses.
- A new criteria to declare the tracking lost in the case of poor camera pose observably. It is able to prevent erroneous pose graph optimizations in the loops that contain highly uncertain camera poses.

We provide a quantitative experimental validation in the EuRoC datasets, in which ORBSLAM-Atlas achieves the best results to date for a global map after multiple sessions. In the monocular EuRoC difficult datasets, it greatly improves the coverage and localization error when compared with the single map ORBSLAM. Additionally, the system has proved outstanding robustness in dealing with dynamic scenes.

This work was supported in part by the Spanish government under grants PGC2018-096367-B-I00 and DPI2017-91104-EXP, by the Aragón government under grant DGA_T45-17R, and by Huawei under grant HF2017040003.

The authors are with Instituto de Investigación en ingeniería de Aragón (I3A), Universidad de Zaragoza, Spain richard@unizar.es; tardos@unizar.es; josemari@unizar.es;

II. Related work

In the literature, the multi-map capability has been researched as a component of collaborative mapping systems. The collaborating agents end up sending frames to a central server where the multiple mapping operations are performed. Foster et al. in [9] proposed for the first time this distributed architecture. In their approach, the agents send frames to the global server, however, they do not get information from the server to improve their local maps. The first system with bidirectional information flow, both from the agents to the server and from the server to the agents was C2TAM [10] that is as an extension of PTAM [11] to RGB-D sensors able to handle multiple maps in multiple robots. Morrison et al. in [12] research a robust stateless client-server architecture for collaborative multiple-device SLAM. Their main focus is the software architecture, not reporting accuracy results. The recent work by Schmuck and Chli [13], [14] proposes CCM-SLAM, a distributed multi-map for multiple drones, with bidirectional information flow, built on top of ORBSLAM. Our system is close to their central server because both are built on top of similar elementary mapping stages. They are focused on overcoming the challenge of a limited bandwidth and distributed processing in the monocular case, whereas our focus is building an accurate global map. According to their reported experiments in EuRoC Machine Hall datasets, our system is about 3 times more accurate in the monocular case. Additionally, our system displays robustness, processing accurately all the EuRoC datasets both in stereo and monocular.

The recent ORB-SLAMM [15] also proposes an extension of ORBSLAM2 to handle multiple maps in the monocular case. Their integration of the multiple maps is not so tight as ours, because their sub-maps are kept as separated entities, each having its own DBoW2 database. Additionally, their merge operation computes a link between the sub-maps but does not replace the merging sub-maps by the merged one.

We also compare with VINS-Mono [4] in the multi-session processing of the Machine Hall EuRoC datasets. VINS-Mono is a visual odometry system, in which loop correction is estimated by pose graph optimization. As ORBSLAM-Atlas is able to detect and process with BA numerous high parallax observations, their individual maps are 2 times more accurate than those of VINS-Mono. ORBSLAM-Atlas multi-session global map retains the 2 times higher accuracy over the VINS-mono global map, because thanks to the map merging, it is able to detect and take profit from high parallax matches also in the multi-map and multi-session case.

The idea of adding robustness to track losses during exploration by means of map creation and fusion was firstly proposed by Eade and Drummond [16] within a filtering approach. One of the first keyframe-based multi-map system was [17], where they proposed the idea of disconnected maps, however, the map initialization was manual, and the system was not able to merge or relate the different sub-maps. In the filtering EKF-SLAM approaches, where covariances are readily available, the camera was declared lost with a double criteria of threshold in the number of matches, and low camera localization error covariance [18]. In the keyframe methods, the criterion was reduced just to the number of matches because the covariances are not computed. We propose to recover the double criteria, with a low cost proxy for the camera pose covariance, which comes from the Hessian of the camera-only pose optimization. This approximated covariance has been recently used in [19] for active perception.

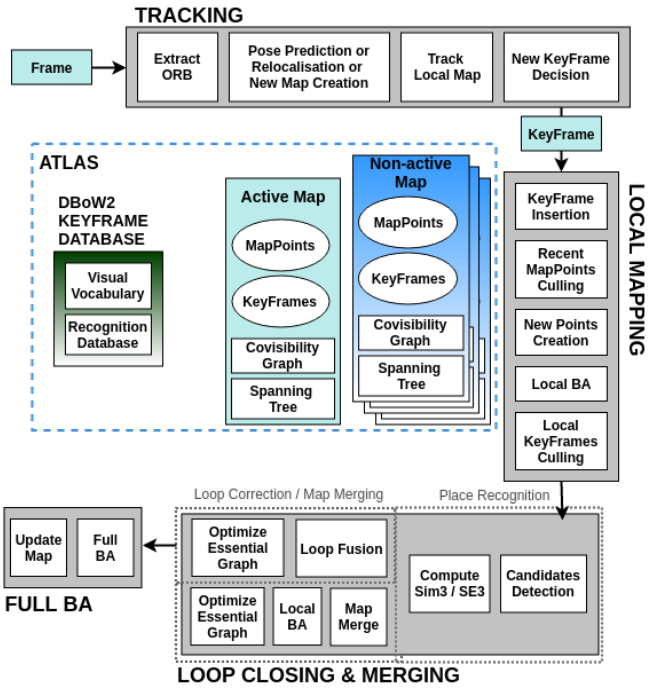


Fig. 1: ORBSLAM-Atlas multi-map representation and workflow.

III. ORBSLAM-Atlas multi-map representation

We call the new multiple map representation *atlas*, from now on, we will use the name *map* to designate each of the atlas sub-maps. Next subsections detail the atlas structure and the criteria to determine when a new map has to be created.

A. Multi-map representation

The atlas (Fig. 1) is composed of a virtually unlimited number of maps, each map having its own keyframes, map points, covisibility graph and spanning tree. Each map reference frame is fixed in its first camera, and it is independent of the other maps references as in ORBSLAM. The incoming video updates only one map in the atlas, we call it the *active map*, we refer to the rest of the maps as *non-active maps*. The atlas also contains a *unique for all the maps* DBoW2 recognition database that stores all the information to recognize any keyframe in any of the maps.

Our system has a single place recognition stage to detect common map regions, if both of them are in the active-map, they correspond to a loop closure, whereas if they are in different maps, they correspond to a map merge.

B. New map creation criteria

When the camera tracking is considered lost, we try to relocalize in the atlas. If the relocalization is unsuccessful for a few frames, the active map becomes a non-active map and is stored in the atlas. Afterwards, a new map initialization is launched according to the algorithms described in [2] and [1].

To determine if the camera is on track, we heuristically propose two criteria that have to be fulfilled, otherwise, the camera is considered lost:

a) Number of matched features: : the number of matches between the current frame and the points in the local map is above a defined threshold.

b) Camera pose observability: : if the geometrical conditioning of the detected points is poor, then camera pose will not be observable and the camera localization estimate will be inaccurate.

Figure 2 displays an example from the Malaga datasets [20], where the usage of the covisibility criteria, combined with the multiple mapping produces a dramatic improvement in the mapping accuracy. A number of points over the threshold are matched in the image, however, they correspond to distant map points, hence the camera translation is estimated inaccurately. Without the observability criterion, the loop closure correction computed by the pose graph optimization is inaccurate due to the poor accuracy of the relative translations included in the loop. Whereas if the observability criterion is used, those uncertain keyframes are removed from the map, the map is fragmented but ORBSLAM-Atlas is able to merge all the sub-maps in an accurate global map.

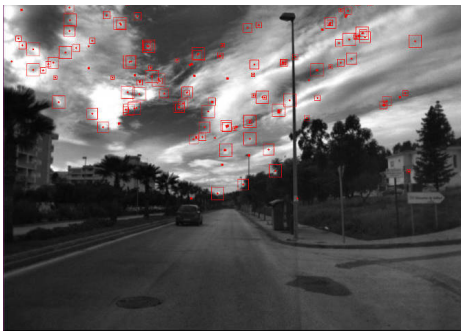
C. Camera pose observability

We estimate the observability from the camera pose error covariance. We assume the map points are perfectly estimated because the real-time operation cannot afford to compute the covariance for the map points per each frame. The measurement information matrix, $\mathbf{\Omega}_{i,j}$, coding the uncertainty for the observation, $\mathbf{x}_{i,j}$, of the map point j in camera i. It is tuned proportional to image resolution scale where the image FAST point has been detected. The uncertainty of the camera i is estimated with the m_i points, where m_i is the number of points in the camera i matched with the map points.

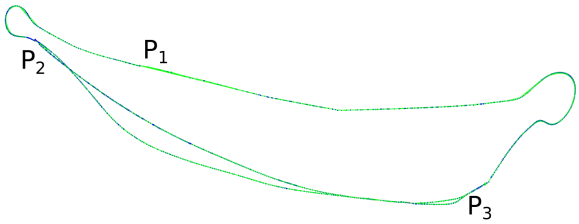
We estimate the 6 d.o.f camera pose as the $\hat{\mathbf{T}}_{i,w} \in \mathrm{SE}(3)$ transformation. Additionally, we code its uncertainty by means of the unbiased Gaussian vector of 6 parameters $\boldsymbol{\varepsilon}_i$ that defines the Lie algebra approximating $\mathbf{T}_{i,w}$ around $\hat{\mathbf{T}}_{i,w}$:

$$
\begin{aligned}
\mathbf{T}_{i,w} &= \mathrm{Exp}\left(\boldsymbol{\varepsilon}_i\right) \oplus \hat{\mathbf{T}}_{i,w} \\
\boldsymbol{\varepsilon}_i &= \begin{pmatrix} x & y & z & \omega_x & \omega_y & \omega_z \end{pmatrix} \sim \mathcal{N}(0, \mathbf{C}_i) \\
\mathbf{H}_i &\simeq \sum_{j=1}^{m_i} \mathbf{J}_{i,j}^{\intercal} \mathbf{\Omega}_{i,j} \mathbf{J}_{i,j} \\
\mathbf{C}_i &= \mathbf{H}_i^{-1}
\end{aligned}
$$

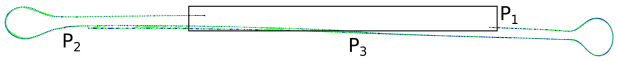
where $\mathrm{Exp} : \mathbb{R}^6 \rightarrow \mathrm{SE}(3)$ directly maps from the parameters space $\boldsymbol{\varepsilon}_i \in \mathbb{R}^6$ to the Lie group $\mathrm{SE}(3)$. The covariance matrix



(a)



(b)



(c)

Fig. 2: Example of mapping accuracy improvement due to observability criterion. (a) Frame where most of the matched points are far from the camera. The number of points criterion is fulfilled but not the observability criterion, and the camera translation is inaccurately estimated. The image corresponds to the region marked as P_1 in the maps below. (b) Camera trajectory without observability criterion. Two loop closures were detected at P_2 and P_3, due to the inaccurate camera poses around P1, the pose graph optimization fails to produce an accurate correction. (c) Camera trajectory with observability criterion. The camera poses in the rectangle in the P_1 region area are excluded. When the low observability region is left, a second map is created. When P_2 is reached, the place recognition fires, and the two maps are merged into a single map. At P_3 a loop closing is detected applying the corresponding correction. The final global map has fewer localized frames but they are more accurate.

$\mathbf{C}_i$ codes the camera estimation accuracy and $\mathbf{J}_{i,j}$ is the Jacobian matrix for the camera pose measurement due to the observation of the map point j in the camera i. As translation is the weakly observable magnitude, we propose to use in the criterion only the $\mathbf{C}_i$ diagonal values corresponding to the translation error:

$$
\begin{aligned}
\max\left(\sigma_x, \sigma_y, \sigma_z\right) &< \sigma_{th}^t \\
\begin{bmatrix} \sigma_x^2 & \sigma_y^2 & \sigma_z^2 & \sigma_{\omega_x}^2 & \sigma_{\omega_y}^2 & \sigma_{\omega_z}^2 \end{bmatrix} &= \mathrm{diag}\left(\mathbf{C}_i\right)
\end{aligned}
\tag{1}
$$

D. Relocalization in multiple maps

If camera tracking is lost, we use the frame to query the atlas DBoW database. This single query is able to find the more similar keyframe in any of the maps. Once we have the candidate keyframe, map, and the putative matched map points, we perform the relocazation following [1]. It includes robustly estimating the camera pose by a first PnP and RANSAC stage, followed by a guided search for matches and a final non-linear camera pose-only optimization.

IV. Seamless map merging

For detecting map merges we use the ORBSLAM place recognition stage. It enforces repeated place recognition for three keyframes connected by the covisibility graph in order to reduce the false positive risk. Additionally, in the merging process, the active map swallows the other map where the common regions have been found. Once the merging is complete the merged map completely replaces the two merging maps. When necessary, we will use the a, s, and m subindexes to refer to the active, swallowed and merged maps respectively.

1) **Detection of common area between two maps**. The place recognition provides two matching keyframes, K_a and K_s and a set of putative matches between points in the two maps M_a and M_s
2) **Estimation of the aligning transformation**. It is the transformation, $\mathrm{SE}(3)$ in stereo or $\mathrm{Sim}(3)$ in monocular, that aligns the world references of the two merging maps. We compute an initial estimation combining Horn method [21] with RANSAC, from the putative matches between M_a and M_s map points. We apply the estimated transformation to K_s for a guided matching stage, where we match points of M_a in K_s, from which we eventually estimate $\mathbf{T}_{W_a,W_s}$ by non-linear optimization of the reprojection error.
3) **Combining the merging maps**. We apply $\mathbf{T}_{W_a,W_s}$ to all the keyframes and map points in M_s. Then, we detect duplicated map points and fuse them, what yields map points observed both from keyframes in M_s and M_a. Afterwards, we combine all M_s and M_a keyframes and map points into M_m. Additionally, we merge the M_s and M_a spanning trees and covisibility graphs into the spanning tree and covisibility graph of M_m.
4) **Local BA in the welding area**. It includes all the keyframes covisible with K_a according to M_m covisibility graph. To fix the gauge freedoms the keyframes that were fixed in M_a are kept fixed in the local BA, whereas the rest of the keyframes are set free to move during the non-linear optimization. We apply a second duplicated point detection and fusion stage updating the M_m covisibility graph.
5) **Pose graph optimization**. Finally, we launch a pose graph optimization of M_m.

The merging runs in a thread in parallel with the tracking thread, the local mapping thread, and occasionally a global bundle adjustment thread (Fig.1.) Before starting the merging, the local mapping thread is stopped to avoid the addition of new keyframes in the atlas. If a global bundle adjustment thread is running, it is also stopped because the spanning tree on which the BA is operating is going to be changed. The tracking thread is kept running on the old active map to keep the real-time operation. Once the map merging is finished, we resume the local mapping thread. The global bundle adjustment, if it has been stopped, is relaunched to process the new data.

V. Experiments

The quantitative evaluation has been made in the EuRoC datasets [22]. To score the results we compute the RMS ATE (Absolute Translation Error) in meters for all the frames in the sequences as proposed in [23]. To factor out the non-deterministic nature of the multi-threading execution, we run each experiment 5 times and report the average or median values. The qualitative evaluation was done in monocular for a hand-held camera traversing a densely populated corridor where occlusions and tracking losses are frequent. For a general overview of the experiments see the accompanying video.

A. Multiple map performance

We focus our quantitative evaluation on the EuRoC V1_03_difficult and V2_03_difficult datasets because ORB-SLAM2 stereo [2] or ORBSLAM monocular [3] reported them as failure due to a coverage below 90 %. Coverage is defined as the fraction of localized frames with respect to the total number of ground truth frames in the dataset. The differences in performance in the rest of the datasets are negligible because ORBSLAM-Atlas never lost track, and hence never used more than a single map.

Table I reports the quantitative comparison, see also Figure 3. We have made new experiments with ORBSLAM to report both the RMS ATE and the coverage. Thanks to the multi-maps, ORBSLAM-Atlas is able to significantly boost the coverage from 10-15 % to 70-90 %, with an RMS ATE lower than ORBSLAM.

In the stereo case, in V1_3 the differences between ORB-SLAM2 and ORBSLAM-Atlas are negligible. In contrast, in V2_3 ORBSLAM-Atlas produces 5 intermediate maps that eventually are merged in a global map able to achieve around 95 % coverage and an RMS ATE lower that ORBSLAM2.

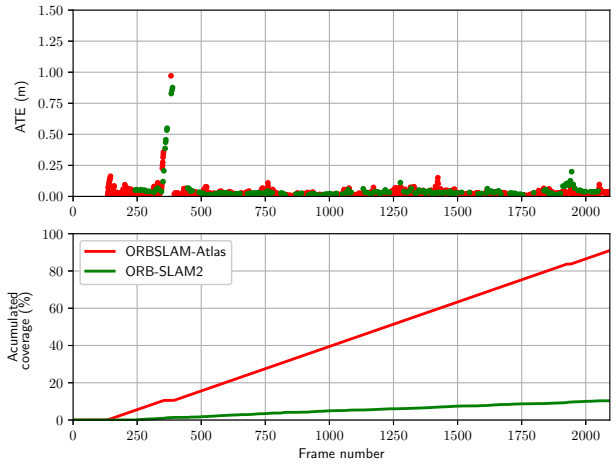
B. Multi-session performance

Table II displays the RMSE ATE for all the datasets in EuRoC, which are processed individually. We also report the global multi-session map after processing the five Machine Hall datasets (MH_01 to MH_05) sequentially for ORBSLAM-Atlas and VINS-Mono. For VINS-Mono and VINS-Stereo we verbatim quote the values reported by the authors in [4], [5]. Trajectories have been aligned by means of $\mathrm{SE}(3)$ transformations.

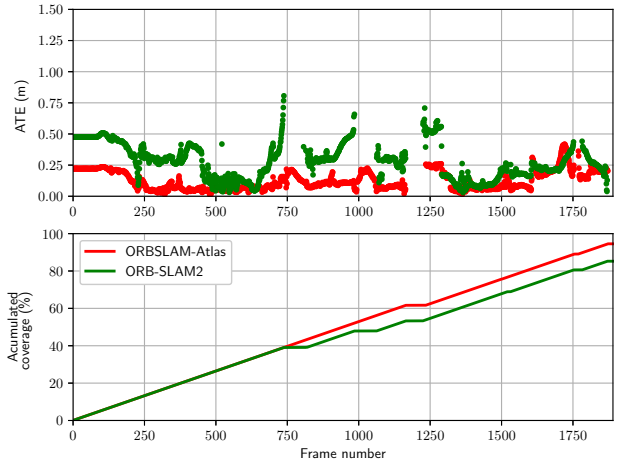
We can conclude that our individual session maps are more accurate than those of VINS-Mono or VINS-Stereo. We

	ORBSLAM-Atlas Monocular			ORBSLAM Monocular			ORBSLAM-Atlas Stereo			ORBSLAM2 Stereo		
	ATE (m)	Cover (%)	# Maps	ATE (m)	Cover (%)	# Maps	ATE (m)	Cover (%)	# Maps	ATE (m)	Cover (%)	# Maps
V1_03	**0.106**	**90.74**	2	0.132	10.32	1	0.051	100	1	**0.046**	100	1
V2_03	**0.093**	**70.74**	2	0.146	15.71	1	**0.218**	**94.55**	5	0.316	89.21	1

TABLE I: Performance on the difficult Vicon Room EuRoC datasets. RMS ATE in meters. Median values after 5 runs.



(a) V1_03 in monocular



(b) V2_03 in stereo

Fig. 3: ATE (m) per each localized frame in the sequence, and accumulated coverage (%). Out of the 5 runs, it is represented the one that gets the median RMS ATE. Best viewed in color.

conjecture that ORBSLAM-Atlas can detect numerous high parallax observations and process them with non-linear BA, and hence is more accurate. The same accuracy advantage between ORBSLAM-Atlas and VINS-Mono is retained in the multiple session case, what proves that ORBSLAM-Atlas is able to detect and exploit the high parallax matches also among the multiple maps, and in the multiple session operation.

In table III, we compare with respect to CCM-SLAM[13], [14], which is a centralised collaborative monocular SLAM

	ORBSLAM-Atlas stereo	VINS stereo	VINS Mono Inertial
V1_01	**0.036**	0.550	0.068
V1_02	**0.022**	0.230	0.084
V1_03	**0.051**	X	0.190
V2_01	**0.034**	0.230	0.081
V2_02	**0.028**	0.200	0.150
V2_03	**0.218**	X	0.220
MH_01	**0.036**	0.540	0.120
MH_02	**0.021**	0.460	0.120
MH_03	**0.026**	0.330	0.130
MH_04	**0.103**	0.780	0.180
MH_05	**0.054**	0.500	0.210
muliple-session MH_01-MH_05	**0.086**	-	0.210

TABLE II: Multiple-session performance on EuRoC datasets. We report the results of the individual mapping sessions, and the global multi-session map after the sequential processing of datasets MH_01 to MH_05. Reported RMS ATE (m) are median values after 5 runs.

	Global map RMS ATE (m)
CCM-SLAM (Mono*)	0.077
ORBSLAM-Atlas (Mono*)	**0.024**
ORBSLAM-Atlas (Stereo)	0.035

TABLE III: RMS ATE (m) in the EuRoC Machine Hall (MH_01, MH_02 and MH_03). * indicates that the aligning transformation prior to ATE computation includes a scale correction. The reported values are the average after 5 runs to make them comparable with results reported in [14].

system where the agents compute a local map and send frames to the central server in order to build a global map. In the experiment reported in their paper, CCM-SLAM is launched with three agents, each of them processes, in parallel, a sequence of the EuRoC Machine Hall experiment (MH_01, MH_02 and MH_03), and the server processes all the information from the three sequences in the global map. The reported RMS ATE is computed with respect to the ground truth after a Sim(3) alignment. We verbatim quote the values as reported by the authors in [14]. We have processed the MH_01, MH_02 and MH_03 datasets sequentially in a multi-session manner with ORBSLAM-Atlas to obtain a global map. We have made the monocular mapping with the corresponding Sim(3) alignment. We have also made the stereo mapping, hence we can recover the scale, and report the RMS ATE after SE(3) alignment. We can conclude that our global map is more accurate than CCM-SLAM in the monocular case. Additionally, the stereo case also shows better accuracy with the advantage that we estimate the scene real scale.

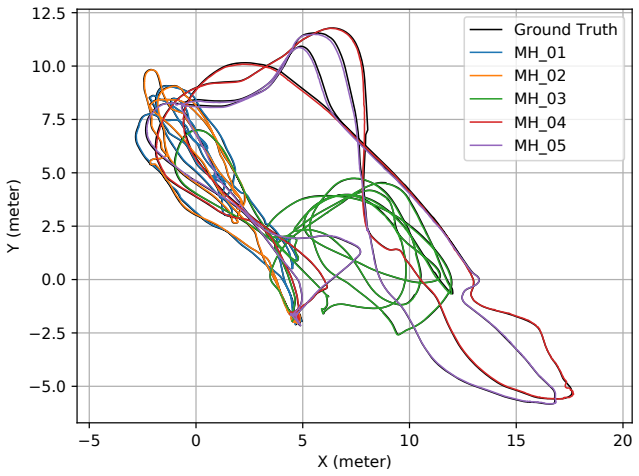


Fig. 4: Trajectories after processing Machine Hall datasets MH_01-MH_05 sequentially as multiple sessions with ORBSLAM-Atlas stereo (top view). Aligned with ground truth by means of global SE(3) transformation. Best viewed in color.

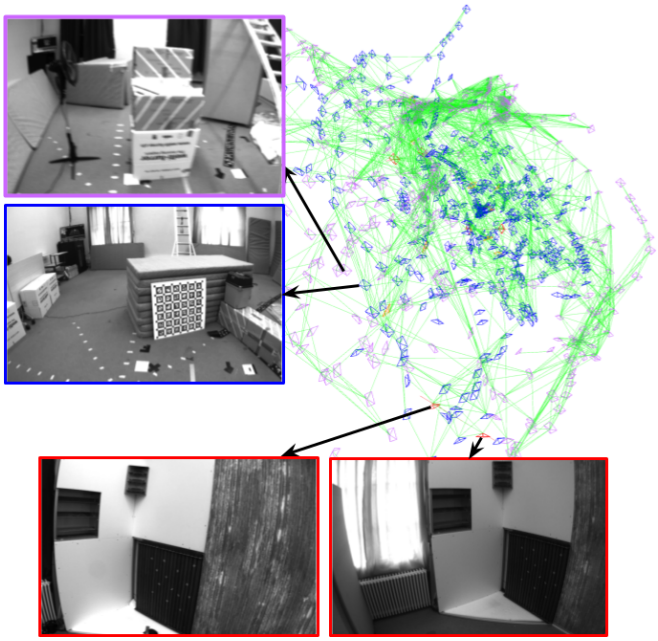


Fig. 5: Keyframes of the Vicon room global map. The map contains the two experiences corresponding to the two versions of the room. All the keyframes of the global map are displayed, the purple keyframes correspond to V1_XX, the blue ones to V2_XX. Two keyframes close in space but corresponding to different experiences are displayed at top left corner. The two bottom keyframes corresponds to the merging keyframes.

C. Mapping in dynamic scenes

In the accompanying video, we provide a qualitative evaluation in a fast dynamic scene, in which a monocular hand-held camera images a densely populated environment. ORBSLAM-Atlas is able to produce a global map for the whole plant corridor. Several intermediate maps have been spawned to survive to camera tracking losses.

To provide quantitative evaluation, we have processed the whole EuRoC dataset in a multi-session manner, feeding the 11 stereo videos in sequence: MH_01, MH_02, MH_03, MH_04, MH_05, V1_01, V1_02, V1_03, V2_01, V2_02, V2_03, without providing any additional information to the system. After the 11 sessions, the system has been able to identify three different maps. The first map corresponds to the five sequences of the Machine Hall. The second map corresponds to V1_01, V1_02, V1_03, V2_01 and V2_02. Experiments V1_XX and V2_XX were grabbed in the same room, however experiments V2_XX were made 112 days later than V1_XX, the distribution of the furniture was changed, and the ground truth reference was moved as well. Our system is able to merge the maps corresponding to the two versions of the room because of the common elements, which mainly correspond to the floor and the elements fixed to the walls, such as the door, the windows or the radiators. The third map corresponds to sequence V2_03 that, due to the fast camera motion, our system is unable to merge with the second map.

The merged map of the Vicon room is interesting because it displays the lifelong capabilities of our system. The same map is able to jointly consider the two different experiences of the same room. There are some pairs of keyframes localized close to each other in the map, however they image two different version of the room (see Fig. 5), and are not connected in the covisibility graph. Thanks to the accuracy of the place recognition and the feature matching, the system never gets confused with the different versions of the room, but reuses the keyframes when the camera observes common scene areas. In Table IV we report the global map error, and the map size in terms of the number of keyframes and the number of map points. In the case of the Machine Hall, there is a reduction in the number of keyframes (82 %) and keypoints (52 %) of the global map with respect to the individual maps. The reduction is proportional to the common areas between the maps (see Fig. 4). In the case of the Vicon room, this reduction is only slightly smaller (89 % for KF and 60 % for KP) despite the drone trajectories are close to each other. There is no bigger reduction because the global map has to represent the two versions of the room. The global reference for the ground truth in the two rooms was different, for this reasons, to compute the RMS ATE we have made two SE(3) alignments, one for the V1 room frames and other to the V2 frames.

D. Computing Time

We have evaluated our ORBSLAM-Atlas algorithm in an Intel Core i7-7700 (four cores @ 3.6 GHz) desktop computer with 32GB RAM. We focus on the V2_03 EuRoC dataset in stereo, the frame rate is 20 Hz. We can achieve real time in the tracking thread with an average processing time of $\approx$ 42 ms. The local mapping, running in a parallel thread, typically consumes $\approx$ 78 ms per keyframe. Place recognition

Dataset	# KF	# MP	RMSE ATE (m)
MH_01	481	10,199	0.035
MH_02	430	16,504	0.018
MH_03	442	19,947	0.028
MH_04	316	18,943	0.119
MH_05	373	21,203	0.060
Total Size	2,042 (100 %)	86,796 (100 %)	-
MH_01+MH_02+MH_03+ MH_04+MH_05	1,666 (82 %)	45,660 (53 %)	0.086
V1_01	112	7,610	0.035
V1_02	145	9,682	0.020
V1_03	228	13,291	0.048
V2_01	109	7,902	0.037
V2_02	292	16,081	0.035
Total Size	886 (100 %)	54,566 (100 %)	-
V1_01+V1_02+V1_03+ V2_01+V2_02	791 (89 %)	32,920 (60 %)	0.040
V2_03	270	13,683	0.218

TABLE IV: Multiple-map in a dynamic scene. ORBSLAM-Atlas stereo identifies 3 different maps. Comparison of the individual session mapping with respect to the multi-session mapping. Median values after 5 runs.

takes ≈ 10 ms to compute the aligning transformation and map merging takes ≈ 670 ms. In any case, as map merging runs in a parallel thread, it does not interfere the real-time tracking thread. Tracking operates on the unmerged map until merging is finished, and then the unmerged map is substituted by the merged one.

VI. Conclusions

We have presented ORBSLAM-Atlas a multi-map system able to bring the outstanding qualities of the single map ORBSLAM to the multiple map arena. It is able, not only to robustly detect wide-baseline matches between the sub-maps but also, to include them in the subsequent non-linear optimizations to yield accurate estimations for the cameras and the map. The resulting multi-map system is more robust because it is able to survive to the tracking losses in exploratory trajectories, and more general because it naturally can handle multi-session operation.

The experimental validation in the EuRoC datasets has revealed that ORBSLAM-Atlas can report the best results to date for a global map after multi-sessions, and for the coverage and error in the EuRoC difficult datasets sing purely monocular vision. Additionally, the system has proved outstanding robustness in dealing with dynamic scenes.

References

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[2] R. Mur-Artal and J. 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.

[3] ——, "Visual-inertial monocular SLAM with map reuse," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 796–803, 2017.

[4] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[5] T. Qin, S. Cao, J. Pan, and S. Shen, "A general optimization-based framework for global pose estimation with multiple sensors," *arXiv preprint arXiv:1901.03642*, 2019.

[6] J. Delmerico and D. Scaramuzza, "A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2502–2509.

[7] 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, 2018.

[8] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "SVO: Semidirect visual odometry for monocular and multicamera systems," *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 249–265, 2017.

[9] C. Forster, S. Lynen, L. Kneip, and D. Scaramuzza, "Collaborative monocular SLAM with multiple micro aerial vehicles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 3962–3970.

[10] L. Riazuelo, J. Civera, and J. Montiel, "C2TAM: A cloud framework for cooperative tracking and mapping," *Robotics and Autonomous Systems*, vol. 62, no. 4, pp. 401–413, 2014.

[11] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, 2007, pp. 225–234.

[12] J. G. Morrison, D. Gálvez-López, and G. Sibley, "MOARSLAM: Multiple operator augmented RSLAM," in *Distributed autonomous robotic systems*. Springer, 2016, pp. 119–132.

[13] P. Schmuck and M. Chli, "Multi-UAV collaborative monocular SLAM," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 3863–3870.

[14] ——, "CCM-SLAM: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams," *Journal of Field Robotics*, vol. 36, no. 4, pp. 763–781, 2019.

[15] H. A. Daoud, A. Q. M. Sabri, C. K. Loo, and A. M. Mansoor, "SLAMM: Visual monocular SLAM with continuous mapping using multiple maps," *PloS one*, vol. 13, no. 4, 2018.

[16] E. Eade and T. Drummond, "Unified loop closing and recovery for real time monocular SLAM," in *Proc. 19th British Machine Vision Conference (BMVC)*, Leeds, UK, September 2008.

[17] R. Castle, G. Klein, and D. W. Murray, "Video-rate localization in multiple maps for wearable augmented reality," in *12th IEEE International Symposium on Wearable Computers*, Sept 2008, pp. 15–22.

[18] B. Williams, G. Klein, and I. Reid, "Real-time SLAM relocalisation," in *IEEE 11th International Conference on Computer Vision (ICCV)*, 2007, pp. 1–8.

[19] Z. Zhang and D. Scaramuzza, "Perception-aware receding horizon navigation for MAVS," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2534–2541.

[20] J.-L. Blanco-Claraco, F.-A. Moreno-Dueñas, and J. González-Jiménez, "The Málaga urban dataset: High-rate stereo and LIDAR in a realistic urban scenario," *The International Journal of Robotics Research*, vol. 33, no. 2, pp. 207–214, 2014.

[21] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," *JOSA A*, vol. 4, no. 4, pp. 629–642, 1987.

[22] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[23] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 573–580.