AirMuseum: a heterogeneous multi-robot dataset for stereo-visual and inertial Simultaneous Localization And Mapping

Rodolphe Dubois[1,2], Alexandre Eudes[1], Vincent Frémont[2]
[1] DTIS, ONERA, F-91123 Palaiseau, France
[2] Centrale Nantes, LS2N, UMR 6004, Nantes, France
E-mail: [1] firstname.lastname@onera.fr, [2] firstname.lastname@ls2n.fr

***Abstract*— This paper introduces a new dataset dedicated to multi-robot stereo-visual and inertial Simultaneous Localization And Mapping (SLAM). This dataset consists in five indoor multi-robot scenarios acquired with ground and aerial robots in a former Air Museum at ONERA Meudon, France. Those scenarios were designed to exhibit some specific opportunities and challenges associated to collaborative SLAM. Each scenario includes synchronized sequences between multiple robots with stereo images and inertial measurements. They also exhibit explicit direct interactions between robots through the detection of mounted AprilTag markers [1]. Ground-truth trajectories for each robot were computed using Structure-from-Motion algorithms and constrained with the detection of fixed AprilTag markers placed as beacons on the experimental area. Those scenarios have been benchmarked on state-of-the-art monocular, stereo and visual-inertial SLAM algorithms to provide a baseline of the single-robot performances to be enhanced in collaborative frameworks.**

***Index Terms*— Robotics, Visual SLAM, Multi-Robot SLAM**

I. Introduction

In mobile robotics, many tasks such as exploration, inspection or navigation in an unknown environment, heavily depend on the ability of the robots to localize themselves and estimate their whole trajectory. In outdoor environments, Global Positioning Systems (GPS) can be used to achieve accurate localization. However, such systems are intrinsically restricted to metric accuracy while some applications, like navigation in cluttered areas, may require up to decimetric or centimetric accuracies. Furthermore and overall, GPS signals are not suitable for localization in indoor environments as they may be mitigated and blocked by walls and roofs. Robots must hence rely on their own embedded sensors. A first family of methods, known as Dead-Reckoning, proceed by integrating measurements from proprioceptive sensors. However, they are likely to accumulate drift depending on the quality of the sensors. Reversely, Simultaneous Localization And Mapping (SLAM) algorithms concurrently estimate the 3D structure of the observed environment and the trajectory within it. They require at least one exteroceptive sensor like LiDARs [2], or cameras [3] for Visual SLAM (VSLAM), those last ones being cheaper, lower power consuming and easier to embed on drones. In the specific case of visual-inertial (VI) SLAM, cameras are enhanced with an additional Inertial Measurement Unit (IMU) [4], [5]. Nowadays, such methods have achieved high maturity with a wide variety of single-robot SLAM algorithms which have been issued [6].

Fig. 1: Screenshots of inter-robot direct observations between the drone and ground robots outfitted with AprilTag markers

A current challenge to the SLAM community is Multi-Robot SLAM (MR-SLAM). It brings new opportunities to enhance the efficiency of a fleet of robots performing SLAM and increase the estimation accuracy of its agents. Each MR-SLAM algorithm combines three key ingredients: i) a task and data allocation scheme, ii) an adapted communication policy and iii) a matching & merging strategy. The allocation scheme governs how each task and data is centralized / decentralized or distributed. The communication policy supervises the topology, the planning and the content of inter-robot exchanges. Finally, the matching & merging strategy focuses on spotting inter-robot correspondences (data association) and jointly refining the map and trajectories of the robots (data fusion). However, MR-SLAM also comes with new challenges stemming from i) the network and time synchronization constraints; ii) the limited computational resources the robots must rely on to process their own data additionally to multi-robot interactions; and iii) heterogeneous data fusion.

Multiple visual MR-SLAM algorithms have been issued over the last decade. A few examples of fully centralized MR-SLAM architectures are *C^2TAM* [7], *CSfM* [8], *CCM-SLAM* [9] which was then extended by *CVI-SLAM* [10] for VI-SLAM. Centralized methods mainly differ w.r.t. the degree of autonomy they grant to the agents. Regarding decentralized MR-SLAM, contributions include full SLAM pipelines like *DDF-SAM* [11], stereo SLAM [12] and VI-SLAM [13], [14] methods. Decentralization is also more conducive to the distribution of tasks and data to compensate for the lack of a central server as agents should process the data received from all the other agents. Hence, decentralized MR-SLAM also covers methods for the distributed detection

of inter-robot correspondences [15], [16], distributed global inference [17] and distributed map storage [18].

Performances of SLAM algorithms are usually benchmarked on available public datasets, which can be classified according to: i) the nature of their data (real vs. synthetic); ii) the used robotic platforms and sensor suites; iii) the properties of the environment in which one they were acquired and the nature of their trajectories (e.g. smooth vs. aggressive motions); and iv) their target applications (odometry, mapping, single-robot SLAM, multi-session mapping, collaborative localization or mapping, MR-SLAM). Regarding the first criteria, synthetic data simulators like *Gazebo* [19] have tremendously improved over the last years to get photo-realistic environments. Fully synthetic datasets have been recently published like the *BlackBird* [20] and *TartanAir* [21] datasets. Nevertheless, most available datasets were built on real data and various kinds of environments have been covered: industrial (*EuRoC* [22]), underwater (*Aqualoc* [23]), urban (*KITTI* [24] and *Malaga* [25]), underground (*Chilean mines* [26]) and emulated extra-planetary environments (*Robex* [27], *Beach Planetary Rovers* [28] and *Canadian Rover Navigation* dataset [29]). Some datasets are explicitly dedicated to specific applications e.g. handling aggressive motions like the *UZH-FPV drone racing* dataset [30] or multi-session mapping like the *Zurich Urban dataset* [31] in which one several trajectories were acquired with drones in Zurich city center. Some of them, being extensively used for benchmarking, have deeply impacted the research on SLAM algorithms, such as *KITTI* [24] and *EuRoC* [22] for V-SLAM and VI-SLAM.

Very few published datasets are dedicated to the specific benchmarking of MR-SLAM algorithms. To the best of our knowledge, the only one which was purposively intended for MR-SLAM is the *UTIAS* dataset [32]. It makes 5 robots, outfitted with a monocular camera, explore a $6\,\mathrm{m} \times 12\,\mathrm{m}$ area beaconed with uniquely identifiable landmarks. Besides, datasets for multi-session mapping – i.e. whose sequences were acquired in the very same environment – can also definitely be used to evaluate such algorithms by synchronizing the trajectories beforehand. However, the resulting scenarios may not be suitable to relevantly assess MR-SLAM algorithms. We believe that the trajectories of a MR-SLAM dataset should be designed such that the individual estimation accuracies of the robots can be significantly enhanced if the evaluated MR-SLAM algorithm succeeds in taking the above mentionned opportunities and in overcoming the associated challenges. Trajectories of a multi-robot scenario should be jointly designed according to: i) the expected drift accumulated along individual trajectories because of their kinematics or the properties of the observed environment; ii) the spatial and temporal distribution of intra/inter-robot correspondences and the impact of the resulting inter-robot loop closures on the accuracies of the trajectory estimates, and iii) the ability to stage challenges arising from network and communication issues.

In this paper, we propose a new dataset, which is intended for heterogeneous Multi-Robot stereo-visual and inertial Simultaneous Localization And Mapping. It includes five scenarios, which were acquired using three ground robots and one drone. The acquisition site is a former Air Museum which is now a large indoor industrial-like warehouse, at ONERA Meudon, France. We provide full calibration for the visual and inertial sensors, as well as the ground-truth trajectories, computed using Structure-From-Motion (SfM) algorithms. The rest of the paper is organized as follows. Section II describes the acquisition area and the robotic platforms, gives an overview of the dataset and describes the motivations behind each scenario. Section III provides additional details on the robots' specifications. Section IV describes the computation of the ground-truth trajectories. In Section V, the properties of the individual sequences are studied by evaluating the single-robot performances for monocular, visual-inertial and stereo-visual setups on state-of-the-art SLAM algorithms. Finally, in section VI, we detail how the provided data is formatted. The dataset will be hosted on IEEE Data Port[1], and associated information will be found on the following *Github* repository[2].

II. Dataset overview

A. Acquisition site

The experimental area is a large hangar which was once part of the Chalais-Meudon Air Museum from 1921 to 1977 and which was then moved to the Bourget *Musée de l'Air et de l'Espace*. Nowadays, it is a large $40\,\mathrm{m} \times 80\,\mathrm{m}$ warehouse, an overview of which one is given by Figure 2. Figure 3 additionally provides a map of the site.

(a) North part (b) South part

Fig. 2: Overview of the acquisition site. Most sequences were acquired in the South area which is more populated with textured structures.

The site is divided into two parts. The North part, which is pictured by Figure 2a, is a vast and empty area. Most of the trajectories were captured in the South part represented by Figure 2b. This area is filled with more obstacles and with richer textures. The site was also beaconed with five AprilCubes, each one consisting of five AprilTag markers pointing towards all observable directions, as shown in Figure 4a. They can be used to initialize common reference frames between the robots, but they were primarily intended to provide additional constraints on the ground-truth trajectories computed from Structure-From-Motion algorithms, as detailed in section IV.

[1] https://ieee-dataport.org/

[2] https://github.com/AirMuseumDataset/AirMuseumDataset.git

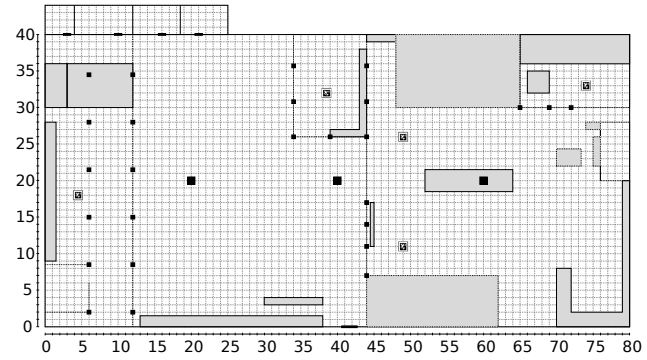
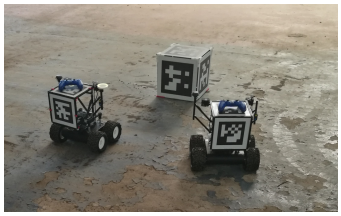


Fig. 3: Plan of the experimental area with metric scale. Gray zones denote inaccessible encumbered areas while the black squares represent pillars. The locations of AprilCubes are denoted using small QR code symbols.

B. Robotic platforms

The dataset was acquired using three ground robots, respectively referred to as A, B and C, and one drone (D), as shown by Figure 4. Each robot is outfitted with a stereo camera bench and an Inertial Measurement Unit (IMU). Robots B and C have mounted AprilTag markers [1] which are used for direct inter-robot observations. More extensive details about the specifications of the robots and the multi-robot setup are provided in section III.



(a) Wifibot™ robots B and C

(b) DJI-M100™ drone

Fig. 4: Used robotic platforms. AprilTag markers [1] are mounted on robots B and C and are used to estimate relative poses from direct inter-robot observations.

C. Multi-robot scenarios

1) Followed guidelines to design multi-robot scenarios: This dataset consists in five multi-robot scenarios whose motivations and design guidelines are detailed below. As we mentioned earlier, multi-robot SLAM brings new opportunities and challenges to enhance the estimation accuracies of the robots. In each scenario, we jointly designed the trajectories of the robots following three guideline principles.

The first guideline was to make the individual trajectories prone to drift accumulation in such a way that the estimation accuracies of the robots can be significantly improved when sharing information between the robots. The accumulated drift depends i) on the trajectory kinematics since aggressive motions and rough rotations may undermine standard VIO algorithms; ii) on the visual richness of the observed environment (e.g. textures, number, salience and conditioning of visual features) and iii) on the distribution of loop closures within the trajectories.

The second guideline was to design a temporal and spatial distribution of inter-robot direct and indirect correspondences which generates informative inter-robot loop closures. An inter-robot loop closure is all the more informative as it covers and constraints a large number of keyframes subjected to significant drift within that loop. Note that such inter-robot loop closures may bring asymmetric information to the involved robots depending on the disparities of information on their respectively covered sub-trajectories. The general idea was to workout a distribution of intra/inter-robot correspondences which makes the trajectories complementary.

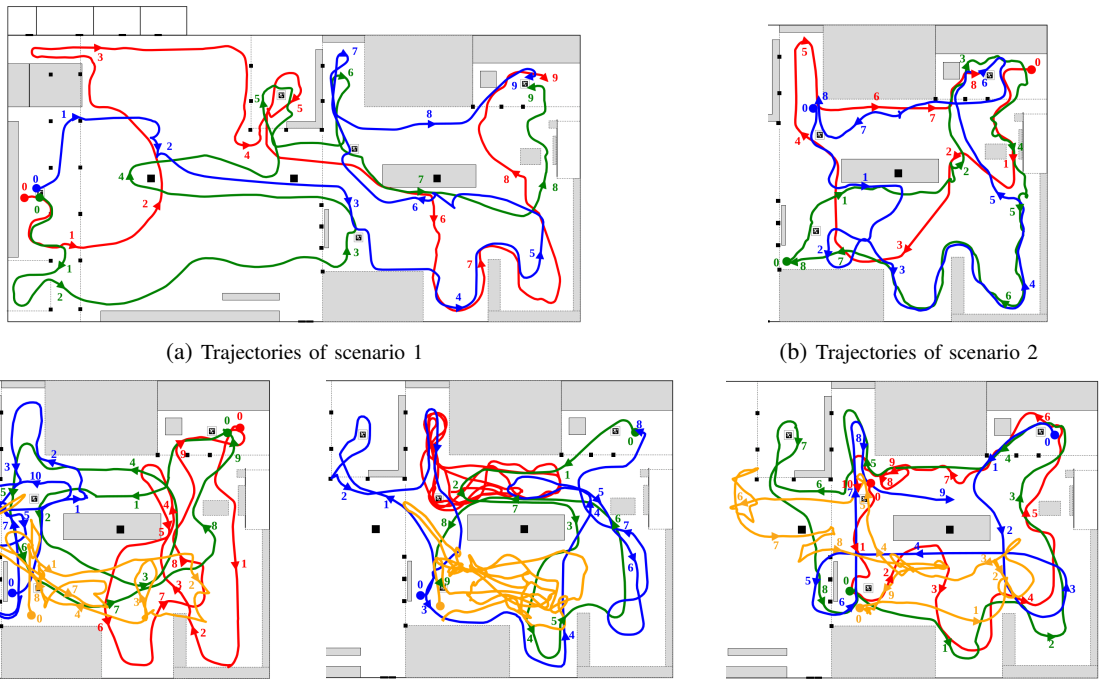
Finally, the third guideline was to provide a realistic framework to stage various challenges arising from network and communication issues. We thus made robots cover wide areas while getting distant from each other. This legitimizes the subsequent simulation of communication losses and recoveries. Furthermore, some robots may never meet or communicate directly, so their data may need to be relayed by third-party robots.

2) Description of the scenarios: Following those principles, we designed five multi-robot scenarios which are detailed below. Figure 5 shows the corresponding trajectories. In scenarios 1, 2 and 5, the trajectories followed by the robots show similar properties such that equivalent drift accumulation can be expected along them. On the contrary, in scenarios 3 and 4, robots follow trajectories with very disparate characteristics: they unequally close loops and individually achieve contrasting estimation accuracies. In scenarios 3 to 5, the drone takes advantage of its higher velocity dynamics to regularly meet distant ground robots and observe their mounted AprilTag markers. The length and duration of each trajectory is provided in Table I.

Scenario		**#1**	**#2**	**#3**	**#4**	**#5**
Duration	[s]	445	387	304	316	304
Robot A traj. length	[m]	229	116	147	182	141
Robot B traj. length	[m]	225	141	143	120	178
Robot C traj. length	[m]	191	162	202	222	131
Drone traj. length	[m]	—	—	203	215	204

TABLE I: Duration and trajectory length for each sequence and robot

Scenario 1. – This scenario, represented by Figure 5a, makes the three ground robots explore the full experimental area. They share the same starting point in the North zone and the same arrival point in the South area. They follow very long and ample trajectories without closing any loop along them, what makes them prone to drift accumulation. However, there exist multiple inter-robot correspondences regularly dispatched between their trajectories, and which allow to close large inter-robot loops. The beginning of those sequences may be challenging for odometry algorithms because the North zone provides few features, most of them belonging to distant structures and exhibiting low parallax. Furthermore, the reflexivity of the ground results into difficult lighting conditions for robot C at the very beginning of the sequence. For each robot, the goal of this scenario is to rely on the large inter-robot loop closures induced by the inter-robot correspondences to mitigate their drift.



(a) Trajectories of scenario 1

(b) Trajectories of scenario 2

(c) Trajectories of scenario 3

(d) Trajectories of scenario 4

(e) Trajectories of scenario 5

Fig. 5: Ground-truth trajectories of the robots in all the scenarios for robots A, B and C, and the drone, which are respectively colored in red, green, blue and orange. The numbered markers along the trajectories indicate the simultaneous localizations of the robots.

Scenario 2. – This scenario, pictured by Figure 5b, is similar to the first one, with the difference that it only covers the South part and makes robots start from distinct locations and come back to their starting point. That aside, each robot covers a significant portion of the South area without closing any loop. Though they mostly navigate close to well-textured surfaces, they occasionally cross empty areas with fewer features and prone to drift accumulation. This is especially the case of robot A near the middle of its trajectory. Multiple inter-robot direct and indirect correspondences are dispatched along the trajectories. For each robot, the objective is similar to the first scenario.

Scenario 3. – This scenario involves the three ground robots and the drone and their trajectories are displayed in Figure 5c. The first two scenarios were designed in such a way that there were no glaring disparities between the properties of the trajectories and that the inter-robot correspondences are evenly distributed among the robots. On the contrary, the third scenario builds on asymmetries. First, robots A and C explore distinct non-overlapping areas and never meet, nor do they share inter-robot correspondences. Robot C also closes more loops than robot A, from which ones it may benefit to estimate its trajectory more accurately. As for robot B, it covers the whole area and meets both other robots. Finally, the drone successively meets all the ground robots. While robot C is not expected to gain much estimation accuracy from its interactions with the other robots, it may bring valuable information to robot B, which may then act as a torchbearer to robot A.

Scenario 4. – Similarly to scenario 3, this scenario, described by Figure 5d, explores how to handle and take advantage on information disparity between the robots. Robot A continuously explores the very same restricted zone. Hence, it closes several loops and may achieve high estimation accuracy. Reversely, robots B and C browse the full South area and periodically meet each other. The drone alternately meets robots B and C, which may result in additional indirect constraints between their trajectories. Furthermore, and overall, robots B and C regularly cross the zone of robot A and accumulate several inter-robot correspondences with it, which then induce informative inter-robot loop closures and constraint large portions of their trajectories. The goal of this scenario for robots B and C is to take advantage on the reliable trajectory estimates of robot A to enhance the accuracy of their own trajectory estimates.

Scenario 5. – In this last scenario, depicted in Figure 5e, all the robots explore the whole South area of the hangar. However, as opposed to all previous scenarios, they never meet each other, nor do they close loops along their own trajectory. Hence, their only means to fuse information is to rely on spotted inter-robot correspondences. Many of them are dispatched since all robots mostly follow each other to visit the same areas in a delayed fashion. This scenario thus aims at assessing how successfully the robots can merge information without any direct inter-robot correspondences.

III. Robotic Platforms and Multi-Robot Setup

The used robotic platforms are displayed in Figure 4. Each ground robot is a Wifibot™, and the drone is a DJI-M100™. All of them are equipped with a stereo camera rig and an IMU. The stereo-rig is composed of two identical IDS UL124xLE monochrome cameras with $4\,\mathrm{mm}$ focal lenses and a baseline of $26\,\mathrm{cm}$ with enabled auto-shutter. The used

resolution is 640×512 pixels with a frame rate of 20 images per seconds. On the ground robots, the IMU is a low-cost MPU-9250™, while we use the IMU integrated in the drone. Robots were connected together using the 5 GHz band of a WiFi router (TP-Link Archer-C7). Clock synchronization was performed by using a common Network Time Protocol (NTP) reference. In each scenario, all individual sequences were acquired simultaneously by controlling the robots manually, while data was recorded on each platform and triggered by a central ground station over the network, using the *ROS* [33] middleware to directly collect data as `.bag` files.

We outfitted robots B and C with mounted AprilTag markers [1] as shown by Figure 4a. Both robots have one AprilTag marker facing backwards to be observed by the other ground robots, and one marker facing upwards to be observed by the drone. Each AprilTag marker is a QR-code tag which can be detected in undistorted images. Knowing the tag size, one can estimate the relative pose $\hat{\boldsymbol{T}}_{\mathrm{CT}}$, between the observer camera frame C and the tag frame T. This allows to get direct inter-robot measurements at meeting points: it could be used as additional constraints between the trajectories of the robots directly or as ground-truth for more advanced methods. An extensive study of the uncertainty associated to AprilTag observations was carried out in [34].

IV. Ground-truth trajectories

Computing the ground-truth trajectories is a critical and delicate point to be able to benchmark odometry and SLAM algorithms on the provided sequences. One general solution is to use specific motion capture sensors during the acquisition of the sequences. This approach was adopted in the *EuRoC* dataset [22]: a Leica Nova MS50 multi-station[3] and a Vicon motion capture system[4] were used to respectively capture the positions and the poses of the drones. However, proceeding this way is limited when it comes to acquiring the ground-truth positions and poses of multiple robots running through a large area. Instead, we used the *Structure-From-Motion* software *Colmap* [35] to build an up-to-scale reconstruction[5] of the environment and extract a reliable trajectory from it. For each scenario, *Colmap* performed an exhaustive matching, which involves extracting and matching SIFT features [36] between all images to find loop closures, and then perform a bundle adjustment optimization to estimate the 3D structure of the environment and the pose of each camera frame. In Table II, we provide the *Colmap* reconstruction statistics for each scenario and which allow to assess their reliability. Such statistics include the number of used images, the number of estimated 3D points, the average track length for each point (i.e. number of observer keyframes) and the average re-projection error. The mean re-projection error is around 0.5 pixels for each reconstruction, what suggests the reconstruction is reliable.

Scenario	#1	#2	#3	#4	#5
Nb. of used images	2478	2330	2442	2534	2448
Nb. of 3D points	295284	474593	323766	314173	309865
Mean tracking length	8.830	9.706	10.555	12.217	11.528
Mean reproj. err. (px)	0.573	0.520	0.549	0.604	0.577

TABLE II: Statistics on *Colmap* reconstructions

Finally, the reconstruction was scaled using the detections of the AprilTag markers of the AprilCubes. For each observation z of a tag T_z from a camera C_z, a relative position ${}^{\mathrm{C}_z}\hat{\boldsymbol{t}}_{\mathrm{C}_z\mathrm{T}_z}$ and relative orientation $\hat{\boldsymbol{R}}_{\mathrm{C}_z\mathrm{T}_z} \in \mathbb{SO}_3$ can be derived. Hence, given a set $\mathcal{Z}$ of AprilTag observations, the scale estimation can be formulated as an optimization problem whose variables Θ include the scale factor s, the orientation matrix $\boldsymbol{R}_{\mathrm{WT}}$ and the position ${}^{W}\boldsymbol{p}_T$ of each observed tag where W denotes the world frame:

$$\Theta^* = \underset{\Theta}{\arg\min} \left\{ \sum_{z \in \mathcal{Z}} \rho\left(\|\boldsymbol{\xi}_R^z\|^2 + \|\boldsymbol{\xi}_t^z\|^2 \right) \right\} \tag{1}$$

$$\text{with} \begin{cases} \boldsymbol{\xi}_R^z \triangleq \log_{\mathbb{SO}_3}\left(\boldsymbol{R}_{\mathrm{WC}_z}^\top \cdot \boldsymbol{R}_{\mathrm{WT}_z} \cdot \hat{\boldsymbol{R}}_{\mathrm{C}_z\mathrm{T}_z}^\top \right)^\vee \\ \boldsymbol{\xi}_t^z \triangleq s \cdot \boldsymbol{R}_{\mathrm{WC}_z}^\top \left({}^{W}\boldsymbol{t}_{\mathrm{WT}_z} - {}^{\mathrm{W}}\boldsymbol{t}_{\mathrm{WC}_z} \right) - {}^{\mathrm{C}_z}\hat{\boldsymbol{t}}_{\mathrm{C}_z\mathrm{T}_z} \end{cases}$$

where ρ denotes the Huber loss function, $\log_{\mathbb{SO}_3}$ denotes the logarithm map from $\mathbb{SO}_3$ to its Lie algebra $\mathfrak{so}_3$ and the *vee* operator maps from $\mathfrak{so}_3$ to $\mathbb{R}^3$. Relative pose constraints between the tags belonging to the same cubes were enforced with additional heavily weighted relative pose factors.

V. Single-robot performances

In the perspective of multi-robot evaluation, we studied the properties and the difficulties inherent to the trajectories by assessing how each robot performs individually in estimating its own trajectory with a monocular, a visual-inertial and a stereo-visual setup. For that purpose, we used three state-of-the-art SLAM algorithms. For the monocular case, we used *ORB-SLAM* [3] which is a keyframe-based SLAM algorithm which employs ORB [37] features for tracking and loop closure detection. For the visual-inertial case, we used *Vins-Mono* [4] which combines a tightly-coupled VI estimator, a loop detection module based on BRIEF [38] features and 4DoF pose-graph optimization for global consistency. Finally, for the stereo-visual case, we used the stereo-only extension of *Vins-Mono* introduced in the framework *Vins-Fusion*. Terrestrial sequences can be more challenging for tightly-coupled visual-inertial odometry since the IMU is not excited in all directions, while robots may furthermore be subjected to significant vibrations and movements through leaps. The odometric drift may also result from locally difficult illumination conditions or from the crossing of less textured areas. To cope with this last point, we increased the number of tracked features up to 300 with a minimum distance of 20 pixels in *Vins-Mono* and *Vins-Fusion*.

Performances were evaluated using the *RPG Trajectory Evaluation toolbox* [39] which first performs a $\mathbb{Sim}_3$ alignment of the estimated trajectory to the groundtruth which

[3]https://leica-geosystems.com/products/total-stations/multistation/leica-nova-ms60

[4]https://www.vicon.com/hardware/cameras/

[5]Unfortunately, *Colmap* does not currently explicitly allow to specify hard baseline constraints between the stereo cameras to perform a stereo reconstruction with an unambiguous scale factor.

Robot	Translation Error [m] A	B	C	D	Scale Factor Error A	B	C	D
Scenario 1								
ORB-SLAM	1.043	1.512	2.135	—	—	—	—	—
Vins-Mono	2.617	2.382	2.314	—	0.091	0.050	0.086	—
Vins-Fusion	0.970	1.244	0.704	—	0.095	0.071	0.067	—
Scenario 2								
ORB-SLAM	0.760	1.851	0.516	—	—	—	—	—
Vins-Mono	1.510	1.998	2.305	—	0.215	0.065	0.059	—
Vins-Fusion	0.354	0.867	0.299	—	0.016	0.033	0.014	—
Scenario 3								
ORB-SLAM	0.682	0.096	0.417	0.163	—	—	—	—
Vins-Mono	3.743	1.306	0.969	0.439	0.059	0.055	0.059	0.015
Vins-Fusion	0.945	0.508	0.144	0.259	0.001	0.011	0.004	0.001
Scenario 4								
ORB-SLAM	0.320	0.967	1.323	0.051	—	—	—	—
Vins-Mono	0.761	1.476	2.482	0.147	0.126	0.209	0.024	0.017
Vins-Fusion	0.164	0.260	0.472	0.122	0.001	0.004	0.014	0.004
Scenario 5								
ORB-SLAM	0.086	0.093	2.405	0.062	—	—	—	—
Vins-Mono	3.338	1.440	1.422	0.254	0.065	0.083	0.132	0.008
Vins-Fusion	0.175	0.360	0.124	0.559	0.002	0.019	0.005	0.026

(a) Absolute Translation Errors

Robot	Relative Error 5 m A	B	C	D	Relative Error 10 m A	B	C	D
Scenario 1								
ORB-SLAM	1.630	2.872	2.537	—	1.458	2.245	2.668	—
Vins-Mono	0.851	0.881	1.278	—	1.680	1.422	2.119	—
Vins-Fusion	0.151	0.162	0.147	—	0.245	0.342	0.718	—
Scenario 2								
ORB-SLAM	0.839	1.748	0.634	—	1.239	1.381	0.446	—
Vins-Mono	0.907	0.934	0.731	—	1.862	1.614	1.712	—
Vins-Fusion	0.146	0.206	0.165	—	0.246	0.419	0.266	—
Scenario 3								
ORB-SLAM	1.266	0.303	0.700	3.183	1.108	0.313	0.709	3.401
Vins-Mono	0.686	0.609	0.791	0.149	1.389	1.056	1.108	0.227
Vins-Fusion	0.292	0.181	0.153	0.127	0.512	0.323	0.254	0.183
Scenario 4								
ORB-SLAM	0.510	1.034	1.778	2.539	0.630	0.831	1.575	3.360
Vins-Mono	0.831	0.778	1.049	0.126	1.363	1.059	1.584	0.173
Vins-Fusion	0.185	0.107	0.169	0.125	0.268	0.195	0.295	0.173
Scenario 5								
ORB-SLAM	0.708	0.593	1.903	4.210	0.335	0.515	1.966	2.966
Vins-Mono	1.065	0.765	0.474	0.134	1.703	1.241	0.794	0.207
Vins-Fusion	0.133	0.145	0.085	0.158	0.231	0.270	0.139	0.242

(b) Relative Translation Errors

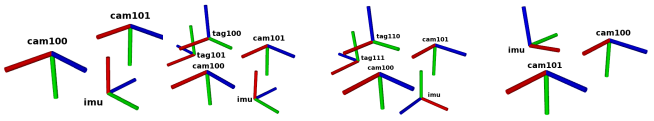
TABLE III: Evaluation of single-robot SLAM algorithms

minimizes the translation errors between the temporally matched frames. Translation errors and the scale factor error (evaluated as the scale correction induced by the above mentionned $\mathbb{S}\text{im}_3$ alignment) are reported in Table IIIa.

VI. Dataset Specifications and Data Format

A. Camera-IMU and AprilTag markers calibration

On each robot, the cameras bench was spatially and temporally calibrated using the *Kalibr* toolbox [40]. This step allows to estimate the camera intrinsic coefficients $\boldsymbol{k}$, its distortion coefficients $\boldsymbol{d}$ and the camera-to-IMU extrinsic transformation $\boldsymbol{T}_{\text{CB}} \in \mathbb{SE}_3$ – where C denotes the camera frame and B is the body frame. We need the distortion coefficients to map any 3D point to its 2D distorted projection onto the image plane using the classical pinhole camera model. The camera-to-IMU extrinsic transformation relates the inertial measurements to the observed visual movement. The camera intrinsic coefficients $\boldsymbol{k} = [f_x, f_y, c_x, c_y]^\top$ include the focal lengths f_x, f_y and the coordinates c_x, c_y of the principal point. The provided distortion coefficients are those of the equidistant distortion model [41] i.e. $\boldsymbol{d} = [k_1, k_2, k_3, k_4]^\top$. Finally, the temporal calibration estimates the time delay $t_{\text{cam}\to\text{imu}}$ between the IMU and camera clocks $t_{\text{imu}} = t_{\text{cam}} + t_{\text{cam}\to\text{imu}}$. Figure 6 provides a view of the estimated frames for the camera, IMU and AprilTag frames for each robot.



(a) Robot A (b) Robot B (c) Robot C (d) Drone

Fig. 6: Reference frames attached to the robots

For each mounted AprilTag marker T, we estimated the pose T_{CT} w.r.t. the frame C of one camera of the robot by using one large AprilGrid calibration target which was simultaneously observed by the robot and one moving calibrated camera which also observed the mounted AprilTag markers. We provide the pose and id of the used AprilTags. All the used tags belong to the `36h11` family (see [1])

B. Sensor acquisitions

The sensor acquisitions are provided as *ROS* `.bag` files. For each robot in each sequence, we splitted the acquisitions into two `.bag` files. The first one holds visual data from camera `cam100` and the IMU measurements; it may be used for monocular and visual-inertial algorithms. The second `.bag` file holds the complementary visual data from the second camera `cam101`; it may be played along with the first one for stereo(-inertial) settings.

C. Ground-truth trajectories

The ground-truth trajectories are provided as text files, organized in the following way:

# timestamp	tx	ty	tz	qx	qy	qz	qw
1566395013.9421284199	-1.23891	-19.98109	0.26091	-0.27863	-0.65316	-0.27460	0.64834
1566395014.4422287941	-1.23891	-19.98107	0.26092	-0.27864	-0.65314	-0.27461	0.64835
1566395014.9421727657	-1.23892	-19.98108	0.26092	-0.27862	-0.65315	-0.27460	0.64836

The poses are reported in the form $\boldsymbol{T}_{\text{WB}} = [{}^{\text{W}}\boldsymbol{t}_{\text{WB}}, \boldsymbol{q}_{\text{WB}}]^\top$ where $\boldsymbol{q}_{\text{WB}}$ denotes the orientation quaternion, W is the world frame and B is the body frame.

VII. Conclusion

In this paper, we presented a new dataset intended for testing multi-robot stereo-visual and inertial SLAM algorithms with ground and aerial robots. This dataset is made publicly available and is intended to fill a void of benchmark data in multi-robot visual SLAM evaluation. Five scenarios were acquired with sequences of various difficulty degrees, and which aim at rendering some of the opportunities and challenges brought by multi-robot SLAM. As perspectives, we plan to augment this dataset with the measurements acquired by the Intel® Realsense™ D435i sensor which also outfitted all the robots, and which consists in a stereo RGB-Depth camera bench associated with another embedded Inertial Measurement Unit.

VIII. Acknowledgements

This work was supported by the *Direction Générale de l'Armement* (DGA). We thank Martial Sanfourche, Julien Moras, Julien Marzat and Guillaume Hardouin for their involvement in the acquisition of the sequences.

REFERENCES

[1] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 3400–3407.

[2] J.-E. Deschaud, "Imls-slam: scan-to-model matching based on 3d data," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2480–2485.

[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] 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] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "Openvins: A research platform for visual-inertial estimation," in *IROS 2019 Workshop on Visual-Inertial Navigation: Challenges and Applications, Macau, China*, 2019.

[6] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[7] L. Riazuelo, J. Civera, and J. M. Montiel, "C2tam: A cloud framework for cooperative tracking and mapping," *Robotics and Autonomous Systems*, vol. 62, no. 4, pp. 401–413, 2014.

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

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

[10] M. Karrer, P. Schmuck, and M. Chli, "Cvi-slamcollaborative visual-inertial slam," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2762–2769, 2018.

[11] A. Cunningham, M. Paluri, and F. Dellaert, "Ddf-sam: Fully distributed slam using constrained factor graphs," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 3025–3030.

[12] M. J. Schuster, C. Brand, H. Hirschmüller, M. Suppa, and M. Beetz, "Multi-robot 6d graph slam connecting decoupled local reference filters," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 5093–5100.

[13] K. Sartipi, R. C. DuToit, C. B. Cobar, and S. I. Roumeliotis, "Decentralized visual-inertial localization and mapping on mobile devices for augmented reality," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 2145–2152.

[14] R. Dubois, A. Eudes, and V. Frmont, "On data sharing strategy for decentralized collaborative visual-inertial simultaneous localization and mapping," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 2123–2130.

[15] T. Cieslewski, S. Choudhary, and D. Scaramuzza, "Data-efficient decentralized visual slam," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2466–2473.

[16] M. Giamou, K. Khosoussi, and J. P. How, "Talk resource-efficiently to me: Optimal communication planning for distributed loop closure detection," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–9.

[17] S. Choudhary, L. Carlone, C. Nieto, J. Rogers, H. I. Christensen, and F. Dellaert, "Distributed trajectory estimation with privacy and communication constraints: a two-stage distributed gauss-seidel approach," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 5261–5268.

[18] T. Cieslewski, S. Lynen, M. Dymczyk, S. Magnenat, and R. Siegwart, "Map api-scalable decentralized map building for robots," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 6241–6247.

[19] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 3. IEEE, 2004, pp. 2149–2154.

[20] A. Antonini, W. Guerra, V. Murali, T. Sayre-McCord, and S. Karaman, "The blackbird dataset: A large-scale dataset for uav perception in aggressive flight," *arXiv preprint arXiv:1810.01987*, 2018.

[21] W. Wang, D. Zhu, X. Wang, Y. Hu, Y. Qiu, C. Wang, Y. Hu, A. Kapoor, and S. Scherer, "Tartanair: A dataset to push the limits of visual slam," *arXiv preprint arXiv:2003.14338*, 2020.

[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] M. Ferrera, V. Creuze, J. Moras, and P. Trouvé-Peloux, "Aqualoc: An underwater dataset for visual–inertial–pressure localization," *The International Journal of Robotics Research*, vol. 38, no. 14, pp. 1549–1559, 2019.

[24] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[25] J.-L. Blanco-Claraco, F.-Á. 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.

[26] K. Leung, D. Lühr, H. Houshiar, F. Inostroza, D. Borrmann, M. Adams, A. Nüchter, and J. Ruiz del Solar, "Chilean underground mine dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 16–23, 2017.

[27] M. Vayugundla, F. Steidle, M. Smisek, M. J. Schuster, K. Bussmann, and A. Wedler, "Datasets of long range navigation experiments in a moon analogue environment on mount etna," in *ISR 2018; 50th International Symposium on Robotics*. VDE, 2018, pp. 1–7.

[28] R. A. Hewitt, E. Boukas, M. Azkarate, M. Pagnamenta, J. A. Marshall, A. Gasteratos, and G. Visentin, "The katwijk beach planetary rover dataset," *The International Journal of Robotics Research*, vol. 37, no. 1, pp. 3–12, 2018.

[29] O. Lamarre, O. Limoyo, F. Marić, and J. Kelly, "The canadian planetary emulation terrain energy-aware rover navigation dataset," *The International Journal of Robotics Research*, p. 0278364920908922, 2020.

[30] J. Delmerico, T. Cieslewski, H. Rebecq, M. Faessler, and D. Scaramuzza, "Are we ready for autonomous drone racing? the uzh-fpv drone racing dataset," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6713–6719.

[31] A. L. Majdik, C. Till, and D. Scaramuzza, "The zurich urban micro aerial vehicle dataset," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 269–273, 2017.

[32] K. Y. Leung, Y. Halpern, T. D. Barfoot, and H. H. Liu, "The utias multi-robot cooperative localization and mapping dataset," *The International Journal of Robotics Research*, vol. 30, no. 8, pp. 969–974, 2011.

[33] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, no. 3.2. Kobe, Japan, 2009, p. 5.

[34] M. J. Schuster, "Collaborative localization and mapping for autonomous planetary exploration."

[35] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4104–4113.

[36] G. Lowe, "Sift-the scale invariant feature transform," *Int. J*, vol. 2, pp. 91–110, 2004.

[37] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[38] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *European conference on computer vision*. Springer, 2010, pp. 778–792.

[39] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7244–7251.

[40] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1280–1286.

[41] J. Kannala and S. Brandt, "A generic camera calibration method for fish-eye lenses," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 1. IEEE, 2004, pp. 10–13.