3D Registration of Aerial and Ground Robots for Disaster Response: An Evaluation of Features, Descriptors, and Transformation Estimation

Abel Gawel*, Renaud Dubé*, Hartmut Surmann†,
Juan Nieto*, Roland Siegwart* and Cesar Cadena*
*Autonomous Systems Lab, ETH Zurich, †University of Applied Sciences Gelsenkirchen

***Abstract*— Global registration of heterogeneous ground and aerial mapping data is a challenging task. This is especially difficult in disaster response scenarios when we have no prior information on the environment and cannot assume the regular order of man-made environments or meaningful semantic cues. In this work we extensively evaluate different approaches to globally register UGV generated *3D* point-cloud data from LiDAR sensors with UAV generated point-cloud maps from vision sensors. The approaches are realizations of different selections for: a) local features: key-points or segments; b) descriptors: FPFH, SHOT, or ESF; and c) transformation estimations: RANSAC or FGR. Additionally, we compare the results against standard approaches like applying ICP after a good prior transformation has been given. The evaluation criteria include the distance which a UGV needs to travel to successfully localize, the registration error, and the computational cost. In this context, we report our findings on effectively performing the task on two new Search and Rescue datasets. Our results have the potential to help the community take informed decisions when registering point-cloud maps from ground robots to those from aerial robots.**

I. Introduction

Multi-robot applications with heterogeneous robotic teams are an increasing trend due to numerous advantages. An Unmanned Ground Vehicle (UGV) can often carry high payloads and operate for extended periods of time, while an Unmanned Aerial Vehicle (UAV) offers swift deployment and the opportunity to rapidly survey large areas. This is especially beneficial in Search and Rescue (SaR) scenarios, see Fig. 1 as an example. Here, an initial overview can be made using UAVs before deploying UGVs for closer exploration in areas of interest. However, when it comes to efficiently combining the strengths of such robotic teams, we face numerous challenges. Additionally to the large difference in point of view, the sensor modalities used for mapping and localization are often drastically different for UAVs and UGVs. While UAVs typically use cameras as the prime sensor, UGVs often rely on LiDAR. This poses a major challenge in efficiently exploiting the UAV data on a UGV as registration between different sensor modalities is difficult to perform. Furthermore, using advanced functionalities such as traversability analysis and path planning for UGVs on UAV generated maps requires tight alignment between the data of different modalities, active localization and suitable map representations.

One critical step in using maps across several robots is the identification of the alignment between their maps. Several techniques are possible with increasing generality [1]. Firstly it is possible to impose a common origin of different robots' maps, e.g., by using common starting locations as done by Michael et al. [2]. Another option is to use global positioning sensors that allow for a good initial guess on the alignment of coordinate frames. In the case that several robots operate concurrently, it is also possible to find an alignment by a relative localization of the robots against each other [3]. However, the most challenging task is to register maps without any prior information regarding their mutual alignment. Furthermore, for the SaR application as in the "Long-Term Human-Robot Teaming for Robots Assisted Disaster Response" (TRADR) project, the scenarios are completely unpredictable which rules out the possibility of using supervised learning into the pipeline [4].

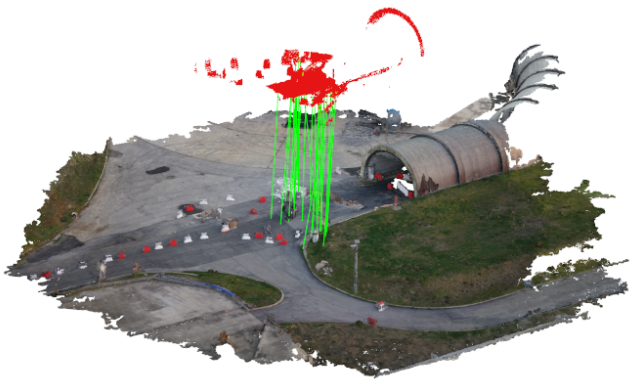


Fig. 1: Global localization of a *3D* UGV sub-map (red point-cloud) in a *3D* UAV reference map (coloured point-cloud). Green lines indicate the resulting matches associated to points in the two point-clouds. The data stems from the Montelibretti outdoor dataset and drawn from the complex experimental set-up.

While our previous work on online multi-robot SLAM for *3D* LiDARs [5][1] demonstrates a reliable registration among point-cloud maps taken from multiple ground robots, there is still the issue of dealing with differences in modality and in point of view between UAVs and UGVs.

The above mentioned challenges motivate us to evaluate several techniques to globally localize a UGV using its LiDAR sensor in a point-cloud map generated using the Multi-View Reconstruction Environment (MVE) [6] from images recorded by a UAV. The global registration (or localization) pipeline schematized in Fig. 2 consists of feature extraction, feature description and matching, and a 3D transformation estimation. We provide an evaluation and an analysis of the

[1]Within this paper, this system will be referred to as LaserSLAM.

implementation and performance of different choices for the modules in this registration pipeline. These choices are:

- Local feature extraction: key-points or segments.
- Feature descriptors: Fast Point Feature Histogram (FPFH), Unique Signatures of Histograms for Local Surface Description (SHOT) or Ensemble of Shape Functions (ESF).
- Transformation estimation: RANSAC based or Fast Global Registration (FGR).

The evaluation is conducted on two real world datasets of an indoor and an outdoor SaR scenario. This paper presents the following contributions:

- Extensive evaluation of global registration realizations for registering UGV and UAV point-clouds from LiDAR and camera data respectively.
- Two new datasets for multi-modal SLAM in SaR scenarios.

II. Related work

The field of *2D* metrical map-merging based on overlapping map segments is well studied in literature [7–9]. However, the task is increasingly difficult when moving to *3D* environments [1], especially when dealing with heterogeneous robotic teams, where *3D* data is generated from different sensors and with different noise characteristics [10]. Michael et al. [2] demonstrate a system for collaborative UAV-UGV mapping. The authors propose a system where a UGV equipped with a LiDAR sensor performs *2.5D* mapping, using the flat ground assumption and consecutively merging scans using Iterative Closest Point (ICP). In dedicated locations a UAV equipped with a 2D LiDAR is launched from the UGV and maps the environment using a pose-graph SLAM algorithm. Maps generated from the UAV are then fused online with the UGV map using ICP initialized at the UAV starting location.

Forster et al. [11] go a step further in fusing UAV-UGV map data from different sensors, i.e., RGB-D maps from the UGV and dense monocular reconstruction from the UAV. The registration between the maps is performed using a *2D* local height map fitting in x and y coordinates with an initial guess within a $3m$ search radius. The orientation is a priori recovered from the magnetic north direction as measured by the Inertial Measurement Unit (IMU)s. In a related setting Hinzmann et al. [12] evaluate different variants of ICP for registering dense *3D* LiDAR point-clouds and sparse *3D* vision point-clouds from Structure from Motion (SfM) recorded with different UAVs into a common point-cloud map using an initial GPS prior for the map alignment.

Instead of using the generated *3D* data for localizing between RGB and *3D* LiDAR point-cloud data, Wolcott and Eustice [13] propose to generate *2D* views from the LiDAR point-clouds based on the surface reflectivity. However, this work focuses only on localization and it is demonstrated only on maps recorded from similar points of view.

In our previous work [14] we presented a global registration scheme between sparse *3D* LiDAR maps from UGVs and vision keypoint maps from UAVs, exploiting the rough geometric structure of the environment. Here, registration is performed by clustering of geometric keypoint descriptors matches between map segments under the assumption of a known z-direction as determined by an IMU.

Zeng et al. [4] present geometric descriptor matching based on learning. However, this approach is infeasible in unknown SaR scenarios, as the descriptors do not generalize well to unknown environments.

Dubé et al. [15] demonstrate better global localization performance in *3D* LiDAR point-clouds by using segments as features instead of key-points. This approach has been demonstrated with multiple UGVs with the same robot-sensor set-up, but it is still to be studied how the approach performs under large changes in point of view.

Assuming good initialization of the global registration, Zhou et al. [16] perform a robust optimization. The work claims faster and more robust performance than ICP.

In summary, the community addresses the problem of heterogeneous localization. However, there is a research gap in globally localizing from one sensor modality to the other in full *3D* without strong assumptions on view-point, terrain or initial guess.

III. Aerial-Ground robot mapping system

In this section, we present our SLAM system. It extends our LaserSLAM system [5] with the component of global map alignment and localization in point-cloud maps from different sources, as well as online extension of these maps. Fig. 2 illustrates the architecture of the proposed system. While the major LaserSLAM system is running on the UGV, it also allows to load, globally align, and use point-cloud maps from other sources. As example, maps generated via MVE from UAVs or point-cloud maps resulting from bundle adjustment on data collected by another LiDAR equipped robot can all be leveraged.

A. Mapping algorithms

In the proposed system, we use a dual map representation for the different tasks of the robots, i.e., point-cloud maps and *OctoMaps* [17]. While the individual robots maintain point-clouds and surface meshes, these are integrated in a global *OctoMap* representation serving as the interface to other modules of the SaR system, e.g., traversability analysis as shown in [18]. Another advantage of the unified *OctoMap* representation is a persistent representation which also incorporates dynamic changes detection.

On the UAV's monocular image data we perform an SfM and Multi-View-Stereo-based scene reconstruction using the MVE [6]. The MVE produces a dense surface mesh of the scene by extensive matching and is therefore an offline method that is computed off-board the UAV. Although, efficient online mapping methods exist, we decide to use a method that produces high quality maps, that can be further used in the TRADR system, e.g., on the UGV for path planning or for situation awareness of first responders.

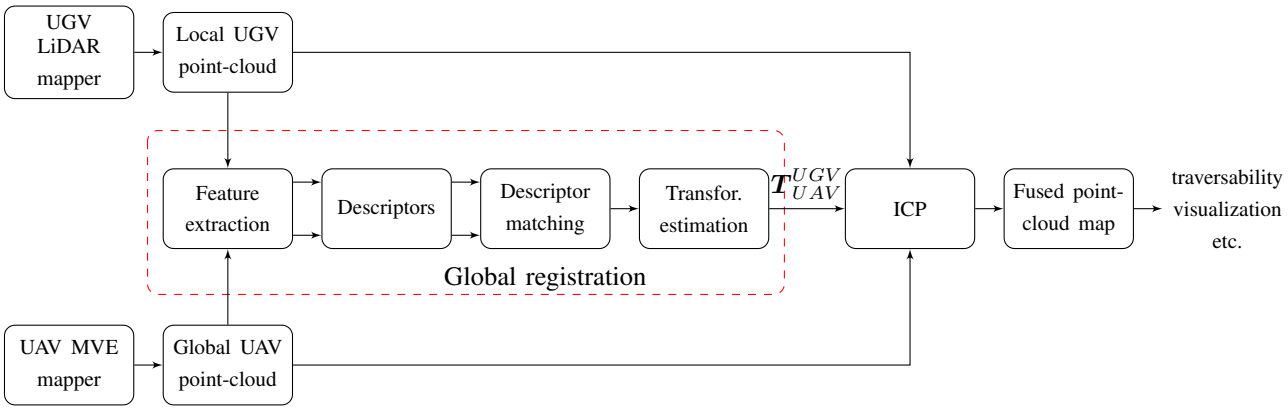


Fig. 2: Mapping System overview. The inputs to the system are the local UGV map and the global UAV reference map. If a global registration is triggered, the key-points are computed on both maps. Consecutively, the system performs descriptor extraction and matching. The initial global transformation is then refined by a step of ICP between the global and the local clouds, resulting in a fused map that is used for further functionalities of the system, such as path planning.

On the UGVs we use a variant of the LaserSLAM system which estimates in real-time the robot trajectory alongside with the *3D* point-cloud map of the environment. LaserSLAM is based on the *iSAM2* [19] pose-graph optimization approach and implements different types of sequential and place recognition constraints. In this work, odometry constraints are obtained by fusing wheel encoders and IMU data using an extended Kalman filter while scan-matching constraints are obtained based on ICP between successive scans.

After the creation of the UAV map, we facilitate a global registration scheme to localize the UGV in the UAV map as described in Sec. IV.

For UGV-only mapping, the LaserSLAM framework enables multiple robots to create consistent *3D* point-cloud maps. However, a different regime must be followed for generating and extending a consistent *3D* map by fusing in the UAV dense *3D* maps. Since the UAV maps are the result of an offline batch optimization process, the maps are already loop closed and represent a consistent initial basis for the global map. Furthermore, we treat the UAV maps as static, i.e., the map is taken *as is*. LaserSLAM is therefore extended to include a mode that allows the robot to use a given base map. This base map is then treated as the fully optimized map and extended with updates from the UGV LiDAR. It is important to note that the point-cloud map is only the internal representation for the robot to perform SLAM. All map updates as well as dynamic changes are maintained in the *OctoMap* representation that is derived from the point-cloud updates and serves as a unified representation for all processes using the mapping data.

B. Map usage

Thanks to the unified *OctoMap* interface, the merged map data can directly be used on other modules of the TRADR system. Notably, it can directly be used for traversability analysis and subsequent metrical path planning. The system therewith enables the UGVs to also use UAV generated maps for path planning. A UGV-loaded UAV map of one testing site (see Fig. 1) is depicted in Fig. 3 indicating traversable areas in green and non-traversable areas in red.

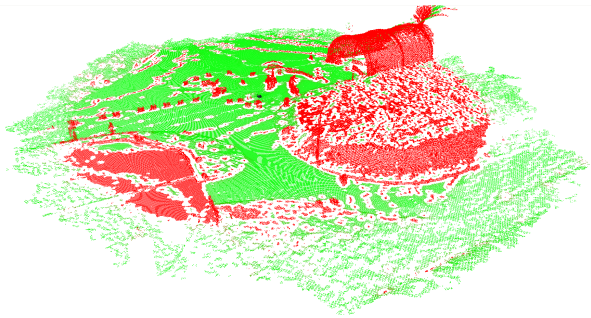


Fig. 3: Resulting UGV traversability estimation on the outdoor Montelibretti dataset. Traversable areas are marked in green, while non traversable areas are indicated with red. The parametrization is identical to the parametrization for LiDAR traversability analysis.

The *OctoMap* serves as interface for further modules of the system, such as novelty detection which is a separate contribution and out of the scope of this work.

IV. Global Registration

This section describes the pipeline that we use to globally register the UGV with respect to the UAV point-cloud map. The evaluation of different choices within this pipeline is the focus of our paper. The global registration consist of four modules: feature extraction, description, matching and, estimation of the $\mathbb{SE}(3)$ transformation.

A. Feature extraction

This module defines which components in the point-cloud map are going to be used for the registration. Key-point are samples from the full point-cloud that have some level of invariance to the point of view. Here, we use the Intrinsic Shape Signatures (ISS) detector [20]. The next option, is to add more information by clustering the point-cloud, resulting in segments. These segments are taken as the local features with the potential of being more descriptive than just 3D points [15]. Here, we follow a Euclidean based clustering

Realization	Feature	Descriptor	Trans. Estimation
FPFH	Key-point	FPFH	RANSAC-based
FPFH FGR	Key-point	FPFH	FGR
FPFH seg	Segments	FPFH	RANSAC-based
SHOT	Key-point	SHOT	RANSAC-based
SHOT FGR	Key-point	SHOT	FGR
SHOT seg	Segments	SHOT	RANSAC-based
ESF seg [15]	Segments	ESF	RANSAC-based

TABLE I: Global registration realization by different choices in the sub-modules.

as the segmentation algorithm. We do not explore global features as they are highly point of view dependent.

B. Descriptors

This module takes each feature and computes a descriptor with the aim of being descriptive enough such that it is reproducible on different maps of the same location. The descriptor is based on the key-point, and its neighborhood, or on the subset of points that belong to a segment. Here, we explore three descriptors:

- Fast Point Feature Histogram (FPFH) [21].
- Unique Signatures of Histograms for Local Surface Description (SHOT) [22].
- Ensemble of Shape Functions (ESF) [23].

C. Description Matching

The matching module is in charge of solving the data association problem between features from both maps by comparing their descriptors. In our implementation we use the nearest neighbor search in the space of the corresponding descriptor.

D. Transformation Estimation

Once a set of 3D point pairs is declared, this module computes the transformation such that the 3D points from one map are moved to the location of their correspondences in the reference map. In absence of outliers, the problem could be solved by minimizing a least square error function. Unfortunately, the presence of outliers is unavoidable and this module must deal with them. Here we explore two alternative methods. The first one is a RANSAC-based approach which is already available in PCL. The second one is based on the recent proposed FGR [16]. FGR uses the scaled Geman-McClure estimator as robust cost function into the optimization objective to neutralize the possible outlier matches.

E. Realizations

We explore different global registration alternatives by choosing different methods in each module. The realizations are as shown in Table I.

As global registration strategies, the evaluation focuses on 11 different configurations. Those that are shown in Table I plus their combinations when removing the ground plane prior to key-point detection, denoted by *gr* at the end. Ground removal is done by RANSAC based plane fitting.

F. Performance metrics

For the evaluation metrics, we use transformation errors $\Delta\boldsymbol{T}$ on the alignment between the UGV and UAV maps that are represented as

$$\Delta\boldsymbol{T} = \begin{bmatrix} \Delta\boldsymbol{R} & \Delta\boldsymbol{t} \\ \boldsymbol{0} & 1 \end{bmatrix} \tag{1}$$

with rotation matrix $\Delta\boldsymbol{R}$ and translation vector $\Delta\boldsymbol{t} = (\Delta\mathbf{x}, \Delta\mathbf{y}, \Delta\mathbf{z})^{\mathbf{T}}$. The translational error e_t is computed as follows:

$$e_t = \|\Delta\boldsymbol{t}\| = \sqrt{\Delta\mathbf{x}^2 + \Delta\mathbf{y}^2 + \Delta\mathbf{z}^2} \tag{2}$$

The rotational error e_r equates to:

$$e_r = \arccos \frac{trace(\Delta\boldsymbol{R} - \boldsymbol{I})}{2} \tag{3}$$

It is important to note that the two map types are not perfectly aligned in all locations due to the multi-modal nature of the data and we can therefore only evaluate errors down to a positional resolution of $0.2m$ and angular resolution of 2° respectively. Furthermore, we register the data using ICP in the basic experiment, to give an indication on the achievable alignment, as ICP always converged to a good solution in our experiments given a good initial guess. Motivated by the results of Hinzmann et al. [12], we consider registrations as successful when ICP is able to perform the final local alignment. Therefore, the thresholds for translational and rotational errors are set to $e_t = 3m$ and $e_r = 5^\circ$ above the resulting ICP solution of the basic experiment to count successful registrations.

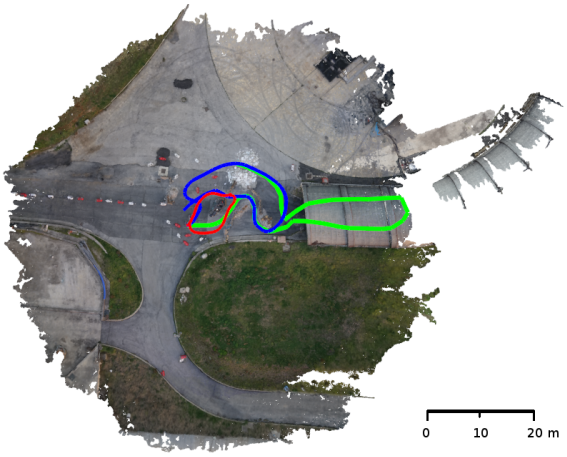
V. Experiments

We evaluate our approach on two challenging SaR datasets recorded within the TRADR project which we will make available with this publication. The evaluation focuses on the global registration of the multi-modal point-cloud data.

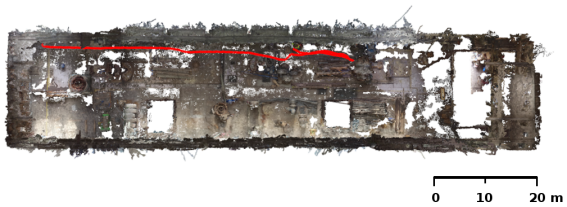
A. Datasets

The first of these datasets was generated in an outdoor firemen training location in Montelibretti, Italy. The scenario simulates a car accident around a tunnel. It consists of six UGV runs in partly overlapping locations of the disaster area and one large UAV-generated map covering the whole site of approximately $80m \times 80m$. The travelled trajectories of the robots are $2 \times 30m$, $2 \times 60m$, and $2 \times 110m$ of consecutive missions following the same paths twice. For the evaluation we use one UGV run of each size. Here, the UGV mapping data is fully covered in the UAV map, except for an indoor exploration of the tunnel which was not accessible to the UAV. The scenario and the trajectories are depicted in Fig. 4a.

The second dataset was recorded at a decommissioned power plant in Dortmund, Germany, consisting of several UGV runs and several large UAV-generated maps covering different parts of the power plant, including the entire machine hall which was also visited by one UGV and has a size of $100m \times 20m$. The UGV run is fully covered in the UAV map and has a travelled distance of approximately $80m$, as illustrated in Fig. 4b.



(a) Montelibretti outdoor dataset.



(b) Dortmund indoor dataset.

Fig. 4: Top-down views of two SaR datasets considered in our experiments. The robot trajectories are indicated in green, red, and blue and overlaid on the colored UAV point-clouds.

B. Experimental setup

We evaluate several global registration strategies on increasingly challenging experimental set-ups. All set-ups consider the iteratively growing UGV map produced by laser-SLAM as local map. For the basic set-up, the global map is a cropped version of the UAV map, that approximately covers the space of the local map at any iteration. The more challenging intermediate set-up uses the cropped UAV map as seen at the last iteration of the UGV mapping as global map. Finally, the complex set-up considers the full UAV map for global localization.

As global registration strategies, the evaluation focuses on the 11 configurations presented in Sec. IV-E.

C. Registration performance

This section evaluates the global registration performance of the different algorithms considered, using the metrics presented in Sec. IV-F.

1) Parametrization: We choose the parametrization of the FPFH and SHOT descriptors to yield good performance across all data used, i.e., a histogram and search radius of $2.0m$ for FPFH and SHOT respectively. Our parameter choice is further motivated by extensive evaluation and shows plateauing performance in a large region around the chosen size, indicating robust performance. The matcher is based on performing fast nearest neighbor search in a FLANN tree [24], while the geometric verification is based on RANSAC and clustering. Furthermore, FGR is parametrized for the best possible performance we could find and using the same parametrization for FPFH and SHOT as the other experiments.

D. Results

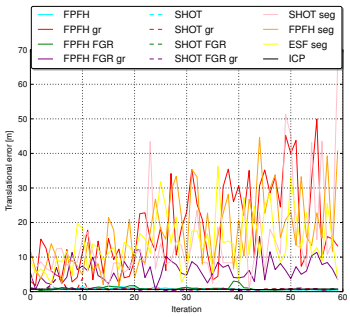
Fig. 1 and Fig. 5 illustrate qualitative global registration between UGV sub-maps and global UAV maps on the tested datasets. In Fig. 6 and Table II the quantitative performance of the evaluated approaches in depicted. While Fig. 6 shows translational and rotational error of the individual approaches over all datasets, Table II reports the minimal amount of cumulated UGV scans, i.e., the minimal travelled distance for reliable global registration. Here, we define reliable global registration, if from the associated UGV sub-map, the errors do not exceed the error thresholds e_t and e_r for 90% of the cases.

In this experimental set-up the descriptor matching approach as described in Section IV performs best throughout all experiments. FPFH yields satisfying performance in the basic experiments. However, its performance drastically degrades in the more complex cases.

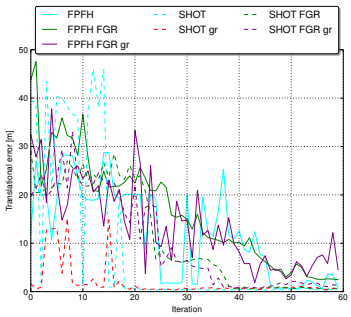
SHOT on the other hand shows reliable performance throughout all experiments, with the required overlap increasing with the complexity of our test-cases. Here, the ground removal does not provide a significant performance boost, especially since the ground plane extraction is unreliable on the large UAV map as we do not have incremental pose-updates from the robot on segments of the map. However, ground removal does not degrade the performance as it does for FPFH, expressing robust performance of SHOT over varying conditions.

While the RANSAC-based geometric verification can reject a large amount of mismatched descriptors and does not rely on the point of initialization, FGR is less robust to poor initialization as done for the intermediate and complex experiments. For the reduced search space in the basic and intermediate experiments, FGR is able to achieve reasonable registration performance and therefore shows high potential to be used for such reduced search problems, when carefully modelling its robust cost function.

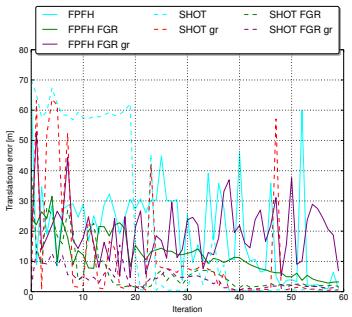
While the segmentation approach shows very good performance for reproducible segmentations, e.g., single-modality localization [15], we found that the considered parametrizations of Euclidean segmentation in combination with the considered descriptors did not generalize well between the modalities and could not deliver interesting results in the experiments. For the sake of clarity of the plots, we therefore only show their performance in the basic experiments in Fig. 6. Since the remainder of the matching algorithm is identical to the well performing descriptor matching, we believe that given a reliable segmentation, the approach has the potential to yield very good performance for the global registration case. However, a purely geometric ground removal and segmentation on the full UAV point-cloud that is comparable to the segmentation on the UGV map is a hard problem, especially for cluttered SaR scenarios.



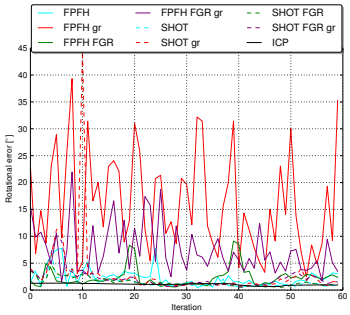
(a) Translational error for basic Montelibretti experiment.



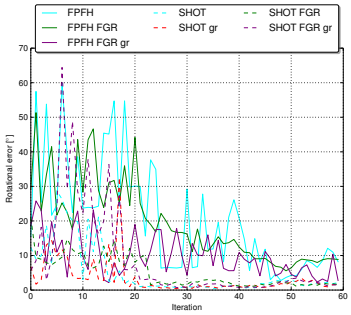
(b) Translational error for intermediate Montelibretti experiment.



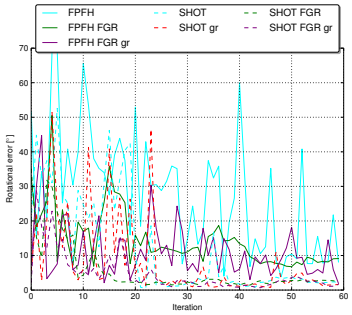
(c) Translational error for complex Montelibretti experiment.



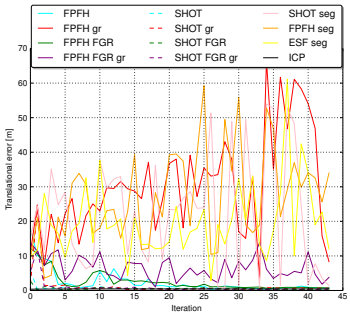
(d) Rotational error for basic Montelibretti experiment.



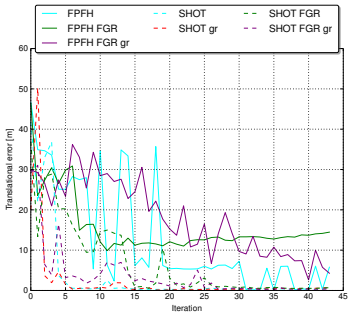
(e) Rotational error for intermediate Montelibretti experiment.



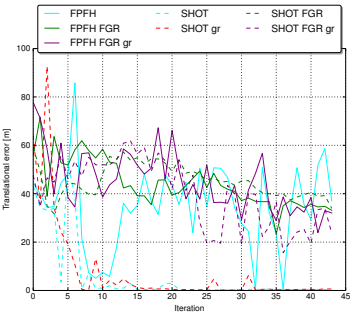
(f) Rotational error for complex Montelibretti experiment.



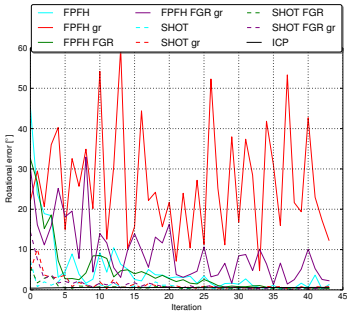
(g) Translational error for basic Dortmund experiment.



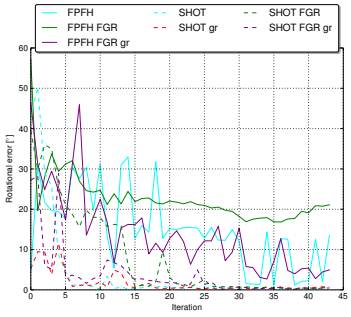
(h) Translational error for intermediate Dortmund experiment.



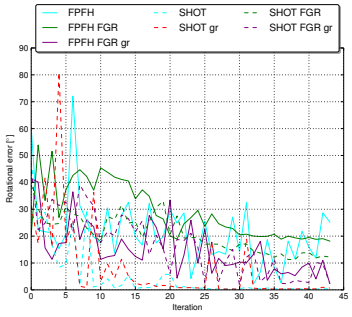
(i) Translational error for complex Dortmund experiment.



(j) Rotational error for basic Dortmund experiment.



(k) Rotational error for intermediate Dortmund experiment.



(l) Rotational error for complex Dortmund experiment.

Fig. 6: Translational and rotational error of plots for the Montelibretti (outdoor) and Dortmund (indoor) experiments on global registration. All errors are plotted over the number of iterations, i.e., the growing size of the UGV map. Here, we indicate experimental configurations as follows: *SHOT*, *FPFH* and *ESF* denote the used descriptors, an additional *FGR* denotes if we used the fast global optimization instead of the RANSAC-based outlier filtering, and we add *gr* for cases in which also ground removal was performed before descriptor extraction. The *ICP* solution is illustrated for the basic experiments.

Configuration	M_b	M_i	M_c	D_b	D_i	D_c
FPFH	1	45	60	2	N/A	N/A
FPFH gr	N/A	N/A	N/A	N/A	N/A	N/A
FPFH FGR	1	50	54	3	N/A	N/A
FPFH FGR gr	60	60	43	3	N/A	N/A
SHOT	1	10	36	1	1	3
SHOT gr	1	1	36	1	1	7
SHOT FGR	1	35	35	1	12	N/A
SHOT FGR gr	1	29	28	1	12	N/A
FPFH seg	N/A	N/A	N/A	N/A	N/A	N/A
SHOT seg	N/A	N/A	N/A	N/A	N/A	N/A
ESF seg	N/A	N/A	N/A	N/A	N/A	N/A

TABLE II: Minimal number of LiDAR scans for successful global registration experiments. Here, M_b, M_i, M_c, denote the basic, intermediate, and complex experiment on the Montelibretti dataset, while D_b, D_i, and D_c denote the different experimental set-ups on the Dortmund data. The UGV travels on average $0.8m$ between two scans.

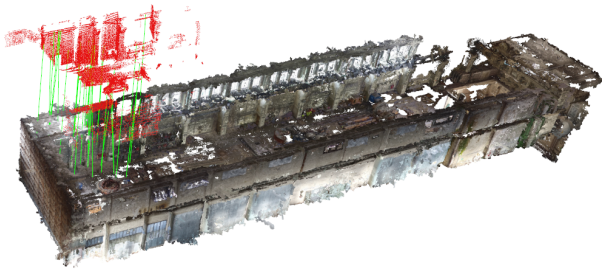


Fig. 5: Resulting global localization between *3D* UGV sub-map (red point-cloud) and global *3D* UAV map (coloured point-cloud). Green lines indicate the resulting descriptor matches associated to points in the two point-clouds. The data stems from the Dortmund indoor dataset and drawn from the complex experimental set-up.

1) Timings: Additionally to the evaluation of residuals, computation times are an important factor in the choice of algorithms in robotics. Table III lists the computational times of the four main components of the global registration algorithms when executed on an Intel i7-4600U CPU @ 2.10GHz. The computation times are reported per UGV LiDAR scan which has an acquisition time of $3s$ on the considered platform.

Although our focus was not on maximizing computational efficiency, all approaches can be performed in this time window and therewith in real-time. We are confident that they can be further improved to also yield faster processing times. With key-point detection times increasing with the amount of points in the point-cloud, the segmentation approach is the fastest in the first step. Descriptor extraction, is fastest for SHOT descriptors, also scaling with the amount of points. However, the largest contribution to the computational time has the descriptor matching which is longest for SHOT and low for FPFH. For the segmentation, we report the timings for the high-dimensional ESF features and achieve low timings, due to the compact representation of segments. Finally, RANSAC-based geometric consistency is slower than the optimization-based FGR.

E. Discussion

The global registration of *3D* UAV and UGV point-cloud data is a difficult problem. Based on our evaluation, the most general solution that we devise is a key-point descriptor matching algorithm using SHOT descriptors. In our evaluation, FPFH descriptors performed well, when the overlap between the maps is large, but failed for the more complex experiments, and showed to be sensitive to ground removal.

The segmentation showed to not deliver satisfying results, as it requires repeatable ground removal and segmentation, which could not be achieved in the considered configurations and scenarios. Key-point detection on the other hand performed well.

SHOT descriptor matching is computationally more expensive than FPFH due to high descriptor dimensionality, but showed best performance throughout. The processing time can be speeded up by removing the ground in environments that allow for reliable ground removal.

Finally FGR can yield additional speed up of the transformation estimation. Yet, when using FGR, the cost function must be carefully considered as the technique is prone to converge to local minima. While the RANSAC-based transformation estimation takes generally longer than FGR the robustness to local minima is greatly increased. Also, the additional computational time for RANSAC was negligible in our experiments.

VI. Conclusion

This paper presented global registration algorithms for UGV and UAV point-clouds generated from heterogeneous sensors, i.e., LiDAR sensors for UGVs and cameras for UAVs, and drastically different view-points. The registration algorithm is based on geometrical descriptor matching. The approach was integrated with a full SaR robotic mapping system, bridging the gap between effective exploitation of UAV mapping data on UGVs. We evaluated several different *3D* descriptor-based registration techniques and identify the best performing approach for the problem of global point-cloud registration from heterogeneous sensors in SaR scenarios.

Future avenues of research could include point-cloud registration by using further informative cues than the geometrical information alone for data registration between the sensor modalities. This could benefit runtime and compactness of point-cloud description of the proposed algorithm.

VII. Acknowledgement

This work was supported by European Union's Seventh Framework Programme for research, technological development and demonstration under the TRADR project No. FP7-ICT-609763.

References

[1] S. Saeedi, M. Trentini, M. Seto, and H. Li, "Multiple-robot simultaneous localization and mapping: A review," *JFR*, vol. 33, no. 1, pp. 3–46, 2016.

Module	*FPFH*	*FPFH gr*	*FPFH FGR*	*FPFH FGR gr*	*SHOT*	*SHOT gr*	*SHOT FGR*	*SHOT FGR gr*	*Seg*
Key-point / Segmentation	35.97 ±13.17	22.92 ±13.30	39.01 ±16.04	24.57 ±14.90	33.1 ±12.16	18.97 ±11.07	35.76 ±13.30	23.06 ±14.87	9.32 ±5
Description	228.87 ±61.99	122.59 ±39.27	242.52 ±64.34	136.63 ±43.91	17.21 ±8.82	12.30 ±8.60	17.35 ±8.51	13.52 ±9.52	105.91 ±50.72
Matching	293.32 ±8.32	86.04 ±27.41	291.76 ±6.49	88.33 ±25.25	2897.94 ±44.01	1992.17 ±40.36	2937.60 ±65.13	2099.55 ±42.31	429.11 ±51.15
Geometric consistency / Optimization	15.91 ±11.54	22.72 ±17.28	2.38 ±1.09	2.05 ±1.05	7.55 ±6.11	8.24 ±6.79	2.23 ±0.99	1.82 ±0.94	0.80 ±1.60
Total	574.07	254.27	575.67	251.58	2926.01	2031.68	2992.94	2137.95	545.14

TABLE III: Mean computation times and standard deviations of the individual approaches in the complex Montelibretti experiment per LiDAR scan in ms as computed on an Intel i7-4600U CPU @ 2.10GHz.

[2] N. Michael, S. Shen, K. Mohta, Y. Mulgaonkar, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida *et al.*, "Collaborative mapping of an earthquake-damaged building via ground and aerial robots," *JFR*, vol. 29, no. 5, pp. 832–841, 2012.

[3] B. Kim, M. Kaess, L. Fletcher, J. J. Leonard, A. Bachrach, N. Roy, and S. Teller, "Multiple Relative Pose Graphs for Robust Cooperative Mapping," in *ICRA*, 2010, pp. 3185–3192.

[4] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3dmatch: Learning local geometric descriptors from rgb-d reconstructions," in *CVPR*, 2017.

[5] R. Dubé, A. Gawel, J. Nieto, R. Siegwart, and C. Cadena, "Online multi-robot slam with 3d lidars: A full system," in *IROS*, 2017.

[6] S. Fuhrmann, F. Langguth, and M. Goesele, "Mve-a multi-view reconstruction environment." in *GCH*, 2014, pp. 11–18.

[7] A. Birk and S. Carpin, "Merging occupancy grid maps from multiple robots," *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1384–1397, 2006.

[8] J.-L. Blanco, J. González-Jiménez, and J.-A. Fernández-Madrigal, "A robust, multi-hypothesis approach to matching occupancy grid maps," *Robotica*, vol. 31, no. 05, pp. 687–701, 2013.

[9] S. Saeedi, L. Paull, M. Trentini, and H. Li, "Multiple robot simultaneous localization and mapping," in *IROS*, 2011, pp. 853–858.

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

[11] C. Forster, M. Pizzoli, and D. Scaramuzza, "Air-ground localization and map augmentation using monocular dense reconstruction," in *IROS*, 2013, pp. 3971–3978.

[12] T. Hinzmann, T. Stastny, G. Conte, P. Doherty, P. Rudol, M. Wzorek, E. Galceran, R. Siegwart, and I. Gilitschenski, *Collaborative 3D Reconstruction Using Heterogeneous UAVs: System and Experiments*. Cham: Springer International Publishing, 2016, pp. 43–56.

[13] R. W. Wolcott and R. M. Eustice, "Visual localization within lidar maps for automated urban driving," in *IROS*, 2014, pp. 176–183.

[14] A. Gawel, T. Cieslewski, R. Dubé, M. Bosse, R. Siegwart, and J. Nieto, "Structure-based vision-laser matching," in *IROS*, 2016, pp. 182–188.

[15] R. Dubé, D. Dugas, E. Stumm, J. Nieto, R. Siegwart, and C. Cadena, "*SegMatch*: Segment based place recognition in 3d point clouds," in *ICRA*, 2017.

[16] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *ECCV*, 2016, pp. 766–782.

[17] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An efficient probabilistic 3d mapping framework based on octrees," *AuRo*, vol. 34, no. 3, pp. 189–206, 2013.

[18] R. Dubé, A. Gawel, C. Cadena, R. Siegwart, L. Freda, and M. Gianni, "3d localization, mapping and path planning for search and rescue operations," in *SSRR*, 2016, pp. 272–273.

[19] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "isam2: Incremental smoothing and mapping using the bayes tree," *IJRR*, vol. 31, no. 2, pp. 216–235, 2012.

[20] Y. Zhong, "Intrinsic shape signatures: A shape descriptor for 3d object recognition," in *ICCV*, 2009, pp. 689–696.

[21] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *ICRA*, 2009, pp. 3212–3217.

[22] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *ECCV*, 2010, pp. 356–369.

[23] W. Wohlkinger and M. Vincze, "Ensemble of shape functions for 3d object classification," in *ROBIO*, 2011, pp. 2987–2992.

[24] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration." *VISAPP (1)*, vol. 2, no. 331-340, p. 2, 2009.