Collaborative Localization of Aerial and Ground Robots through Elevation Maps

Roman Käslin[1], Péter Fankhauser[1,2], Elena Stumm[1], Zachary Taylor[1], Elias Mueggler[3], Jeffrey Delmerico[3], Davide Scaramuzza[3], Roland Siegwart[1], Marco Hutter[2]

***Abstract*—Collaboration between aerial and ground robots can benefit from exploiting the complementary capabilities of each system, thereby improving situational awareness and environment interaction. For this purpose, we present a localization method that allows the ground robot to determine and track its position within a map acquired by a flying robot. To maintain invariance with respect to differing sensor choices and viewpoints, the method utilizes elevation maps built independently by each robot's onboard sensors. The elevation maps are then used for global localization: specifically, we find the relative position and orientation of the ground robot using the aerial map as a reference. Our work compares four different similarity measures for computing the congruence of elevation maps (akin to dense, image-based template matching) and evaluates their merit. Furthermore, a particle filter is implemented for each similarity measure to track multiple location hypotheses and to use the robot motion to converge to a unique solution. This allows the ground robot to make use of the extended coverage of the map from the flying robot. The presented method is demonstrated through the collaboration of a quadrotor equipped with a downward-facing monocular camera and a walking robot equipped with a rotating laser range scanner.**

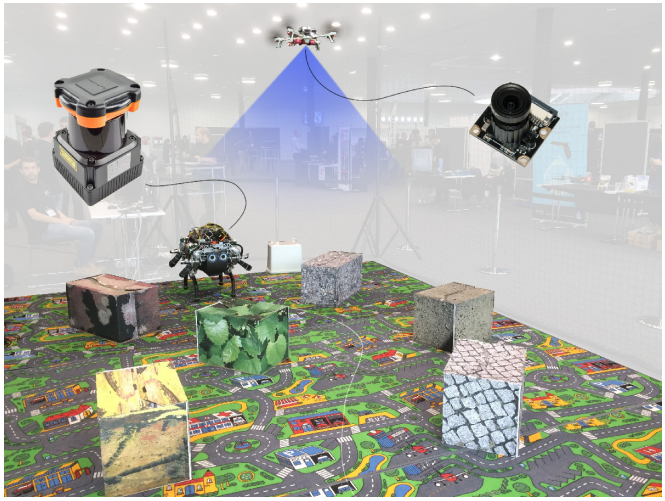


Fig. 1. Collaborative localization is demonstrated with a quadrotor carrying a monocular camera and a walking robot equipped with a rotating laser range sensor. Localization between the robots is achieved by matching elevation maps generated from the onboard sensors. The proposed method performs reliably even for different perspectives of the robots and for incomplete maps due to occlusion from obstacles.

I. Introduction

Ground robots may carry substantial payloads and can actively interact with their environment. In a search and rescue mission they could be sent into settings that are too dangerous for humans, reducing the time required to reach victims by removing the need to first secure the area. However, the operator only receives limited information about the ground robot's surroundings due to its low viewpoint. Other objects at the height of the robot may block the operators view, increasing the difficulty of navigating within hazardous environments.

A flying robot, on the other hand, can provide a situational assessment of the environment by its ability to quickly cover large areas with its bird's-eye view. This data enables global navigation of the ground robot in potentially unknown and challenging terrain. Therefore, working in a heterogeneous team of flying and ground robots enhances the capabilities of robots to support human rescue teams.

In this work we introduce a method for ground robots to localize within a map created by a flying robot. This is essential as the map provided by the flying robot is only of use to the ground robot if its own location within the map is known. The presented approach assumes sufficient coverage by the flying robot for use as a reference map in which to localize. Our method is independent of the viewpoint and the sensors of both robots, given that they provide an elevation map of their surroundings. In addition, both elevation maps may be updated continuously.

A. Related Work

Prior work on the collaboration between aerial and ground robots often performs the entire perception process on one robot and guides the other robot through the mission. For example, a flying robot can track a ground robot using a QR-code [1], visual markers [2], [3], or edge detection [4]. Other work uses a camera on a ground robot to track LEDs on a flying robot [5], [6].

Additionally, many approaches exist to collaboratively navigate in large outdoor environments with at least partial availability of GPS [7]–[9]. As a consequence, these approaches are not applicable in indoor scenarios and in GPS denied locations such as forests or urban street canyons. In addition, even if a GPS signal is available, the position is often inaccurate and provides no heading direction.

The authors are with the [1]Autonomous Systems Lab, ETH Zurich, the [2]Robotics Systems Lab, ETH Zurich, and the [3]Robotics and Perception Group, University of Zurich, Switzerland (rkaeslin@ethz.ch).

On the other hand, methods to globally localize planetary rovers in the absence of GPS exist for space exploration. For this purpose, visually detectable landmarks [10] or generated surface elevation maps of the rover surroundings are matched to a global map. The elevation maps are therefore searched for topographic peaks [11] or get compared directly by zero-mean normalized cross-correlation [12].

In [13], a framework for the collaborative navigation of a flying and a walking robot is described. In the presented scenario both robots use a camera to perceive the environment and rely on visual features to find their relative positions. As a consequence, the approach assumes that the environment provides appropriate texture, and that each robot has a similar viewpoint, in order to detect visual correspondences.

In the work of Michael et al. [14] a flying robot complements the map of a ground robot using an initial guess of the relative robot position to perform an iterative closest point (ICP) method. Thereby, the flying robot takes off on top of the ground robot and both robots use laser range scanners to detect their environment.

In contrast to these works, we contribute a global localization and tracking method that is independent of the observation viewpoint of the robots, external localization (such as GPS), and textured environments. The method does not depend on a precise initial guess and can combine depth sensing from different sensors (e.g. cameras, laser range sensor, etc.). For the method to succeed, no walls or ceilings are necessary, which is of high importance for outdoor applications. After an initialization period of a few seconds, tracking of the ground robot is performed in real-time using a particle filter [12], [15].

Similar to the work of Forster et al. [15] on the localization of a quadrotor within a given map of a ground robot, we use 2.5D elevation maps generated by the robots to compute the relative position. We extend the work in [15] by evaluating different similarity metrics. Additionally, we do not make any assumption on the heading direction since the use of magnetic north is often unreliable within buildings and can be distorted by onboard and external electronics.

The congruence of such maps may be determined by pixel-based methods, by entropy, or Fourier-based methods. In medical imaging, the entropy-based mutual information is broadly applied for multimodal image registration [16]. An example of a Fourier-based method is the Fourier-Mellin transform, which allows for the estimation of position and orientation, but needs strong smoothing of the borders [17], [18].

B. Paper Outline

Our goal is to identify the best method for finding the relative position of the aerial and ground robot. Therefore, we compare the merit of the following similarity measures: sum of absolute differences (SAD), sum of squared differences (SSD), normalized cross-correlation (NCC), and normalized mutual information (NMI). In the case of Gaussian noise, SSD would be optimal. However, we compare these similarity measures on two real datasets of the environments shown in Fig. 1 and 7, where the assumption of Gaussian noise may not hold.

We describe the experimental setup in Sec. II. Our evaluations show that with all similarity measures presented in Sec. III we are able to track the robot's position in real-time using a particle filter introduced in Sec. IV. The results in Sec. V illustrate that the motion of the robot can be successfully used to converge to a unique solution. Finally, the work is concluded in Sec. VI.

II. Experimental Setup

In our experiments we use a lightweight quadrotor equipped with a monocular downward-facing camera [19] (Fig. 1). As a ground robot we use a walking robot due to its superior ability to overcome rough terrain often found in search and rescue missions. The walking robot StarlETH [20] uses a laser range scanner that maps the local surroundings and allows local foothold planning.

The map gathered by the quadrotor is created with the dense 3D reconstruction pipeline (REMODE) presented in [19], [21]. In this approach, the quadrotor performs real-time monocular dense depth estimation for a set of keyframes that cover the mapped area. These individual depth maps are integrated into an elevation map based on the quadrotor's pose estimate from visual odometry [22]. The metric scale is guaranteed through initialization with a single-point distance sensor and inertial measurements. The walking robot fuses several measurements of its rotating laser range scanner with the legged state estimation [23]. As a result, the method [24] allows for the computation of the variance of multiple measurements.

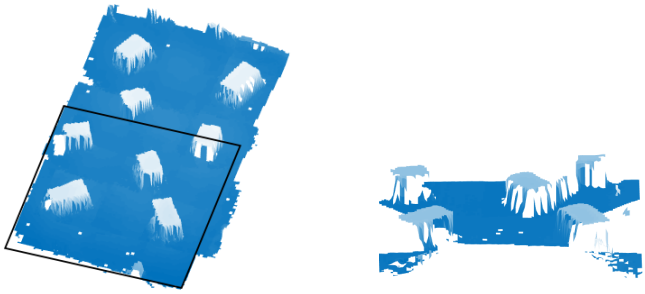
Although we used this particular setup, our method is intended to work for any other combination of robots with depth sensing capability.

III. Global Localization

Without any initial position given, a global search has to be performed in order to localize the ground robot within the map of the environment provided by the flying robot.

Objects in the environment, especially taller ones, typically obstruct the view of the ground robot, causing occlusions in its map. On the other hand, the flying robot with its downward-facing camera has difficulties in mapping vertical surfaces. As a result, the maps may not look alike and each may contain data that is missing from the other, as can be seen in Fig. 2. Additionally, the maps may not overlap fully for all times considered. Due to the challenges caused by viewpoint differences, occlusions, repetitive structures, and noisy sensor data, we believe that local feature based approaches (2D or 3D) cannot be applied in this situation as pointed out in [15], [25].

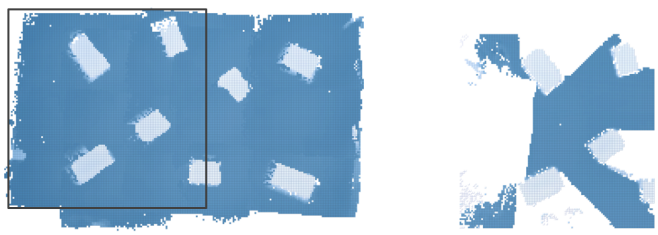
Our method is therefore based on comparing 2.5D elevation maps in a template matching approach. The resulting template search is performed within the map of the flying robot. Therefore, we will refer to the map of the flying robot as the *reference* and to the map of the ground robot as the *template* for the rest of this work.



(a) Map perceived by the flying robot

(b) Map of the ground robot

Fig. 2. The two robots perceive their surroundings from a vastly different viewpoint using two different types of sensors. The black rectangle in (a) marks the area perceived by the ground robot.



(a) Reference (elevation map of the flying robot)

(b) Template (elevation map of the ground robot)

Fig. 3. Using the gravity vector provided by the IMU of the robots, the maps are projected onto the ground plane. This results in a simple, sensor- and viewpoint-independent representation.

A. Elevation Maps

Utilizing the fact that both robots use an inertial measurement unit (IMU) for their state estimation, we project the maps onto the ground plane along the provided gravity vector. The resulting elevation maps, shown in Fig. 3, are represented by 2D grids [24], [26]. They are generated in a way that, if multiple points are projected on the same grid cell, the highest one is taken. This allows the closest representation to the downward-facing camera of the flying robot.

The elevation maps shown in Fig. 3 provide a simple representation of the observed surroundings and allow for the comparison of different sensors with different viewpoints. An important property of the elevation maps is that they are correctly scaled, which is given in our scenario by the use of a laser range scanner and visual-inertial odometry.

B. Template Matching

In the global localization stage the similarity for all considered template positions and orientations $\mathbf{u} = \{u_1, \ldots, u_N\}$ are computed. Every state $u_i = [x_i, y_i, \theta_i]$ describes the relative position $[x, y]$ and orientation θ of the ground robot in the 2D coordinate system of the reference map.

For practical reasons, we aim for a short initialization time of around 5 s for an environment of 5 m × 5 m. Taking into account the computation time of one sample of $\mathbf{u}$, we determine the number of possible samples to calculate. These samples are then spread evenly over the entire dataset. As a result, we utilize the center of every 5×5 patch and a 5 degree increment for the global localization. Additionally, the similarity scores are computed sparsely using the center of every 5×5 patch.

Given a certain position and orientation u of the template, the similarity is calculated. For this purpose, the sparsely selected points of the template are projected onto the reference and compared to the height of the nearest cell. We define all points used to compute the similarity as $X(u) = X_t \cap X_r(u)$ with X_t the sparsely selected points with a defined height in the template and $X_r(u)$ the points with defined height in the reference for the given template position and orientation u.

To prevent bias in the similarity metrics causing them to favor areas of low overlap, a threshold of a minimum overlap between reference and template of 25% is applied. Matches with an overlap lower than the threshold are discarded.

C. Similarity Calculation

The similarity is computed using different similarity measures. These are implemented as zero-mean to be independent of the z-position of the maps (except for the normalized mutual information, which is independent of the absolute values). Having found the best similarity, the z-alignment is recovered by the difference of the mean heights $\Delta z = \hat{H}_r(\tilde{u}) - \hat{H}_t$ with $\hat{H}_r$ and $\hat{H}_t$ the mean of the heights of the template $H_t(\mathbf{x})$ and of the reference $H_r(\mathbf{x} + \tilde{u})$ of all points $\mathbf{x} \in X(\tilde{u})$ and $\tilde{u}$ being the best position and orientation found.

To reduce the influence of uncertain and erroneous measurements on the obtained score a weight inversely proportional to the variance of the heights is used [27]. Since our data only provides the variance for the heights in the template, we only include these. This could nevertheless be extended to the reference. Furthermore, to be independent of the number of points in X, a normalized form is used.

The implemented similarity measures are:

1) Sum of Absolute Differences (SAD): As stated in [28] the SAD is relatively robust to noise due to its moderate penalty on height differences. The zero-mean normalized and weighted SAD is computed by

$$\mathrm{SAD}(u) = \alpha \sum_{\mathbf{x} \in X(u)} \frac{\left| \left(H_t(\mathbf{x}) - \hat{H}_t\right) - \left(H_r(\mathbf{x} + u) - \hat{H}_r(u)\right) \right|}{\mathrm{Var}\left(H_t(\mathbf{x})\right)} \tag{1}$$

with normalization $\alpha = \sum_{\mathbf{x} \in X(u)} \mathrm{Var}\left(H_t(\mathbf{x})\right)$. The best relative position and orientation is then found by

$$\tilde{u} = \arg\min_u \mathrm{SAD}(u) \ . \tag{2}$$

2) Sum of Squared Differences (SSD): An unweighted variant of the SSD is used in in the work of Forster et al. [15]. The zero-mean weighted and normalized SSD is defined as

$$\mathrm{SSD}(u) = \alpha \sum_{\mathbf{x} \in X(u)} \left(\frac{\left(H_t(\mathbf{x}) - \hat{H}_t\right) - \left(H_r(\mathbf{x} + u) - \hat{H}_r(u)\right)}{\mathrm{Var}\left(H_t(\mathbf{x})\right)} \right)^2 \tag{3}$$

with normalization $\alpha = \sum_{\mathbf{x} \in X(u)} \mathrm{Var}\left(H_t(\mathbf{x})\right)^2$. The best relative position and orientation is then found at the minimum

$$\tilde{u} = \arg\min_u \mathrm{SSD}(u) \ . \tag{4}$$

3) Normalized Cross-Correlation (NCC): In the work of Wendel et al. [29] a normalized cross-correlation is used to refine the alignment of two 3D models. Including the proposed weight, the zero-mean normalized and weighted NCC [30] is calculated by

$$\mathrm{NCC}(u) = \alpha \sum_{\mathbf{x} \in X(u)} \frac{\left(H_t(\mathbf{x}) - \hat{H}_t\right) \cdot \left(H_r(\mathbf{x}+u) - \hat{H}_r(u)\right)}{\mathrm{Var}\left(H_t(\mathbf{x})\right)} \tag{5}$$

with normalization

$$\alpha = \frac{1}{\sqrt{\sum_{\mathbf{x} \in X(u)} \frac{\left(H_t(\mathbf{x}) - \hat{H}_t\right)^2}{\mathrm{Var}(H_t(\mathbf{x}))} \cdot \sum_{\mathbf{x} \in X(u)} \frac{\left(H_r(\mathbf{x}+u) - \hat{H}_r(u)\right)^2}{\mathrm{Var}(H_t(\mathbf{x}))}}} \, .$$

The best relative position and orientation is then found at the maximum correlation

$$\tilde{u} = \arg\max_u \mathrm{NCC}(u) \, . \tag{6}$$

4) Normalized Mutual Information (NMI): We propose to use the Shannon entropy based Mutual Information (MI) introduced in [28]. In its basic form the MI is strongly biased towards the borders of the reference map since in the case of low numbers of samples the approximate MI value calculated by the utilized histogram can show a bias to minimizing the number of samples. Therefore, we provide the same number of points for every position and orientation of the template considered (possible due to the sparse similarity calculation). Additionally, we use the normalized version proposed in [31] to eliminate the influence of entropy variance in the reference map. In opposition to the other similarity measures used, no weight based on the variance of the heights in the template is included. The normalized mutual information is defined as

$$\mathrm{NMI}(u) = \frac{E_t(u) + E_r(u)}{E_{tr}(u)} \tag{7}$$

with marginal entropies of the template t and reference r

$$E_i(u) = \sum_{b \in B} -p_i\left(b|u\right) \cdot \log_2\left(p_i\left(b|u\right)\right) \qquad i \in \{t, r\} \tag{8}$$

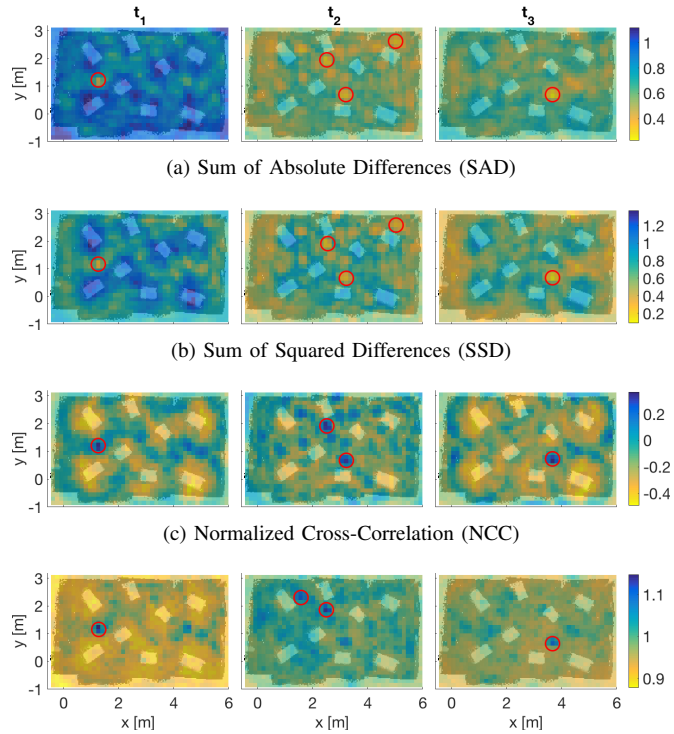
and joint entropy

$$E_{tr}(u) = \sum_{b_t \in B} \sum_{b_r \in B} -p_x\left(b_t, b_r|u\right) \cdot \log_2\left(p_x\left(b_t, b_r|u\right)\right) \, . \tag{9}$$

Whereas p_i describes the probability of all defined heights being within bin b and p_x represents the probability of the heights in t and r to be within the bins b_t and b_r simultaneously for the same physical location. The bins used have a fixed height of $H_b = 0.08\,\mathrm{m}$ resulting in the number of bins $N_b = (H_{max} - H_{min})/H_b$ and $B = \{b_1, b_2, \ldots, b_{N_b}\}$. The best relative position and orientation is then found at the maximum mutual information

$$\tilde{u} = \arg\max_u \mathrm{NMI}(u) \, . \tag{10}$$

The resulting scores of the implemented similarity measures on the dataset with the boxes are shown in Fig. 4 for robot positions at three different times t_1, t_2, and t_3. It can be seen that the minima of the SAD and SSD as well as the maxima of the NCC and NMI are not unique for the position at t_2. However, taking into account the previous iterations, the reliability of the scores is sufficient to identify the correct position.



(a) Sum of Absolute Differences (SAD)

(b) Sum of Squared Differences (SSD)

(c) Normalized Cross-Correlation (NCC)

(d) Normalized Mutual Information (NMI)

Fig. 4. The costmaps illustrate the scores of the different similarity measures at t_1: 25%, t_2: 50%, and t_3: 75% of the trajectory for the best orientation at every relative position overlayed on the height map. The minima of the SAD and SSD as well as the maxima of the NCC and NMI are not unique for all time instances indicated by multiple yellow (SAD, SSD) and blue (NCC, NMI) peaks (strong minima, respectively maxima are highlighted in red).

IV. Particle Filter

To allow us to track multiple hypotheses of relative locations we propose to use a particle filter, similar to the Monte Carlo Localization in [15] and introduced in [32]. The motion of the ground robot is then used to converge to the correct location.

The set of particles $U = \{u_1, u_2, \ldots, u_{N_P}\}$ represents the belief of the position and orientation of the ground robot within the reference map. Given that we do not have prior information about the relative location of the robots, the particles are for practical reasons initialized as the sparse global localization presented in Sec. III, which differs from the random initialization of the standard particle filter. Subsequently, we compute the similarity scores for all particles and identify the best score obtained. If this score is indicating a good match, the particles are resampled. Otherwise, another global localization is performed, as shown in Fig. 5.

For the resampling, we want to guarantee that particles with higher scores are sampled with a higher probability than others. We achieve this by converting the different similarity scores into a belief representing probability

$$p(u) \propto \exp\left(\frac{S(u)}{\rho_S}\right) \tag{11}$$

where S represents the different similarity scores used. The parameter ρ_S depends on the resolution of the particu-

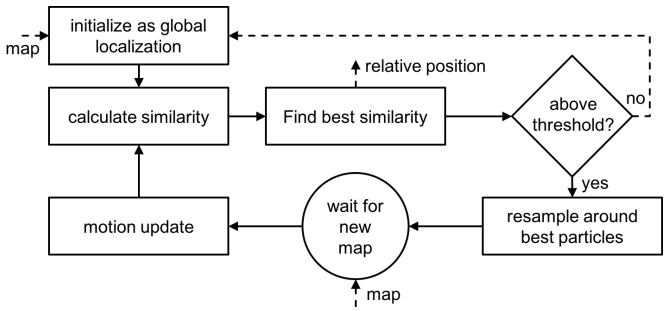


Fig. 5. The particle filter performs a sparse global localization as initialization, then resamples particles with a dense resolution around the best similarity scores and shifts them by the motion estimate of the robot. In the next iteration it computes the similarity of all locations represented by particles for a given map at the new position.

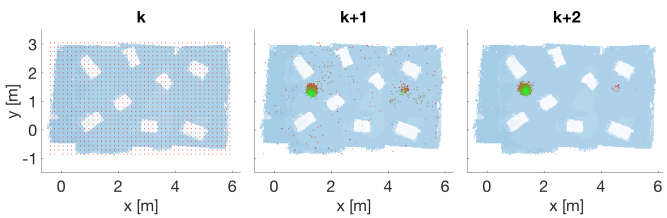


Fig. 6. The particles converge to the correct position within 2 iterations of the particle filter starting from a sparse initialization at iteration k using NMI.

lar similarity score and is chosen as $\rho_{SAD} = -0.0025$, $\rho_{SSD} = -0.0004$, $\rho_{NCC} = 0.025$ and $\rho_{MI} = 0.015$. The parameters are chosen to guarantee fast convergence, while still providing enough robustness to keep track of multiple hypotheses.

We resample N_P particles with probability p for all iterations of the particle filter. Normally distributed noise (standard deviations $\sigma_{\text{pos}} = 0.02\,\text{m}$ and $\sigma_{\text{rot}} = 2°$) is added to the resampled particles to get to a higher resolution and increase the robustness. To speed up the computation performance, we discretize the resampling step with the resolution of the elevation map for translation and a one degree increment for rotation. Higher resolutions did not result in significantly more accurate results but increased the computation time. After resampling, all duplicate particles are removed to save computation time in the next iteration $k+1$.

When a new map is provided by the ground robot every particle i is shifted by

$$\hat{u}_{i,k+1} = u_{i,k} + \Delta u \tag{12}$$

with Δu the motion estimation of the robot, provided by the proprioceptive state estimation of the walking robot [23]. Then the process is restarted by computing the similarity scores for all resampled particles.

To further increase the robustness and the recovery capability, randomly sampled particles could be added in every iteration. In our experiments this was not necessary since the particle filter never lost track of the correct location. Moreover, the particles converged to the correct location after only two to three iterations given motion of the ground robot, as can be seen in Fig. 6.

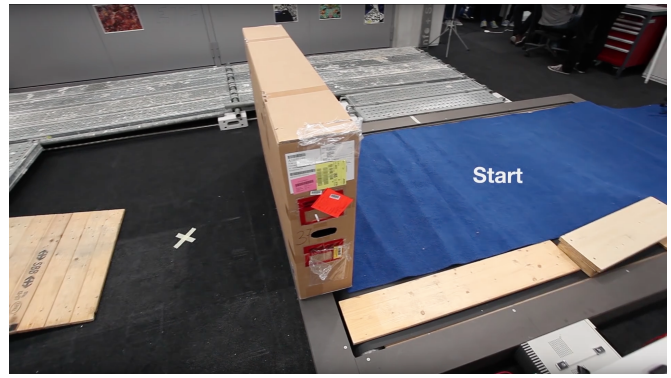


Fig. 7. In dataset 2 the robot walks around a cardboard box down on a ramp to its goal position using a map to plan its path, which was previously created by a flying robot.

V. Results

The localization method introduced in this work was successfully verified within two different environments, shown in Fig. 1 and 7. The datasets consist of the elevation map of both robots, of which the one of the ground robot includes variance information of the heights. The ground truth reference trajectory for the walking robot was acquired by first manually determining the initial pose of the robot. Subsequently, ICP localization was applied while deliberately making use of walls and ceiling to achieve a tracking accuracy of $< 0.01\,\text{m}$.

The computation is run on a single core of an Intel Core i7 620-M 2.66 GHz processor. For our experiments the number of particles is set to $N_P = 4000$. It is chosen such that real-time performance is achieved while densely covering the space around the best particles. As a consequence, the particle filter is able to process all map updates from the walking robot with a refresh rate of 0.5 Hz once initialized. All similarity measures take roughly the same amount of time to compute.

Fig. 8 shows the restored trajectory of the location with the best score for each of the similarity measures in the environment with the boxes compared to the ground truth. All similarity measures correctly recover the trajectory of the walking robot. The initial global localization takes less than 4 s for the dataset of the first environment of approximately $6\,\text{m} \times 4\,\text{m}$ and a resolution of 0.03 m per cell of the elevation maps.

The results of the restored trajectories in the second environment used to validate our method are shown in Fig. 9. The dataset has a resolution of 0.04 m per cell and has a size of $9\,\text{m} \times 6\,\text{m}$. The introduced localization method succeeds in tracking the trajectory with all similarity measures for this new environment without adapting any parameters. The initialization takes 10 s. For this environment the trajectory recovered by the NCC starts at the wrong position and is offset towards the right at the beginning of the trajectory due to multiple hypotheses. The particle filter handles the

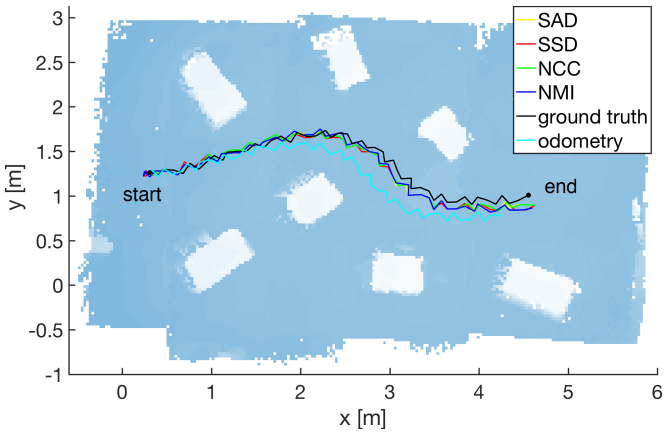


Fig. 8. The proposed method recovers the trajectory of the walking robot with all similarity measures on dataset 1 (Fig. 1). The zig-zag motion of the walking robot is a result of the shifting of the weight of the body, required during stepping.

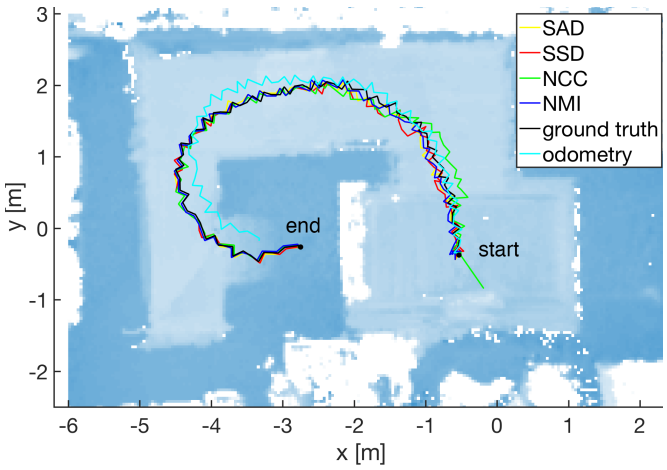


Fig. 9. For dataset 2 (Fig. 7) the recovered trajectory with the NCC starts at the wrong position and is offset at the beginning due to multiple hypotheses. The correct position is identified with all other similarity measures and subsequently with the NCC by covering the area around the hypotheses with a higher resolution and taking the motion of the robot into account.

situation by covering the area around the hypotheses with a higher resolution and considering the motion of the robot. Therefore, the position is correctly identified in the subsequent iterations.

The root-mean-square errors (RMSE) for position and orientation over the entire recovered trajectories are listed in Tab. I. Note that in our implementation with a discretized elevation map and 4000 particles, the results behave nearly deterministically. We observe that all similarity measures perform comparably (within the accuracy of the ground truth reference trajectory) except for the NCC on the second dataset. We suspect that the ability of the NCC to cope with other linear dependencies such as offset and scale changes (which do not occur in the test scenarios) makes the results more susceptible to noise than the other methods (see the trajectory offset in Fig. 9). In comparison to the trajectory restored by odometry, all similarity scores provide much more accurate results using the proposed method (Tab. I).

TABLE I
ROOT-MEAN-SQUARE ERROR OF THE RECOVERED TRAJECTORIES

	SAD	SSD	NCC	NMI	odometry
dataset 1	0.075 m	0.076 m	0.075 m	0.080 m	0.232 m
(boxes)	1.58°	1.61°	1.48°	1.22°	1.17°
dataset 2	0.070 m	0.077 m	0.111 m	0.063 m	0.471 m
(ramp)	1.69°	2.52°	2.44°	1.64°	16.72°

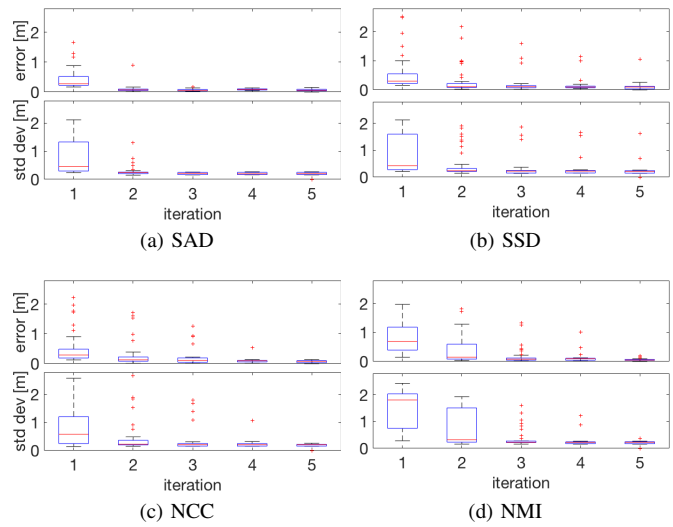


Fig. 10. The particles converge in two to three iterations to the correct position for all similarity measures. The standard deviation provides a good indicator on the convergence of the particles.

Additionally, we tested the particle filter on 40 sequences of 5 sequential elevation maps registered by the walking robot of both environments. In Fig. 10 the position error of the mean of all particles and the standard deviation of the particles are shown for 5 iterations of the particle filter. Notice that after 3 iterations the mean of all particles is in at least 90% of the experiments closer than 0.2 m to the correct position. Our approach is therefore capable of initializing at any position along the trajectories and converging qucikly to the correct result, which proves the high robustness of the method.

Although the accuracy of the alignment was further increased by reducing the sparsity of the similarity computation at the cost of higher computation times, the algorithm can still run in real-time. The standard deviation can then be considered as a measure for the convergence of the particles.

VI. CONCLUSION

We have presented a localization method based on elevation maps that allows a ground robot to determine its position within the reference map of a flying robot, independently of the sensors used by the robots. The method assumes some variability of elevation within the environment and is therefore not applicable in completely flat surroundings.

It was shown that the method is effective in two different indoor environments using four different similarity scores.

Weighting of the height information with the variance of its measurements helped to reduce the influence of noisy measurements. The sparsity introduced for the global localization keeps the computation time in the range of several seconds. After the initialization step, the particle filter allows real-time performance. Especially in self-similar environments the particle filter is essential and uses the motion of the ground robot for convergence. In our experiments, all similarity metrics performed similarly and additional work on different degraded environments is necessary to make a recommendation on a similarity measure for broader use.

In future work, we intend to evaluate our method in larger outdoor environments using GPS to focus the initial global search. Furthermore, we plan to add a measure of certainty to the resulting localization.

REFERENCES

[1] E. Mueggler, M. Faessler, F. Fontana, and D. Scaramuzza, "Aerial-guided navigation of a ground robot among movable obstacles," *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 1–8, 2014.

[2] E. Harik, F. Guerin, F. Guinand, J.-F. Brethe, and H. Pelvillain, "UAV-UGV cooperation for objects transportation in an industrial area," *IEEE International Conference on Industrial Technology (ICIT)*, 2015.

[3] G. Heppner, A. Roennau, and Dillman, "Enhancing sensor capabilities of walking robots through cooperative exploration with aerial robots," *Journal of Automation Mobile Robotics and Intelligent Systems*, vol. 7, no. 2, 2013.

[4] M. Garzón, J. Valente, D. Zapata, and A. Barrientos, "An Aerial-Ground Robotic System for Navigation and Obstacle Mapping in Large Outdoor Areas," *Sensors*, vol. 13, no. 1, pp. 1247–1267, 2013.

[5] P. Rudol, M. Wzorek, G. Conte, and P. Doherty, "Micro Unmanned Aerial Vehicle Visual Servoing for Cooperative Indoor Exploration," *IEEE Aerospace Conference*, 2008.

[6] M. Faessler, E. Mueggler, K. Schwabe, and D. Scaramuzza, "A monocular pose estimation system based on infrared leds," *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[7] T. Stentz, A. Kelly, H. Herman, P. Rander, O. Amidi, and R. Mandelbaum, "Integrated Air/Ground Vehicle System for Semi-Autonomous Off-Road Navigation," *Robotics Institute*, 2002.

[8] B. Grocholskya, S. Bayraktara, V. Kumar, and G. Pappas, "UAV and UGV Collaboration for Active Ground Feature Search and Localization," in *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, 2004.

[9] T. A. Vidal-Calleja, C. Berger, J. Solà, and S. Lacroix, "Large scale multiple robot visual mapping with heterogeneous landmarks in semi-structured terrain," *Robotics and Autonomous Systems*, vol. 59, no. 9, pp. 654–674, 2011.

[10] E. Boukas, A. Gasteratos, and G. Visentin, "Towards orbital based global rover localization," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[11] P. J. Carle, P. T. Furgale, and T. D. Barfoot, "Long-range rover localization by matching LIDAR scans to orbital elevation maps," *Journal of Field Robotics*, vol. 27, no. 3, pp. 344–370, 2010.

[12] B. V. Pham, A. Maligo, and S. Lacroix, "Absolute Map-Based Localization for a Planetary Rover," in *Symposium on Advanced Space Technologies for Robotics and Automation, Noordwijk*, 2013.

[13] P. Fankhauser, M. Bloesch, P. Krüsi, R. Diethelm, M. Wermelinger, T. Schneider, M. Dymczyk, M. Hutter, and R. Siegwart, "Collaborative Navigation for Flying and Walking Robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[14] N. Michael, S. Shen, K. Mohta, Y. Mulgaonkar, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida, K. Ohno, E. Takeuchi, and S. Tadokoro, "Collaborative mapping of an earthquake-damaged building via ground and aerial robots," *Journal of Field Robotics*, vol. 29, no. 5, pp. 832–841, 2012.

[15] C. Forster, M. Pizzoli, and D. Scaramuzza, "Air-ground localization and map augmentation using monocular dense reconstruction," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[16] W. M. Wells III, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis, "Multi-modal volume registration by maximization of mutual information," *Medical Image Analysis*, vol. 1, no. 1, pp. 35–51, 1996.

[17] N. Hurtós, D. Ribas, X. Cufí, Y. Petillot, and J. Salvi, "Fourier-based Registration for Robust Forward-looking Sonar Mosaicing in Low-visibility Underwater Environments," *Journal of Field Robotics*, vol. 32, no. 1, pp. 123–151, 2015.

[18] T. Kazik and A. H. Göktoğan, "Visual odometry based on the Fourier-Mellin transform for a rover using a monocular ground-facing camera," in *IEEE International Conference on Mechatronics (ICM)*, 2011.

[19] M. Faessler, F. Fontana, C. Forster, E. Mueggler, M. Pizzoli, and D. Scaramuzza, "Autonomous, vision-based flight and live dense 3D mapping with a quadrotor MAV," *J. of Field Robotics*, pp. 1556–4967, 2015. [Online]. Available: http://dx.doi.org/10.1002/rob.21581

[20] M. Hutter, C. Gehring, M. Bloesch, M. A. Hoepflinger, C. D. Remy, and R. Siegwart, "StarlETH: A compliant quadrupedal robot for fast, efficient, and versatile locomotion," 2012.

[21] M. Pizzoli, C. Forster, and D. Scaramuzza, "REMODE: Probabilistic, monocular dense reconstruction in real time," *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[22] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

[23] M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. Hoepflinger, and R. Siegwart, "State estimation for legged robots on unstable and slippery terrain," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[24] P. Fankhauser, M. Bloesch, C. Gehring, M. Hutter, and R. Siegwart, "Robot-Centric Elevation Mapping with Uncertainty Estimates," *International Conference on Climbing and Walking Robots (CLAWAR)*, 2014.

[25] C. Frueh and A. Zakhor, "Constructing 3d city models by merging ground-based and airborne views," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2003.

[26] P. Fankhauser and M. Hutter, "A Universal Grid Map Library: Implementation and Use Case for Rough Terrain Navigation," in *Robot Operating System (ROS) – The Complete Reference (Volume 1)*, A. Koubaa, Ed. Springer, 2016, ch. 5.

[27] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*, 5th ed. Wiley, 2013, pp. 188–194.

[28] A. A. Goshtasby, "Similarity and Dissimilarity Measures," in *Image Registration: Principles, Tools and Methods*, A. A. Goshtasby, Ed. London: Springer London, 2012, pp. 7–66.

[29] A. Wendel, A. Irschara, and H. Bischof, "Automatic alignment of 3d reconstructions using a Digital Surface Model," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2011.

[30] F. Pozzi, T. Di Matteo, and T. Aste, "Exponential smoothing weighted correlations," *The European Physical Journal B*, vol. 85, no. 6, pp. 1–21, 2012.

[31] C. Studholme, D. L. G. Hill, and D. J. Hawkes, "An overlap invariant entropy measure of 3d medical image alignment," *Pattern Recognition*, vol. 32, no. 1, pp. 71–86, 1999.

[32] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo localization for mobile robots," *Artificial Intelligence*, vol. 128, no. 1, pp. 99–141, 2001.