

A Monocular Vision-based System for 6D Relative Robot Localization

Conference Paper**Author(s):**

Breitenmoser, Andreas; Kneip, Laurent; Siegwart, Roland

Publication date:

2011

Permanent link:

<https://doi.org/10.3929/ethz-a-010025269>

Rights / license:

[In Copyright - Non-Commercial Use Permitted](#)

Originally published in:

<https://doi.org/10.1109/IROS.2011.6094851>

A Monocular Vision-based System for 6D Relative Robot Localization

Andreas Breitenmoser, Laurent Kneip and Roland Siegwart

Abstract—The objective of this paper is the full 6D relative localization of mobile devices, and direct robot-robot localization in particular. We present a novel relative localization system that consists of two complementary modules: a monocular vision module and a target module with four active or passive markers. The core localization algorithm running on the modules determines the marker positions in the camera image and derives the relative robot pose in 3D space. The system is supported by a prediction mechanism based on regression. The modules are tested successfully in experiments with a quadrotor helicopter as well as on a team of two e-puck robots performing a coverage task. The relative localization system provides accuracies of a few centimeters in position and up to a few degrees in orientation. Furthermore, the system is lightweight, with low complexity and system requirements, which enables its application to a wide range of mobile robot platforms.

I. INTRODUCTION

Localization is one of the fundamental components in mobile robotics. The robot has to localize itself in an often unknown environment with respect to an absolute frame of reference (*global localization*), or in relation to a local object like a landmark or other robots in its vicinity (*relative localization*). In this work, we focus on vision-based relative localization, especially in the context of multi-robot systems. Collaborative tasks often require only relative localization, i.e. absolute positioning of the robots in the world is not necessarily needed for successful task completion. Collision avoidance, pattern formation, coordinated manipulation or coverage with a group of mobile robots are examples, where mutual information of relative poses is sufficient for planning the next actions. Moreover, relative localization can lead back to global localization if a robot measures its location relative to a fixed landmark of known position.

[1] distinguished relative localization methods based on the level of sensed information: some relative localization systems only measure range or angle information, others the position or the full pose, including the orientation of the observed device. Furthermore, localization systems can be classified into planar or spatial systems, depending on whether 2D or 3D location information is measured. However, most of today’s relative localization systems rely on

This work was supported by ALSTOM. The authors like to thank Philipp Küttel for his preparatory studies on the relative localization system. We thank Prof. Raffaello D’Andrea and his group for giving us access and support for their “Flying Machine Arena” [19] to obtain ground truth for the experiments.

Andreas Breitenmoser, Laurent Kneip and Roland Siegwart are with the Autonomous Systems Laboratory (ASL), ETH Zurich, Tannenstrasse 3, 8092 Zurich, Switzerland, {andreas.breitenmoser, laurent.kneip}@mavt.ethz.ch, rsiegwart@ethz.ch

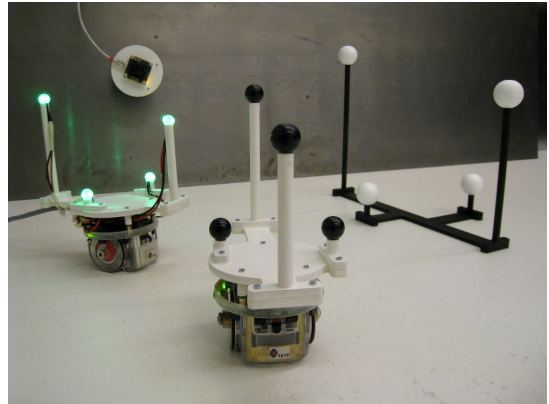


Fig. 1. Relative localization system. The localization modules are installed on three e-puck robots: the robot climbing the metallic wall in the background is equipped with a camera module, the robot on the left features a target module with the smaller target composed of active color LED markers, and the robot in the center carries the passive version of the same target module. An additional target module on the right with a larger passive target of the same geometry represents an example of a target that can be deployed in a workspace as a fixed landmark.

the assumption of a 2D workspace, and provide only 2D information of a robot’s position.

In this paper, we present a relative localization system that allows for measuring the full 6D pose of a device in 3D space. The localization system consists of a monocular camera module and a complementary spatial target module, which includes four markers in a fixed arrangement of known geometry. The position and orientation information between target and camera is determined from continuously detecting the markers in the camera image and then solving a Perspective-Three-Point (P3P) problem. Tracking of the target in the 3D space rather than in the 2D image plane improves the prediction and final detection of the modules’ poses.

Our relative localization system is in line with classic work in visual servoing [2], and is motivated by a lack of existing solutions for relative on-board localization of robots moving in 3D space. Industrial robots that climb walls and ceilings and perform coordinated inspection tasks in small teams [3], or flying robots that monitor a certain area or hover in place for landing need compact solutions for relative 6D localization. As the relative localization system is based on a simple target and a monocular camera, it only requires a single passive sensor. This results in a lightweight, low-cost and low-power system, which relies on relatively simple algorithms with low computational complexity, and offers the potential to perform relative localization in real-time.

The remainder of the paper is organized as follows. First, we review related work on relative localization systems. In Section III we present the system overview and describe the hardware components of the two localization modules. The fundamentals of the relative localization algorithm are explained in Section IV. In Section V the system is evaluated by experiments in two different scenarios: localization of a flying robot against a fixed landmark and relative localization of a ground and climbing robot in a coverage task.

II. RELATED WORK

The system in [4] uses beacons that emit radio frequency (RF) and ultrasonic pulses to measure distance information between robots located in the 2D plane; three Millibots can jointly infer the position of a fourth robot by trilateration. Similar to the Millibots, the Cricket indoor location system [5] consists of several beacons that sense 3D position from RF and ultrasonic pulses. The Wii remote, as presented in [6], and its sensor bar counterpart can be understood as localization modules that provide 4D information, i.e. 3D position plus roll. Combined with the built-in accelerometer, full pose information can be gained with the cost of an additional sensor.

Vision-based localization modules with single or multiple cameras are oftentimes more general and need less hardware development, which results in increased flexibility for integration on existing robots. The approach in [7] uses stereo vision to detect a single marker target and measures 3D position and 2D orientation information. In [8], monocular omnidirectional vision is used to build a localization system of similar functionality as in [4]. It is capable of measuring range information and 2D position within a robot team. A monocular vision approach that exploits prior information of target shape and orientation is presented in [9]; it can estimate up to 5 degrees of freedom (DOF) of a quadrotor.

QRTags, ARTags and the ARToolKit [10] represent useful tools for 6D relative localization, which is beyond their primary application in augmented reality. The ARToolKit provides comparable accuracy for tags of similar size but - unlike our system - is not designed for omnidirectional localization in 3D space.

A robotic system that achieves 6D relative localization is presented in [11]. The idea is similar to ours but more restrictive on the assumptions taken: a minimum number of four heterogeneous robots is required, from which one is climbing above the others to track them, and visibility of the ground robots among each other must be guaranteed. Especially in real applications, where often only a pair of robots or robots with similar capabilities are available, and in presence of obstacles and occlusions, the planning of trajectories to maintain visibility is neither trivial nor very practical. This clearly favors the ability of direct robot-robot localization.

Besides, 6 DOF motion capturing systems, such as the Vicon motion capture system¹ are becoming more and more

popular in the robotic community. [12] for example presents a low-cost pose tracking system and a target design methodology, from which we found inspiration. However, motion capturing systems are composed of multiple cameras which are installed in a fixed configuration in the environment in order to provide multiple views of the markers of the tracked targets. The pose is determined from the projections to the different views. In order to obtain complete 6D pose information, a minimum number of three markers per target need to be detected in at least two cameras. In contrast, our relative localization system requires only one camera but at least four markers.

III. SYSTEM OVERVIEW

Figure 1 shows the camera and target modules of the relative localization system as they are applied to a group of three e-puck robots [13].

The robot that is equipped with the camera module computes the relative localization between two robots. In general, this is the direct relative position and orientation of a neighboring robot in the robot's own frame, but also the indirect distance and orientation between two robot neighbors in the case of three (or more robots) can be inferred. The robot that carries the target module serves as a mobile landmark, which is detected by the robots that are equipped with the camera module. The 6D relative pose can be inverted and communicated from the camera to the target module. If no communication devices are available to the robots, a pair of camera and target modules may be placed on each robot to provide for mutual relative localization.

The modules are very flexible and allow for customization into many directions. Any calibrated camera can be used with the camera module for detecting the target. However, for localization in close to planar settings, omnidirectional cameras are the preferred choice due to their 360° field of view (FOV) in the direction parallel to the image plane. Fish-eye lenses are particularly suitable in 3D scenarios where devices need to be localized when moving into free open space on hemispherical trajectories around the robot.

The target on the target module is an optimized configuration of four spherical markers, all identical in size and shape. Markers are either active or passive. Active markers can be built from light-emitting diodes, such as infrared LEDs or LEDs of a specified color (see Figure 1: smaller target with markers of diameter $b = 1$ cm, made from green SMD LEDs integrated into a spherical diffusor). Passive markers are colored or reflective balls (see Figure 1: smaller target with $b = 1.5$ cm, larger target with $b = 3$ cm, both with painted markers). The camera might thus be either color or monochrome, depending on the chosen target.

The target size and geometry, the diameter of the markers and the camera resolution are important system characteristics because they define the maximum distance over which relative localization can still be performed in a reliable way. During the design of the system all three characteristics must be adjusted with respect to the final application.

¹<http://www.vicon.com>

A. Optimization of target geometries

The target has known geometry and the markers are all identical. Hence, once the markers are detected in the camera image, the identities of the markers within a target as well as the identities of several targets among each other can only be resolved by the knowledge of the markers' spatial arrangement. The target geometry is essential for the relative localization, as a good design lowers the occurrence of occlusions and similarities among different target views, and contributes to the overall robustness of the target pose prediction.

First, the self-similarity and symmetries in a single target must be minimized, such that they appear different if observed from different directions. This guarantees correct estimation of the target's orientation in space. Second, if multiple targets are present in the same system, similarity among different targets must be minimized to restore the markers' identities reliably. In addition, degenerate arrangements, such as configurations with three colinear markers or flat tetrahedrons with four nearly coplanar markers, should be avoided.

We define the similarity of target geometries according to [14] as the smallest difference of the pairwise marker distances over all markers and targets in the system. The objective of the target optimization is to find arrangements of minimum similarity to form n targets with m markers each, i.e. the maximization of $s_{min} = \min(S)$, where $S = \bigcup_{i \neq j} |d_i - d_j|$ with d_i, d_j the pairwise Euclidean marker distances between all the markers $i, j \in I = \{1, \dots, (n \cdot m)\}$. The optimization procedure follows the procedure explained in [14] to a large part.

At the beginning, the targets are initialized by random sampling from a predefined set of discrete positions and lengths. The initial set of markers is then optimized iteratively by maximizing the cost function

$$f(P, w) = \begin{cases} w \cdot s_{min} + (1 - w) \cdot s_{avg}, & D_1 \leq d_i \leq D_2, \\ & \wedge h_{min} \geq H \\ -G, & \text{otherwise} \end{cases} \quad (1)$$

with s_{avg} the average distance difference over S , w a free weighting parameter to balance worst-case and average target quality, D_1 and D_2 the lower and upper bounds of the allowed pairwise marker distances and H the minimum allowed height of the target faces. H introduces a colinearity constraint, which prevents the target to turn out too flat. G is a constant positive value that penalizes violations of the constraints imposed by D_1, D_2 and H . The dimensions of the larger target used in this paper (see Figure 1 right) are given by the tetrahedron with vertex set $\{(0.00, -16.51, 23.66), (0.00, 13.04, 18.45), (6.26, -2.21, 5.39), (-8.02, 2.49, 6.21)\}$, and circumsphere of radius 15.2 (all in [cm]). It was obtained from an optimization with parameters set to $w = 2/3$, $D_1 = 15$ cm, $D_2 = 30$ cm and $H = 4 \cdot b = 12$ cm. The smaller passive and active targets installed on the e-puck robots have the same proportions but are scaled to half the size (see Figure 1 on the left and center).

The actual optimization was carried out with the Nelder-Mead simplex algorithm [15]. The optimization was repeated over 1000 runs and resulted in several best solutions due to the existence of local optima. Especially, as we optimized for only a single target ($n = 1$), many valuable solutions remain. In order to select a final solution, the projection of the markers to the support plane tangent to the robot base was considered in a last optimization step. The target is rotated, such that the cost function in Equation (1) with adjusted constraints D_1, D_2 and H is maximized. This leads to targets that are most distinct when viewed from the top, which is particularly favorable for robots climbing on opposite surfaces, or flying robots that localize with respect to the ground (like in our experiments of Section V for example).

IV. RELATIVE LOCALIZATION ALGORITHM

Next we describe step-by-step the processing pipeline of the relative localization algorithm that runs on top of the localization system hardware introduced in the previous section. We consider the case of a single camera with camera frame C and one visible target with target frame $T_l = T$, where $l \in I_{tar} = \{1, \dots, n\}$ and $n = 1$. Figure 2 illustrates the sequence of operations of the proposed method.

A. Pose and marker prediction

The relative localization system uses single cameras to determine poses in 3D space. Poses can be predicted by tracking the markers in the image plane or the relative camera location in 3D space. We track the pose of the camera relative to the target frame T . Pose prediction in 3D improves robustness of the algorithm as spatial information is retained. The 3D pose estimation resolves situations with crossing marker trajectories in the image plane or markers leaving the image plane for a certain fraction of time, and allows for direct inclusion of the underlying relative motion model of the moving camera. It also allows for the application of standard pose estimation methods, such as Kalman filters.

In our current implementation we make use of a simplified approach for predicting the image coordinates of the target's markers. The 6D camera pose $[\hat{R}_{TC}^k | \hat{\mathbf{t}}_{cam,T}^k]$ at discrete time k is estimated based on the camera pose history $\{[\mathbf{R}_{TC}^{k-k_R} | \mathbf{t}_{cam,T}^{k-k_R}], \dots, [\mathbf{R}_{TC}^{k-1} | \mathbf{t}_{cam,T}^{k-1}]\}$ relative to the target frame T , with k_R the number of regression samples taken into account. Linear regression with $k_R = 2$ already results in an accurate prediction with only slight overshooting at abrupt motion changes, as can be expected from the linear motion model. Through the modular character of our algorithm, more advanced pose estimators that take into account measurements from additional sensors - as for instance inertial readings - can easily replace the current pose prediction.

After the prediction of the 6D pose of the camera frame C relative to the target frame T is obtained, the predicted marker positions $\hat{Q}^k = \{\hat{\mathbf{q}}_i^k\}, \forall i \in I$, in the camera image plane follow from the projection of the target's marker positions $P_{tar,T} = \{\mathbf{p}_{tar_i,T}\}, \forall i \in I$. In the case of a pinhole camera model, the predicted marker positions are given by $\hat{\mathbf{q}}_i^k =$

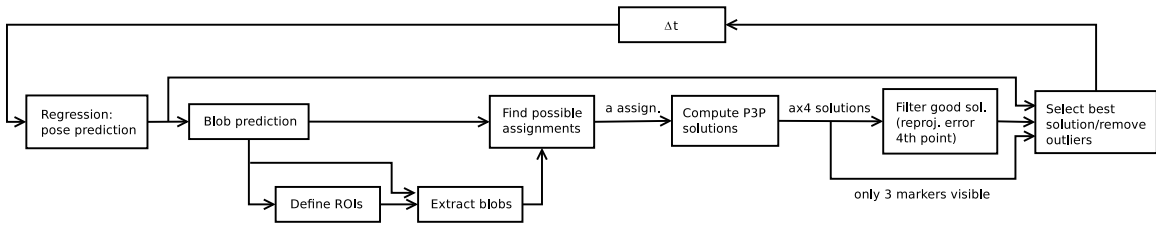


Fig. 2. Processing pipeline of the relative localization algorithm running on the camera modules.

$K \cdot \left[\widehat{R}_{TC}^k \mid -\widehat{R}_{TC}^k \cdot \widehat{\mathbf{t}}_{cam,T}^k \right] \cdot \mathbf{p}_{tar_i,T}$, with K the intrinsic camera parameter matrix.

B. Blob extraction

The predicted marker positions \widehat{Q}^k allow for extracting a region of interest (ROI) in the image up-front. Thus any image processing operations can be constrained to the ROI, which leads to substantial increase in speed and robustness. During the first k_R runs of the initialization phase, where the prediction is still inaccurate or unavailable, an adapted version of the blob extraction is applied to the whole image. Besides, in the case of detection failure blobs may be searched in the entire image for reinitialization.

The actual blob extraction mainly applies standard image processing methods, which makes it simple to reimplement on other platforms. The blob extraction for grayscale images works similarly for active and passive markers, only that the thresholds are computed inversely depending on whether light or dark markers are used. First, the image foreground is extracted by iterated thresholding and filtering (see Figure 3 right). A strategy based on successive application of two adaptive thresholds has shown to be especially useful for initial blob detection over the entire image, or for scenes with changing light. The first threshold F_1 is computed after Otsu's criterion and assumes that the image contains clear fore- and background. Whereas threshold F_1 locates the blobs in the image, threshold F_2 refines their shape and location by being less restrictive. F_2 exploits the fact that there are much more background pixels than foreground pixels. A Gaussian is fitted to the grayscale histogram of the image and F_2 is defined at a multiple of the standard deviation. After thresholding with F_1 , the extracted set of blobs is filtered by executing the first round of outlier rejection: too small and too large blobs as well as blobs with strong non-convexity and excentricity are removed. The window that encloses all the remaining blobs is subject to threshold F_2 , and a second round of outlier rejection is started.

If the number of blobs still exceeds the specified number of markers ($n \cdot m$) in the system, the nearest neighbor of each blob is computed to subsequently remove the blobs with farthest distance to all the remaining blobs, or to merge the two closest blobs respectively. An alternative would be to make use of the prediction of the blob position in the image to reject false positives. On the other hand, if the number of detected markers is below $n \cdot m$, a cascade of recovery methods with increasing complexity - from simple erosion to circular Hough transform (applied to the unwrapped image

for lenses of high distortion) - is executed. Finally, the blob centers $Q^k = \{\mathbf{q}_i^k\}, \forall i \in I$, result from the centroid calculation of the blob areas with subpixel precision (see Figure 3 left).

We noticed that passive markers sometimes are affected by shadows on the bottom side, which has the unfavorable effect of cutting a segment of the circular blob away when thresholding. In these cases, we suggest to fit an ellipse to the (unwrapped) blob and determine the blobs' centers by intersection of the perpendicular bisectors of two chords. A circular Hough transform could be used alternatively.

The use of color cameras in combination with colored markers rather simplifies the blob extraction step. The foreground extraction as described above for grayscale images is then replaced by thresholding the image in the YUV color space, as proposed in [16]. This adds robustness to the blob detection because the relevant color information on the U and V dimensions is decoupled from marker brightness.

Remark 1: In naturally dark environments (e.g. at night, in tunnels or pipelines), active targets are favorable. In man-made environments of changing light conditions (e.g. indoor hallways or industrial structures), passive color targets are most useful. A clear limitation of the approach is shown in cluttered indoor or outdoor environments of changing light and color. Investigations of infrared or laser-emitting targets as solutions for such environments are left for future research.

C. Pose update

After having found the marker positions Q^k in the image plane, one still needs to determine their correspondences to the 3D target points. In the worst case (e.g. during initialization when no prediction exists) a maximum of $(n \cdot m)! = 4! = 24$ permutations S_σ , with $\sigma : X_{nm} \rightarrow X_{nm}$, results for a single target with four markers ($n = 1, m = 4$). After initialization, this number can be strongly decreased by the pose prediction and the resulting distances of the predicted (correspondence known) and extracted (correspondence unknown) marker positions \widehat{Q}^k and Q^k in the image plane. Thus S_σ is restricted to a subset of permutations with high likelihood. However, there might be ambiguous situations remaining, for instance when two of the markers are close to each other.

Once all the possible permutations are determined, the final camera pose with respect to the target frame $\left[R_{TC}^k \mid \mathbf{t}_{cam,T}^k \right]$ is to be computed. The identical problem of finding the pose of a camera given three points in the world frame (and their corresponding points in the image plane)

is well-known in computer vision as the Perspective-Three-Point (P3P) problem.

There are several approaches to solve the P3P problem, such as the algorithms presented in [17] and by references therein. We make use of a novel closed-form solution [18] that derives translation $\mathbf{t}_{cam,T}^k$ and rotation R_{TC}^k of the camera with respect to the target frame T directly, i.e. without further need of intermediate derivation of first the target point coordinates in the camera frame C , and then the aligning transformation of two point groups. Increased numerical stability and especially computational efficiency is gained, which favors any lightweight implementation.

As the geometry of the target and the positions of the four markers $P_{tar,T}$ are known, projecting three of the four marker points into the parametrized camera pose $\left[R_{TC}^k(\theta) \mid \mathbf{t}_{(P_{tar_1}^{cam}),T}^k(\theta) \right]$ results in the quartic equation

$$a_4 \cdot \cos^4(\theta) + a_3 \cdot \cos^3(\theta) + a_2 \cdot \cos^2(\theta) + a_1 \cdot \cos(\theta) + a_0 = 0, \quad (2)$$

with the coefficients a_4, \dots, a_0 as defined in [18]. Solving for parameter $\cos(\theta)$ using Ferrari's approach and substituting back leads to four solutions for the camera position $\mathbf{t}_{cam,T}^k = \mathbf{p}_{tar_1}^k + \mathbf{t}_{(P_{tar_1}^{cam}),T}^k(\theta)$ and orientation $R_{TC}^k = R_{TC}^k(\theta)$ in T . The target pose with respect to the camera frame C is inversely given by $R_{CT}^k = R_{TC}^{kT}$ and $\mathbf{t}_{tar,C}^k = -R_{CT}^k \cdot \mathbf{t}_{cam,T}^k$.

In our case, the correspondences between the 2D points in the image plane and the 3D world reference points are not predefined. Each of the permutations in S_σ serves as starting point to solve the above P3P problem with $4! \cdot 4 = 96$ solutions in the worst case. The reprojection of the fourth marker position \mathbf{q}_4^k is then used for disambiguation; the candidate transformations for which the fourth point does not fit are removed from the solution set. In the initialization phase, multiple valid hypotheses may be maintained until there is only one remaining. If not in the initialization mode, the unique solution $\left[R_{TC}^k \mid \mathbf{t}_{cam,T}^k \right]$ is finally obtained by selecting the remaining candidate that is closest to the prediction $\left[\hat{R}_{TC}^k \mid \hat{\mathbf{t}}_{cam,T}^k \right]$. Alternatives for outlier rejection filtering are the inclusion of prior knowledge, such as the knowledge that a ground robot is always moving in the 2D plane, the use of robot odometry information, or the inclusion of further predictions for the same target by other camera modules.

Remark 2: Opposed to our assumption, markers could also be distinct. By means of different colors, color codes or varying emission frequencies of pulsed active markers, the identities are assigned to the markers statically, independent of any 3D information. However, staying with the more general problem formulation of having identical markers per target leaves the possibility to use colors or emission patterns for the differentiation among multiple targets. In summary, target modules with distinct markers improve the robustness of the relative localization by simplifying the marker identification process, increasing the stability when searching for 2D/3D point correspondences and reducing the importance of optimality in the target design.

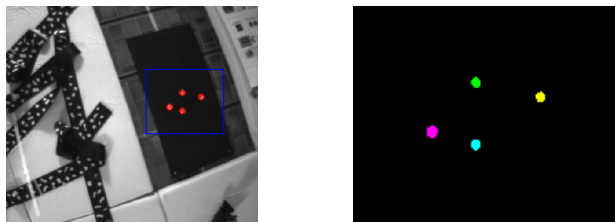


Fig. 3. Blob extraction from Experiment 1. Left: four markers are detected in the ROI. Right: labeled blobs resulting from the foreground extraction.

V. EXPERIMENTAL EVALUATION

The relative localization system is tested in two different contexts: in Experiment 1, a quadrotor helicopter with a camera module is localized against a target module placed as landmark on the ground, whereas two e-puck robots jointly cover a ground and wall surface in Experiment 2.

A. Experiment 1: Aerial vehicle localization

The goal of the first experiment is to characterize the relative localization system and analyze its accuracy. A Point Grey Firefly MV monochrome camera² with 752×480 resolution and 90° FOV lens is mounted on the bottom of a AscTec Hummingbird quadrotor³ pointing toward the ground, where the larger target is positioned as a fixed landmark (see Figure 3 left). The experiment is carried out in a laboratory room with a Vicon motion capture system installed, which provides the ground truth of the robot trajectory.

In a first run, the quadrotor is moved over the target by hand to produce rather rich and well-controlled trajectories in an environment close to the one of the final robot deployment. The ground truth and the measured pose obtained from our relative localization system, as well as the corresponding localization errors, are presented in Figure 4. The system achieves following accuracy: the position error is between 0.1 cm and 12.2 cm with a mean of 1.5 cm and standard deviation of 0.7 cm; the orientation error lies between 0.1° and 4.5° with a mean of 1.2° and standard deviation of 0.4° . The distance from the camera on the quadrotor to the target module varied between 67.7 cm and 174.1 cm.

In a second run, the quadrotor was flown over the target up to a height of 3 m. At this distance, the localization accuracy is 2.2 cm and 1.0° in average. The 3D trajectory of the flying quadrotor is visualized in Figure 5. A video with sequences of the experimental runs accompanies the paper.

In both experiments, we observed only a minor increase in the localization error when the distance between the camera and target module was increased to the maximum range. However, the relation of the distance between modules, the target and marker sizes as well as the camera resolution define limitations of the system. At a distance of 3 m, the size of the detected markers is reduced to only a few pixels in the camera of 752×480 resolution for example. In the

²<http://www.ptgrey.com>

³<http://www.asctec.de>

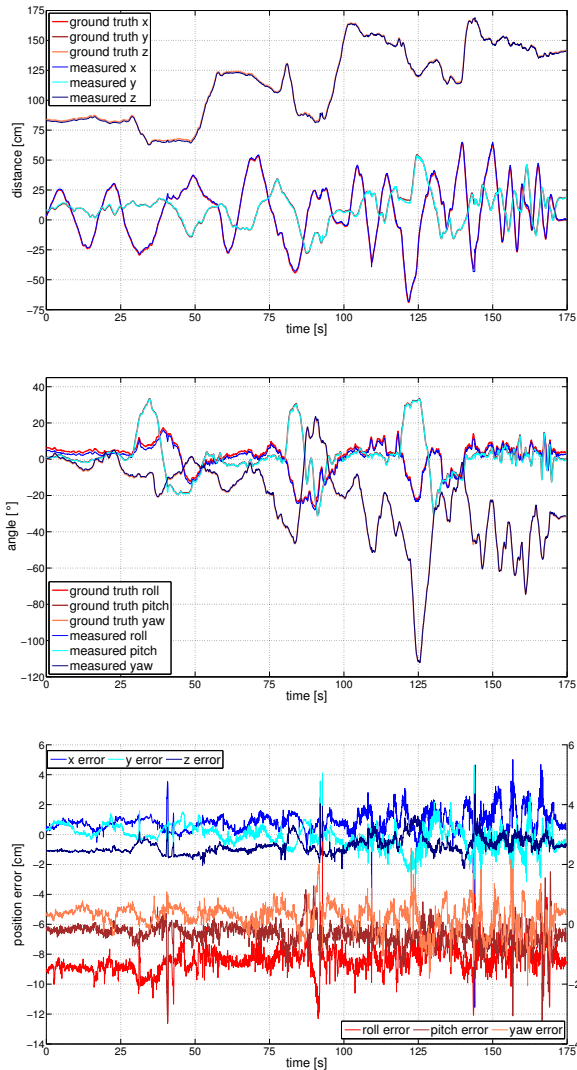


Fig. 4. Relative localization of a quadrotor, which is moved manually over a target module landmark. 6D pose (position and angles) and corresponding errors are shown.

experiments, several situations occurred where only two or three of the four markers were visible in the camera for a certain duration. The localization algorithm proved to be robust and recovered from these situations, even though the accuracy is affected. The maximum errors of 12.2 cm and 4.5° from the first run for instance are both caused by situations with only three markers visible in the camera image.

B. Experiment 2: Relative localization for multiple robots

The second experiment aims at the application of the relative localization modules to a multi-robot system. The experimental setup comprises an active and passive target module, and two camera modules. The first camera module consists of a Point Grey Flea2 color camera with 1280 × 960 resolution installed overhead at a height of 230 cm. It remains static throughout the duration of the experiment and can be thought of representing a robot, which is aware of its

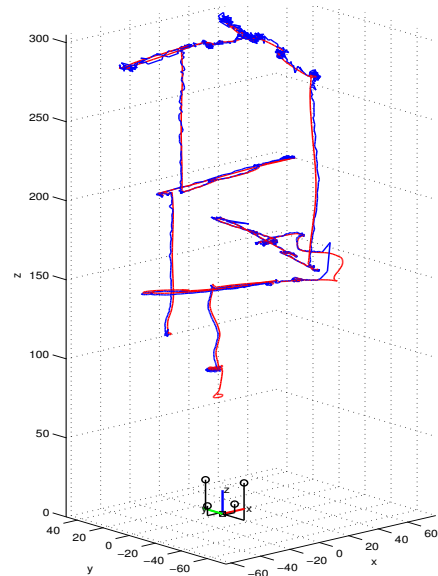


Fig. 5. Relative localization of a flying quadrotor. The ground truth (red) and measured (blue) trajectories are visualized (position in [cm]).

absolute position at the ceiling (e.g. it might have means for climbing and performing global localization). The passive or active target module, respectively, is fitted to robot A, an e-puck robot that covers the surface on the ground with a swath coverage pattern of dimensions 30 cm × 42 cm. The second camera module uses an IDS uEye color camera⁴ with 752 × 480 resolution and 150° FOV lens, and is mounted on robot B, an e-puck robot further augmented with magnets in the structure, such that it can climb the wall to cover the inclined surface (swath coverage pattern with dimensions 25 cm × 21 cm).

Figure 6 shows the two e-puck robots during the coverage task and the views of the camera modules, as well as the view from an additional external camera recording the experiment. The ground truth coverage paths and the actual coverage paths are projected into the image planes. The measured paths deviate from the ground truth by a few centimeters. The localization accuracy of the target module however varies significantly between robot A and robot B. The average and maximum position errors are 1.0 cm and 3.3 cm for robot A and 1.7 cm and 7.8 cm for robot B. Whereas the overhead camera features a higher resolution and remains completely static during the experiment, robot B carries a camera with lower resolution and moves while detecting the moving target module on robot A. Furthermore, the trajectory of robot B is transformed via robot A into the frames of the overhead camera and the external camera, which results in an addition of localization errors.

The same setup was tested for robot A equipped with the active target in a dimmed room with similar results (see Figure 7). Two runs of the experiment, one with passive and one with active target module, are included in the accompanying video.

⁴<http://www.ids-imaging.de>

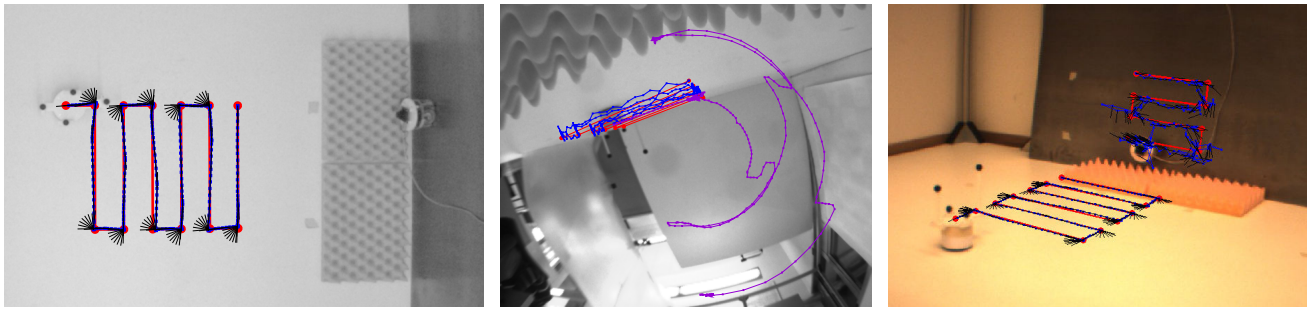


Fig. 6. Relative localization of multiple robots (ground truth paths in red, measured paths in blue, robot orientations in black). Left: View of the overhead camera. Center: View of robot B (the trajectory in violet is obtained from the projection when varying over time). Right: View of the external camera.

VI. CONCLUSIONS

The paper addresses the problem of 6D relative localization for robots like flying or climbing robots, which move in 3D space. We propose a novel relative localization system, composed of a target and a camera module, which offers a simple, lightweight and portable solution that relies on monocular vision. The modules were applied in two different scenarios, namely a hovering quadrotor helicopter and two e-puck robots collaboratively covering surfaces of different inclination. The experiments proved the applicability of the relative localization system to real robots and the evaluation of the measured poses resulted in centimeter/degree accuracies, which is sufficient for most robot applications.

The system will be integrated and tested on the MagneBike inspection robots [3], which are able to climb tube-like industrial structures for non-destructive testing. As the long-term goal is coordinated inspection among several MagneBikes, the optimization and disambiguation of multiple targets in the same system, as well as the combination of our relative localization system with cooperative localization approaches (e.g. [1]) offer further interesting directions for future research.

REFERENCES

- [1] I.M. Rekleitis, G. Dudek and E.E. Miliotis, "Multi-Robot Cooperative Localization: A Study of Trade-offs Between Efficiency and Accuracy", in *Proc. of The IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2690–2695, 2002.
- [2] W.J. Wilson, C.C. Williams Hulls and G.S. Bell, "Relative End-Effector Control Using Cartesian Position Based Visual Servoing", in *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 684–696, 1996.
- [3] A. Breitenmoser, F. Tâche, G. Caprari, R. Siegwart and R. Moser, "MagneBike - Toward Multi Climbing Robots for Power Plant Inspection", in *Proc. of The 9th Int. Conf. on Autonomous Agents and Multiagent Systems*, 2010.
- [4] L.E. Navarro-Serment, C.J.J. Paredis and P.K. Khosla, "A Beacon System for the Localization of Distributed Robotic Teams", in *Proc. of The Int. Conf. on Field and Service Robotics*, pp. 232–237, 1999.
- [5] A. Smith, H. Balakrishnan, M. Goraczko and N.B. Priyantha, "Tracking Moving Devices with the Cricket Location System", in *Proc. of The 2nd Int. Conf. on Mobile Systems, Applications and Services*, 2004.
- [6] S. Olufs and M. Vincze, "A Simple Inexpensive Interface for Robots using the Nintendo Wii Controller", in *Proc. of The IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 473–479, 2009.
- [7] A.J. Davison and N. Kita, "Active Visual Localization for Multiple Inspection Robots", in *Advanced Robotics*, vol. 16, no. 3, pp. 281–295, 2002.

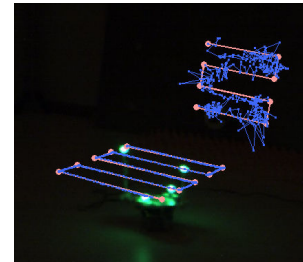


Fig. 7. Robot A equipped with the active target module. The ground truth (red) and measured (blue) coverage patterns are projected into the image.

- [8] J. Spletzer, A.K. Das, R. Fierro, C.J. Taylor, V. Kumar and J.P. Ostrowski, "Cooperative Localization and Control for Multi-Robot Manipulation", in *Proc. of The IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 631–636, 2001.
- [9] D. Eberli, D. Scaramuzza, S. Weiss and R. Siegwart, "Vision Based Position Control for MAVs Using One Single Circular Landmark", in *Journal of Intelligent and Robotic Systems*, vol. 61, no. 1-4, pp. 495–512, 2011.
- [10] H. Kato and M. Billinghurst, "Marker Tracking and HMD Calibration for a video-based Augmented Reality Conferencing System", in *Proc. of The 2nd Int. Workshop on Augmented Reality*, 1999.
- [11] Y. Feng, Z. Zhu and J. Xiao, "Self-Localization of a Heterogeneous Multi-Robot Team in Constrained 3D Space", in *Proc. of The Int. Conf. on Intelligent Robots and Systems*, pp. 1343–1350, 2007.
- [12] T. Pintaric and H. Kaufmann, "Affordable Infrared-Optical Pose-Tracking for Virtual and Augmented Reality", in *Proc. of IEEE VR Workshop on Trends and Issues in Tracking for Virtual Environments*, pp. 44–51, 2007.
- [13] F. Mondada et al., "The e-puck, a Robot Designed for Education in Engineering", in *Proc. of The 9th Conf. on Autonomous Robot Systems and Competitions*, pp. 59–65, 2009.
- [14] T. Pintaric and H. Kaufmann, "A Rigid-Body Target Design Methodology for Optical Pose-Tracking Systems", in *Proc. of The ACM Symp. on Virtual Reality Software and Technology*, pp. 73–76, 2008.
- [15] J.C. Lagarias, J.A. Reeds, M.H. Wright and P.E. Wright, "Convergence Properties of the Nelder–Mead Simplex Method in Low Dimensions", in *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 112–147, 1998.
- [16] J. Bruce, T. Balch and M. Veloso, "Fast and Inexpensive Color Image Segmentation for Interactive Robots", in *Proc. of The IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2061–2066, 2000.
- [17] X.-S. Gao, X.-R. Hou, J. Tang and H.-F. Cheng, "Complete Solution Classification for the Perspective-Three-Point Problem", in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 930–943, 2003.
- [18] L. Kneip, D. Scaramuzza and R. Siegwart, "A Novel Parametrization of the Perspective-Three-Point Problem for a Direct Computation of Absolute Camera Position and Orientation", in *Proc. of The IEEE Conf. on Computer Vision and Pattern Recognition*, 2011.
- [19] S. Lupashin, A. Schöllig, M. Sherback and R. D'Andrea, "A Simple Learning Strategy for High-Speed Quadcopter Multi-Flips", in *Proc. of The IEEE Int. Conf. on Robotics and Automation*, 2010.