2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Active Monocular Localization: Towards Autonomous Monocular Exploration for Multirotor MAVs

Christian Mostegel, Andreas Wendel and Horst Bischof

***Abstract*— The main contribution of this paper is to bridge the gap between passive monocular SLAM and autonomous robotic systems. While passive monocular SLAM strives to reconstruct the scene and determine the current camera pose for any given camera motion, not every camera motion is equally suited for these tasks. In this work we propose methods to evaluate the quality of camera motions with respect to the generation of new useful map points and localization maintenance. In our experiments, we demonstrate the effectiveness of our measures using a low-cost quadrocopter. The proposed system only requires a single passive camera as exteroceptive sensor. Due to its explorative nature, the system achieves autonomous way-point navigation in challenging, unknown, GPS-denied environments.**

I. Introduction

Simultaneous Localization and Mapping (SLAM) plays an important role for autonomous robotic systems as it grants a notion of location awareness in unknown environments. The quality of the localization is strongly bound to the perception range and accuracy of the system's sensors. Micro Aerial Vehicles (MAVs), which can acquire images from unconventional viewpoints at low cost, have very limited power resources at their disposal. This means that each additional power consuming device on board the MAV directly leads to a reduction of the maximum mission time. Thus, a range of different sensors have been proposed for this task.

Aside from GPS, which is unreliable in urban areas, many approaches use other active sensors, such as laser scanners [1], [2] or RGBD-cameras [3], [4], [5], to obtain an accurate localization. Irrespective of the increased payload, the main drawback of these active sensing technologies is that their maximum depth perception range is limited to a few meters. In contrast, passive cameras have a significantly lower power consumption and a high perception range, which is only limited by atmospheric influences such as the humidity of the air. Thus, some approaches [6], [7] use passive stereo-cameras on customized MAV setups for localization. In this work, however, we aim to achieve autonomous navigation and localization with a minimum of sensors; i.e. a monocular camera.

Current monocular SLAM approaches [8], [9], [10], [11] strive to passively determine the current camera pose and map the environment. In practice as in theory, not all view points and camera motions are equally suited for monocular SLAM. Due to the missing saliency, regions with a lack of texture or contrast are ill-suited for localization. When suitable correspondences between salient image structures have been established, the 3D uncertainty of these structures mainly depends on the triangulation angle between the observations and the 3D structures [12]. Thus, it is crucial that salient structures are observed from different angles. This means that certain maneuvers, such as a pure rotation, are inapt for the task of monocular SLAM and can easily lead to a loss of the visual localization.

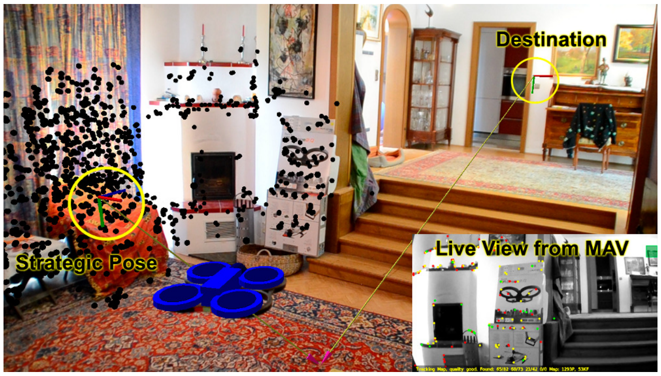

Destination
Strategic Pose
Live View from MAV
Tracking Map, quality good. Found: 65/82 68/73 21/42 0/0 Map: 1293P, 53KF

Fig. 1. Active Visual Localization. Our system actively maintains the visual localization at all times during a flight to the destination. This is achieved by analyzing the SLAM map (black dots), and by strategic positioning of the vehicle for enhanced map triangulation. The blue object visualizes the estimated MAV pose.

In this work, we overcome these problems by integrating the perceptual needs of the system into our navigation approach. Thus, we propose a set of novel measures that provides the necessary means to enable autonomous monocular navigation in unknown GPS-denied environments. The measures allow the system to generate new map points in an intelligent fashion and to plan the camera trajectories such that the visual localization is maintained at all times.

II. Related Work

The task of ensuring the integrity of the visual localization can be seen as a Next-Best-View (NBV) problem. Most current literature on NBV planning is concerned with the maximization of the reconstruction completeness and minimization of the reconstruction uncertainty. In contrast, our work is mainly concerned with the stability of the visual localization and the generation of new map points for localization purposes. Although the main aims differ, good reconstruction vs. good localization, some ideas of the reconstruction-focused literature can be adapted for our purposes.

C. Mostegel and H. Bischof are with the Aerial Vision Group at Graz University of Technology, Austria. A. Wendel is an alumnus of this group and now affiliated with Google Inc. (email: *lastname*@icg.tugraz.at). The research leading to these results has received funding from the EC FP7 project 3D-PITOTI (ICT-2011-600545) and from the Austrian Research Promotion Agency (FFG) FIT-IT project Pegasus (825841).

Most of the criteria for uncertainty reduction in current literature are based on covariance matrices. Wenhardt et al. [13] analyzed the three most prominent criteria, which are based on the determinant (*D-optimally*), the largest eigenvalue (*E-optimally*) and the trace (*T-optimally*) of the covariance matrix. They conclude that there is no significant advantage of one criterion over the others.

Pioneer work in the field of active vision was done by Davison and Murray [14]. They use a pair of two movable cameras on a non-holonomic ground robot and try to observe map points as perpendicular as possible to the axis of the highest uncertainty.

Haner and Heyden [15] model the camera motion in a discrete fashion and concurrently optimize the reconstruction uncertainty (*T-optimally criterion*) as well as a weighted function of the camera path in a Levenberg-Marquardt optimization scheme. Due to the complexity of the optimization formulation, this approach is inapt for real-time applications.

Other approaches, such as [16], [17], use meshing techniques to additionally incorporate visibility constraints. Hoppe et al. [17] use the 3D mesh to optimize the *E-optimally criterion* while considering the necessary camera overlap through counting the number of mesh triangles which are visible in neighboring cameras. Additionally to the reconstruction uncertainty, Dunn and Frahm [16] also incorporate the matching probability into their optimization scheme. They analyze the area which a surface patch on a 3D mesh occupies in the image projection and measure the surface texture saliency.

As the localization accuracy depends on the reconstruction uncertainty, we incorporate a term which penalizes a high point uncertainty in our optimization criterion. Inspired by the work of Dunn and Frahm [16], we also incorporate a viewpoint-dependent feature recognition probability that does not require the computation of a 3D mesh.

As Strasdat et al. [18] have shown that bundle-adjustment-based SLAM outperforms filter-based SLAM in matters of accuracy, we base our measures solely on data which is available in every bundle-adjustment-based approach; i.e. a set of keyframes with 2D feature points, the 3D map points, and the related observation information. The required data is shortly outlined in Fig. 2 with a reconstruction example. A detailed introduction to bundle-adjustment-based SLAM can be found in [8].

III. Localization Quality

In order to prevent a localization loss, we propose a novel measure to estimate the influence of possible camera motions on the stability of the visual localization. A fast evaluation of the "localization quality" is ensured by basing it solely on the geometric constellation of a sparse reconstruction (point cloud or map) and the related observations.

A. Geometric Point Quality

As monocular SLAM relies on a map for localization, we integrate the reliability of a map point into our localization quality measure. As observed by Beder and Steffen [12], the uncertainty of a map point mainly depends on its triangulation angle. Thus, we avoid the computationally expensive calculation and evaluation of covariance matrices and base our measure solely on the triangulation information of a map point. The idea of our formulation is to penalize 3D points with small triangulation angles as they have an increased location uncertainty.

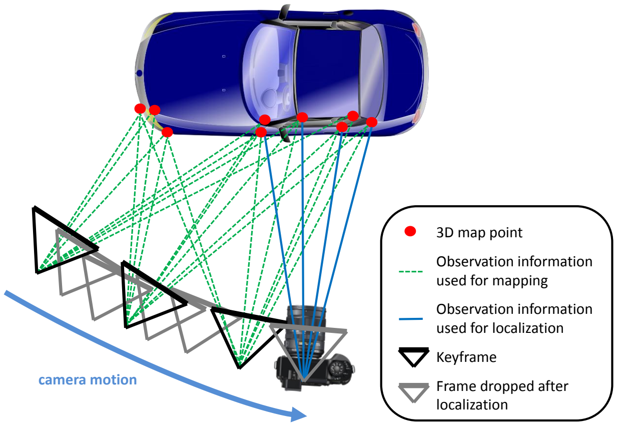

3D map point
Observation information used for mapping
Observation information used for localization
Keyframe
Frame dropped after localization
camera motion

Fig. 2. In this example a parking car is reconstructed by a moving camera. During the camera motion a video stream is transmitted to the SLAM module. All video frames are used to track the already constructed map points and thereby estimate the current camera pose. In contrast, only a subset of frames, i.e. the set of keyframes, is used for the map construction. By observing salient structures in multiple keyframes, 3D map points can be triangulated. Aside from 2D feature points, our approach only requires the keyframes (black triangles), the 3D map points (red dots), and the observation information between them (dotted green lines) as input.

Thus, we define the triangulation angle dependent point quality q_α as

$$q_\alpha = \begin{cases} |\alpha_{max}|/\alpha_{cap} & \text{if } |\alpha_{max}| \leq \alpha_{cap} \\ 1 & \text{otherwise} \end{cases}, \qquad (1)$$

where α_{max} is the maximum angle between a point f and its related keyframes, which is scaled and capped by the angle α_{cap}. We limit the angle to α_{cap} because it was observed that the largest triangulation angle is often found on outliers that appear very close to the camera. The value of α_{cap} is inferred at start-up and set to the upper quartile of all point's angles.

An image measurement of a map point can be seen as a proof of its existence. It was observed that a high percentage of bad outliers, i.e. map points that are far away from any object, has only very few image correspondences. Hence, the proposed quality q_{out} penalizes map points that have insufficient proof of their existence as

$$q_{out} = \begin{cases} 1 & \text{if } |K_f| \geq N_g \\ |K_f|/N_g & \text{otherwise} \end{cases}, \qquad (2)$$

where $|K_f|$ is the total size of the keyframe set of point f. N_g represents a "good" minimum size for the set for which the outlier probability is sufficiently low ($N_g = 4$ in our experiments).

By defining the geometric point quality as $q_f = q_\alpha \cdot q_{out}$, a point can only get a high score if its maximum triangulation

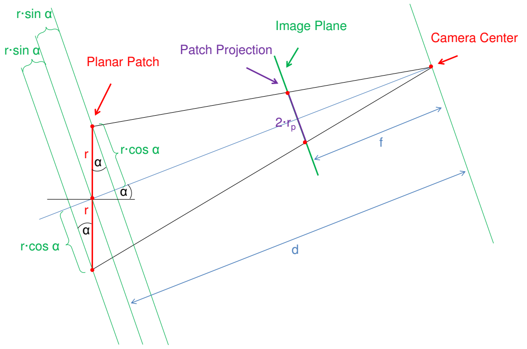


Fig. 3. Area of projection of a planar patch in relation to the viewing angle. All green lines are orthogonal to the principle axis of the camera (blue line).

angle is sufficiently large and it has a sufficient number of image measurements to attest its existence.

B. *Point Recognition Probability*

For monocular localization it is crucial that already generated 3D map points can be repeatedly recognized over a series of images. Consequently, it is important to know if a certain map point will be recognizable from a certain viewpoint. In our model we treat viewing angle and scale separately.

Viewing Angle Dependency. Using affine transformation, the projection of a circular planar patch can be approximated with an ellipse. Thus, the area of projection A_p is defined by the length of its major axis r_q and its minor axis r_p as $A_p = \pi \cdot r_q \cdot r_p$. If we consider a pure change of the viewing angle, the length of major axis r_q stays constant no matter in which direction the viewing angle changes. Using simple 2D geometry, we infer the length of the minor axis $r_p(\alpha)$ as

$$r_p(\alpha) = \frac{f \cdot r \cdot cos\alpha}{2}\left(\frac{1}{d - r \cdot sin\alpha} + \frac{1}{d + r \cdot sin\alpha}\right), \quad (3)$$

where f is the focal length, d the distance from the camera center to the center of the circular patch and r the radius of the patch as illustrated in Fig. 3. Using the assumption that $r \ll d$ the function can be simplified to

$$r_p(\alpha) \approx \frac{f \cdot r \cdot cos\alpha}{d}. \quad (4)$$

If we express the recognition probability p_α in terms of $A_p(\alpha)$ and normalize the term with a maximum area of $A_p(0)$, all constants drop out of the equation and p_α is reduced to

$$p_\alpha = \begin{cases} cos\alpha & \text{if } \alpha \in [-\pi/2, \pi/2] \\ 0 & \text{otherwise} \end{cases}. \quad (5)$$

Scale Dependency. Respecting scale is more complex than the angle, as most systems are based on an image pyramid. Thus, one has to respect at which scale the original feature was detected to know at which scale the recognition probability starts to decay. In our model we use a piecewise linear function to approximate the recognition probability, where the parameters vary for each scale level.

Let the relative scale change of a feature point f between a keyframe k and an arbitrary query frame i be defined as $s(k,i) = d(f,i)/d(f,k)$, where $d(f,i)$ and $d(f,k)$ are distances from the feature point f to the camera centers of the frames i and k respectively. Thus, we model the scale dependent recognition probability as

$$p_s = \begin{cases} \frac{s(k,i) - S_{0,l}}{S_{1,l} - S_{0,l}} & \text{if } s(k,i) \in [S_{0,l}, S_{1,l}] \\ 1 & \text{if } s(k,i) \in \;]S_{1,l}, S_{2,l}[\\ 1 - \frac{s(k,i) - S_{2,l}}{S_{3,l} - S_{2,l}} & \text{if } s(k,i) \in [S_{2,l}, S_{3,l}] \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

where $S_{*,l}$ are experimentally determined constants of the extraction scale level l of the concerned map point f in the original keyframe k. Finally, we define the overall recognition probability as the product $p_f = p_\alpha \cdot p_s$.

C. *Localization Quality Measure*

The main purpose of the localization quality measure is the prediction of an impending loss of the visual localization. Thus, the proposed measure does not only respect feature recognition probability, but also the eventuality of unpredictable occlusions and navigational imperfection.

First we split the image into 8×8 bins to simplify the data. Empirically, we found that this size is the smallest power of two which still retains sufficient location information. Then we project the already mapped points back into the virtual image of the query frame and sum the information of all map points inside a bin b to a single score $s_0(b)$

$$s_0(b) = \sum_{f \in b} p_f \cdot q_f, \quad (7)$$

where p_f is the point recognition probability of a point f and q_f is the point quality. As not to over-reward huge amounts of points in a bin, we limit the maximum quality of a bin to $s_{0,max}$. When we start our autonomous navigation, we analyze the quality of the current view and set $s_{0,max}$ to the maximum score of all bins of this view. Hence, no manual parameter tuning is required. Then we define the normalized quality q_b of a bin b as

$$q_b = \begin{cases} \frac{s_0(b)}{s_{0,max}} & \text{if } s_0(b) < s_{0,max} \\ 1 & \text{otherwise} \end{cases}. \quad (8)$$

Similar to the depth uncertainty of a 3D point, the location uncertainty of a camera pose increases if the relative angles between the camera and its linked 3D points decrease. As the relative angles get smaller the spatial distance between the 2D image correspondences of the 3D points declines as well. This leads to an increasing influence of the 2D location uncertainty and consequently an increased depth uncertainty of the pose estimate. In order to ensure that the influence of the 2D uncertainty is kept minimal, we reward large distances between the image correspondences of the 3D map points. Hence, we define the relative score of a bin not only on its own value, but in relation to the other bins and their geometric constellation. Thus, we define the relative score $s_r(x)$ of a bin x from the total set of bins B as

$$s_r(x) = q_x \cdot \max_{y \in B}\{q_y \cdot \frac{d(x,y)}{d_{max}(x)}\}, \quad (9)$$

where $d(x, y)$ is the relative distance between the bins x and y and $d_{max}(x)$ is the maximum achievable distance. Through this formulation a bin can only score high, if it has a good quality score itself and there exists another bin with a high quality which is located far-away in the image. In averaging over all bins we then obtain what we call level-0-quality q_0. Note that this measure rewards views with well distributed image correspondences as these views are very robust to partial occlusion.

Robustness against the navigational imperfection is achieved through a "robust" pyramid based on the 8×8 grid. In the pyramid each higher level has half the grid size of the previous level. We disregard the outer regions of the image in the higher pyramid levels as the inner parts of the image are more likely to survive translational and rotational navigation errors. For our experiments, we have chosen to disregard the outmost ring of bins on level 1 and 2 and an additional ring on level 3. In building the pyramid with maximum rather than mean values, we receive higher scores for scenes that have high quality features, which are evenly distributed in the image, but do not cover all of the scene. The scores of each pyramid level are averaged and then each level-x-quality represents the localization quality with respect to a different level of feature density and distribution as well as robustness. The final localization score is obtained by averaging all levels.

IV. Point Generation Likelihood

A topic which is strongly neglected by current passive monocular SLAM approaches is the fact that not every scene or viewpoint is equally suited for map generation. Information is not found in equality and repetition, but in change and variation. In order not to lose the visual localization, it has to be ensured that the camera only adopts views which contain enough useful information.

To locate unmapped areas that are well-suited to create new map points, we make use of unmapped 2D feature points as well as the already mapped 3D points.

First of all, we separate already mapped areas from the unmapped ones by rejecting all potential points of a bin if the bin contains more than a threshold t reprojected map points. An example can be found in Fig. 4.

Then we estimate a depth probability distribution of the current view using the available depth information of the currently tracked map points.

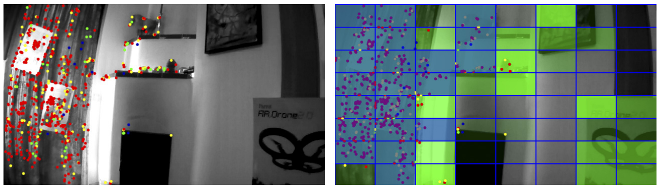


Fig. 4. 2D feature point rejection scheme for the potential point map generation. Left: Original image from PTAM [8], where the colored dots represent tracked 3D points. Right: The original image with an 8×8 grid. The intensity of the green color corresponds to the number of feature points located in the bin. The blue fields represent bins for which no potential 3D points are projected, as the bin already contains enough map points.

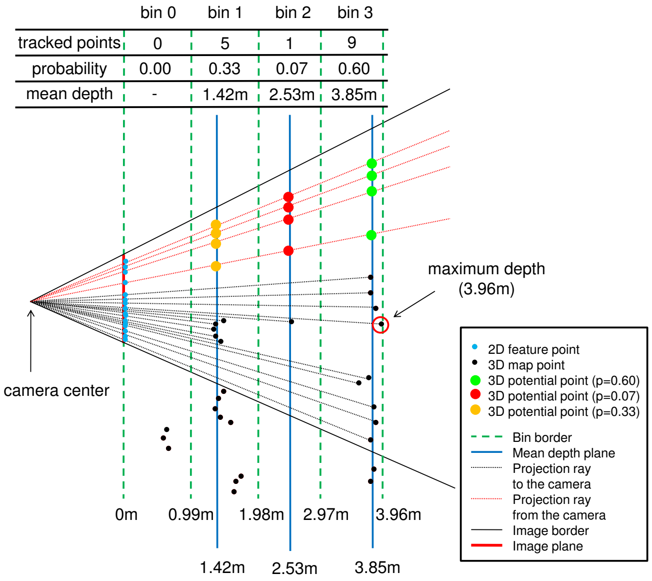


Fig. 5. Potential point map generation. **Construction of the variable depth distribution (VDD):** Set the maximum value range to the maximum depth value of all currently tracked 3D map points and split the depth value range into k equal parts (here: $k = 4$, experiments: $k = 20$). Assign each tracked 3D map point to one of the k bins according to its depth value. For each bin calculate the probability of a map point being in this bin and the mean depth of all points contained in a bin. **Potential point projection scheme:** For each suitable 2D feature point project n potential points. n is the number of bins with a point occurrence greater than zero (here: $n = 3$).

To keep the computational effort manageable, we use a discrete depth approximation, which we call "variable depth distribution" (VDD). Using a random subset of 2D feature points per keyframe ($n = 100$), the VDD and the pose of the keyframe camera, we generate a map of potential 3D points. Opposed to a normal point cloud, a point in this map does not only hold information about the location of the point but also its probability of existence. A detailed description of the VDD construction process as well as the projection of the potential points can be found in Fig. 5.

The purpose of the point generation likelihood is to evaluate which camera pose of a given set of real or virtual poses has the greatest chance of generating new map points.

We define the angular score $s_\alpha(f_p)$ of a potential 3D point f_p based on the triangulation angle $\alpha(k, i)$ between its source keyframe k and the query frame i as

$$s_\alpha(f_p) = \begin{cases} \frac{|\alpha(k,i)|\cdot cos(\alpha(k,i))}{\pi/2} & \text{if } \alpha(k,i) \in [-\pi/2, \pi/2] \\ 0 & \text{otherwise} \end{cases}. \tag{10}$$

Through this formulation we reward large triangulation angles as they can improve the quality of the resulting points, but still respect the recognition probability as in (5).

The score of a single potential point f_p can then be defined as $s_{f_p} = s_\alpha(f_p) \cdot p_{f_p}$, where p_{f_p} is the probability of existence defined through the related VDD.

The final point generation likelihood score for a given query pose can be obtained by summing up the scores of all potential points which have a 2D projection within the image frame of the query pose.

V. Autonomous Explorative Navigation

In this work, the task of the system is to reach an arbitrary *client defined* destination pose in an unknown environment while avoiding collisions and maintaining the visual localization at all times. To achieve this task, the system is not only required to close the distance to the destination, but also explore relevant parts of the scene autonomously.

A. Planning Logic

As most of the scene is still unexplored after start-up, we take a local planning approach rather than a global one. In our approach, we evaluate possible destination poses in the neighborhood of the MAV and analyze the related linear trajectories. Then we transmit the best safely reachable destination pose to the pose controller for execution. As we are on an explorative mission, we replan after a fixed amount of time (10 seconds).

In a first step of our local planning scheme, we aim to reduce the four-dimensional continuous search space (1 angular and 3 linear dimensions) to a fixed-sized set of destination poses. In our experiments, we sample the MAV poses in linear steps of 30 cm up to a distance of 1.2 m and the rotation in steps of 0.2 rad up to 0.6 rad.

In a next step, we have to find the optimal destination in the discrete set of destination poses. Therefore, we evaluate the collision probability and the localization quality of each pose in the set and reject bad poses with fixed thresholds. In the resulting subset we choose the ”best” pose depending on the current system mode.

After finding a suitable pose, we evaluate the localization quality as well as the collision probability on the trajectory. If both values are far from critical on the discretely sampled trajectory, we have found the optimal reachable destination. In the opposite case, we retreat to the previous step, select the next best destination and recheck the reachability constraint.

B. System Modes

To ensure that all requirements are fulfilled simultaneously, we design our system as a hierarchical state machine, where each state (mode) has its own position in the hierarchy depending on its importance for the overall safety of the system, as shown in Fig. 6. Although we respect the matter of navigational safety and localization maintenance in our planning phase, it is highly likely that due to external disturbances some of the safety constraints will be violated during the execution and need immediate attention.

Goal-Striving (GS). The main purpose of the GS mode is to close the distance to a *client defined* destination and thus the optimization criterion for path planning is the distance to the destination.

Strategic Exploration (SE). This mode has two purposes, namely active map point generation and free space carving. When the system is unable to safely reduce the distance to the destination, it changes the optimization criterion to the point generation likelihood. As not all potential points are relevant for the current task, we reduce the set of potential points to a ”useful” subset. Therefore, we discretely sample a linear trajectory from the current MAV pose to the destination and estimate at which point along this trajectory the visual localization will very likely get lost (localization quality < 0.35). For the calculation of the point generation likelihood, we then only consider potential points that would be visible from this camera pose.

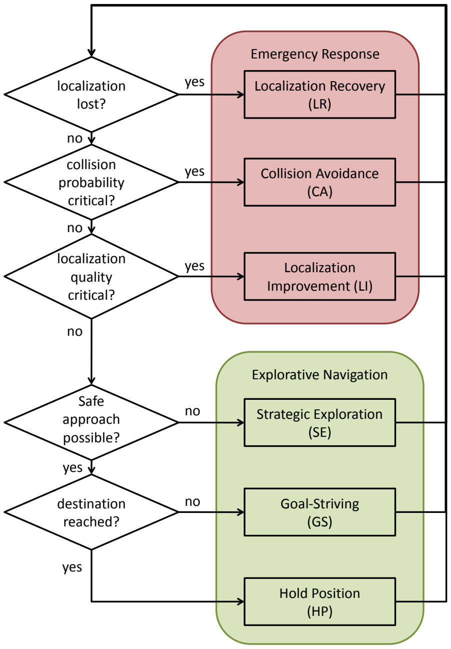


Fig. 6. Diagram of the basic system logic and the corresponding system modes. The system modes can be categorized into two types of planning logic; the emergency response logic and the explorative navigation logic. If the system enters an emergency mode this means that one of the basic needs of the system is not satisfied and needs immediate attention. The order in the decision tree can be seen as a ranking of the modes by their importance and defines which mode can be interrupted by another mode.

To ensure the navigational safety of our system, we only allow the MAV to pass through parts of the scene that very likely do not contain any obstacles; i.e. free space. To grant our system enough free space to navigate towards a *client defined* destination, we generate a strategic destination such that the MAV looks in the direction of a newly received destination.

Collision Avoidance (CA). In this emergency scenario the MAV is in a physically dangerous situation. This means that it moved either too close to a known obstacle or to unknown parts of the scene which might contain obstacles (collision probability > 0.4). We calculate the collision probability based on a probabilistic occupancy grid [19]. As it is crucial in such a situation to react immediately, we choose the closest keyframe pose which is sufficiently safe (collision probability < 0.3) as destination.

Localization Improvement (LI). In the case that localization quality reaches a critical level (< 0.35), our system strives to improve the stability of the localization (> 0.4) while it concurrently takes care to keep the collision probability reasonably low (< 0.4). We take all keyframe poses and the related trajectories into account.

Localization Recovery (LR). When the system loses the visual localization, we immediately set the closest keyframe pose as a destination and navigate a short time (10 seconds) based on IMU data only in the attempt to restore the visual localization.

VI. Experiments

In our experiments we first evaluate our point recognition probability model and the localization quality measure. Then we demonstrate the full functionality of our system in two challenging scenarios.

A. Setup

We can split our hardware setup into an active component (MAV), a computational component (notebook), and a human interface, which allows the user to take over in the case of an emergency (notebook and/or gamepad).

We use a Parrot AR.Drone 2.0 as an active component. It has a forward looking camera, which we run at a frame rate of 10 fps with a resolution of 640×360 px and which is the only exteroceptive sensor used by our implementation. As computational component we use a notebook with a 2.4 GHz quadcore processor.

For monocular SLAM, we use Parallel Tracking and Mapping (PTAM) [8] with some adaptations by Weiss et al. [20] and ourselves. This version can be metrically initialized with a known distance between the first two keyframes (scale uncertainty below 4%). Our obstacle detection module uses the OctoMap framework [19] for the generation of a volumetric representation of the scene. As a visual control module we use the fuzzy control logic of Wendel et al. [21].

We present our experiments in four independent parts. First of all, we prove the reasonableness of our point recognition probability model by comparing it to the measured probability of PTAM [8]. In the second experiment we show that our localization quality measure is perfectly suited to predict a loss of the visual localization by forcing the system into extreme situations. The last two experiments demonstrate the capabilities of our system in two challenging scenarios. In both cases the system has to safely navigate between way-points in unknown environments while concurrently maintaining the visual localization at all times. Note that all experiments were conducted with the same set of parameters independent of the scene.

B. Point Recognition Probability

In this experiment we compare the point recognition probability of PTAM [8] to our predictive model.

For scale invariance, PTAM uses an image pyramid with four levels. Rotation invariance is achieved through affine warping of the region around a feature point. This warping procedure is built on the assumption that when a new feature point is initialized, the normal of the planar 3D patch around a feature point is oriented parallel to the principle axis of the camera.

In general, it is not possible to find the effective point recognition probability for all possible scenes and scenarios as the assumptions made by PTAM can be violated in many ways. Thus, we restrict this experiment to the ideal case, where none of the assumptions are violated, to set an upper bound on the point recognition probability.

We split the experiment into three parts. For all parts we initialize the system so that the image planes of the two initial keyframes are oriented parallel to a planar feature target with the size 0.5 m x 1 m.

In the first experiment, we analyze the relation between the point recognition probability and the viewing angle. We initialize the system with two keyframes and afterwards disable any further point generation. Then we move the camera along an arc with a radius of 2m such that the optical axis is always pointing at the center of the feature target. We record the number of successfully tracked points versus the number of map points that could have been tracked until the localization fails. The average results of seven runs can be found in Fig.7a.

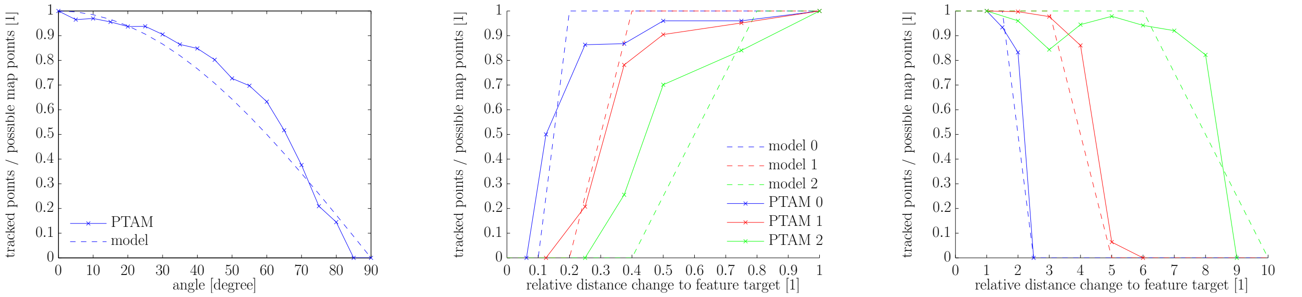
In the next two experiments, we analyze the effect of changing the relative scale, one where we move the camera closer to the feature target and one where we move farther away. We analyze the features of each layer individually by only using the features of one layer at a time for tracking. Both experiments were repeated four times and are displayed in Fig.7. This comparison clearly shows that our model is a reasonable approximation of the actual point recognition probability. For the determination of the parameters of the scale depending model, it is sufficient to determine the properties of the system at the finest scale level ($l = 0$) and then multiply these parameters with 2^l for the other scale levels (for PTAM [8]: $S_{*,l=0} = [0.1, 0.2, 1.5, 2.5]$). In contrast, the other model, which only depends on the viewing angle, does not require any parameters and is valid for every structure-from-motion approach that is based on small feature patches.

C. Localization Quality

We focus the evaluation of localization quality on the effects of rotation as it is the easiest way to lose the visual localization. The experiment was conducted in two different scenes; sparsely textured and richly textured.

After the generation of 7 keyframes, the keyframe creation is deactivated to lock the state of the virtual scene representation. The parameters of the localization quality (α_{cap} and $s_{0,max}$) are automatically tuned after the initialization to adapt to the current scene. Then we rotate the camera in a hand-held manner into 8 different directions (left, left-upwards, upwards, etc.). For each direction, we start with a central view and rotate the camera in the chosen direction until the localization is lost. Afterwards the camera is returned to the central view for the next direction.

After the automatic tuning step, the localization quality is approximately 0.55 for the sparsely textured scene and 0.5 for the richly textured scene for the central view. If we take a look at the statistics in Table I, we can observe that for both scenes the localization quality is always clearly beneath 20 percent at the time of the localization loss. This means that our measure is perfectly suited for the prediction of states



(a) Varying viewing angle.

(b) Decreasing distance to the feature target.

(c) Increasing distance to the feature target.

Fig. 7. Comparison of the measured point recognition probability of PTAM [8] to the proposed model. In (b) and (c) each scale level (0-2) is analyzed and modeled independently.

TABLE I

LOCALIZATION QUALITY AT THE POINT OF LOCALIZATION LOSS.

scene	mean	STD	min	max
sparsely textured	0.112	0.061	0.036	0.183
richly textured	0.049	0.054	0.002	0.177

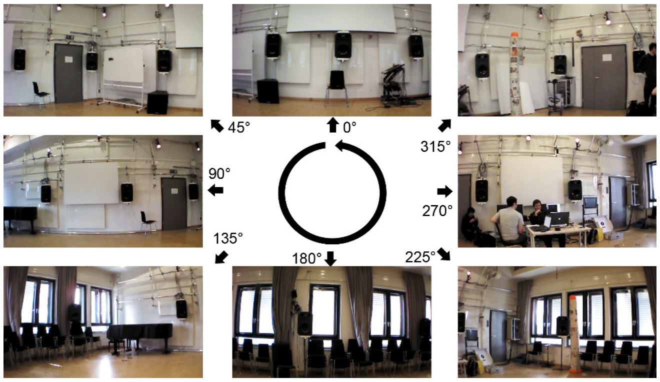


Fig. 8. Experimental scene. A 360° view of the experimental scene captured from the MAV during the experiment. During the experiment the MAV turns counter-clockwise until it has turned a full circle.

with an unstable visual localization and therefore can be used to avoid such critical states.

D. *Full* 360° *Turn Experiment*

For this experiment, we choose an autonomous 360° turn as a high level objective. This objective can only be achieved through generating a ”reasonable” baseline which allows the construction of new 3D map points. We conduct this experiment in a barely textured laboratory, as depicted in Fig. 8.

In order to achieve a 360° turn we send 4 destination poses to the system, where the yaw angle is increased by 90° from one destination pose to the next.

During the execution our system autonomously evaluates the localization quality of all possible motions and thereby is able to avoid states in which the localization is very likely to get lost. If the current map is insufficient to ensure a localization-safe navigation, our system enters its explorative mode to generate new useful map points. During this experiment this event occurred five times of which Fig. 9 depicts three samples.

Although the system had to cope with network lags from up to 20 seconds during the experiment, it was able to complete the full 360° turn and even automatically close the loop at the end of the experiment. In Fig. 10 we show the autonomously constructed map of the room together with a metric ground truth plan of the laboratory.

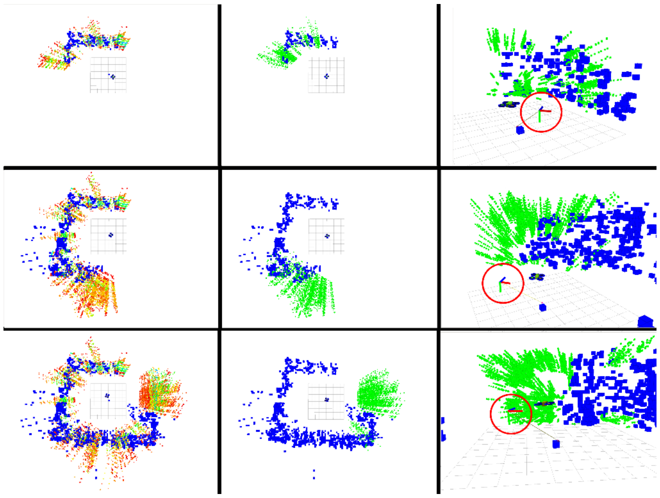


Fig. 9. Strategic point generation. The dark blue cells can be regarded as known obstacles. The colored dots on the left hand side represent potential 3D map points. The color of these dots codes the likelihood of existence of a point at this location (”red” means low and ”green” high probability). The columns in the center and on the right side depict the subset of potential points which the system deemed to be ”relevant” for the current task. The column on the right side additionally shows a ”strategic pose” which maximizes the chance of generating new useful map points while localizing successfully at the same time.

E. *Multi-Level Experiment*

The previous experiment mainly focused on the maintenance of the monocular localization and the generation of new map points. In contrast, this experiment additionally evaluates the capability of safely navigating through an unknown scene with multiple floor levels. The high-level objective in this scenario is to move across the stairs and face the antique desk as shown in Fig. 11. This means that additionally to a 90° turn, the MAV has to close a distance of roughly 4m. While the stairs inherently increase the navigational imprecision, the frame around the stairs drastically limits the available space for navigation.

For this experiment we initialize the system in the lower room with the MAV oriented parallel to the stairs. Due to

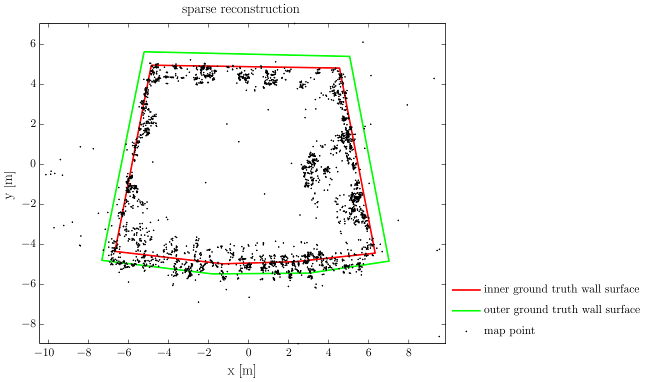


Fig. 10. Sparse reconstruction compared to 2D ground truth. The black dots are the 3D map points of the point cloud after the full 360° turn. The red and green lines represent the inner and outer wall surface of the laboratory according to the building floor plan. For the alignment of the wall ground truth with the point cloud only translation and rotation were taken under consideration to maintain the scale.

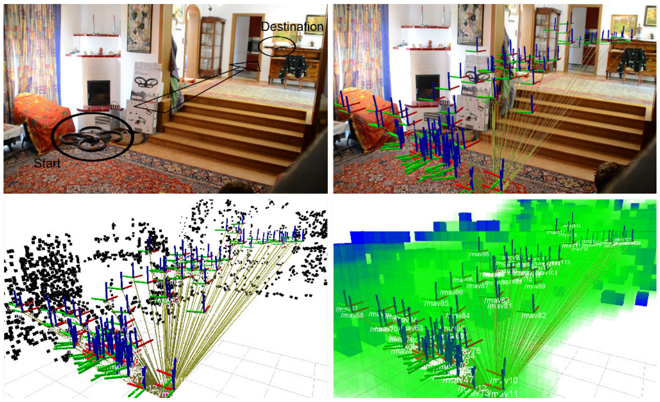


Fig. 11. Multi-Level Experiment. The top-left image shows the high level objective. Each other image shows the same discretely sampled trajectory (at 1 Hz) with a different background. The background in the top-right image is the actual scene, whereas the bottom-left image only shows the sparse reconstruction of the scene (point cloud). The bottom-right image shows the volumetric representation of the scene. Blue means that a cell is occupied (obstacle) and transparent green means that a cells is very likely to be free of obstacles (free space).

the limited space available, we only grant the system a very small initial free space cuboid with $1.2m$ in the x-direction (towards the curtains), $1.6m$ in the y-direction and $2m$ in the z-direction (from the floor upwards). To increase the expressiveness of this experiment, we repeat this experiment six times. In Fig. 11 we display the trajectory of a single flight.

In the selected flight it took the MAV approximately 100 seconds after the receipt of the destination pose to reach this destination. Throughout six flights it took the MAV 187.3 seconds on average to reach the destination, with a standard deviation of 68.7 seconds. On average the system entered 1.33 times the strategic mode to generate new points (STD: 1.03). In all experiments the MAV was able to reach the destination without any collision or loss of the visual localization. A better visualization can be found in the accompanying video.
`http://aerial.icg.tugraz.at`

VII. Conclusions

In this work, we present a set of novel measures that allows the fast evaluation of the localization stability of a monocular system and likelihood of generating new points without any prior knowledge about the topology of the scene. Although our approach was intended for MAVs, it can be straightforwardly extended to other robotic platforms by adjusting the pose sampling strategy and motion model. To the best of our knowledge, we are the first to succeed in demonstrating a fully functional system capable of autonomous explorative navigation only using a single passive camera as exteroceptive sensor.

References

[1] S. Grzonka, G. Grisetti, and W. Burgard, “Towards a navigation system for autonomous indoor flying.” in *ICRA*, 2009.

[2] R. He, S. Prentice, and N. Roy, “Planning in information space for a quadrotor helicopter in a gps-denied environment.” in *ICRA*, 2008.

[3] W. Morris, I. Dryanovski, and J. Xiao, “3d indoor mapping for micro-uavs using hybrid range finders and multi-volume occupancy grids,” in *RSS 2010 workshop on RGB-D: Advanced Reasoning with Depth Cameras*, 2010.

[4] A. Bachrach, S. Prentice, R. He, P. Henry, A. S. Huang, M. Krainin, D. Maturana, D. Fox, and N. Roy, “Estimation, planning, and mapping for autonomous flight using an RGB-D camera in GPS-denied environments,” *The International Journal of Robotics Research*, vol. 31, no. 11, 2012.

[5] S. Shen, N. Michael, and V. Kumar, “Autonomous Indoor 3D Exploration with a Micro-Aerial Vehicle.” in *ICRA*, 2012.

[6] K. Schauwecker, N. R. Ke, S. Scherer, and A. Zell, “Markerless visual control of a quad-rotor micro aerial vehicle by means of on-board stereo processing,” in *Autonomous Mobile Systems*, 2012.

[7] F. Fraundorfer, H. Lionel, D. Honegger, G. Lee, L. Meier, P. Tanskanen, and M. Pollefeys, “Vision-based autonomous mapping and exploration using a quadrotor MAV.” in *IROS*, 2012.

[8] G. Klein and D. Murray, “Parallel Tracking and Mapping for Small AR Workspaces,” in *ISMAR*, 2007.

[9] A. Wendel, M. Maurer, G. Graber, T. Pock, and H. Bischof, “Dense Reconstruction On-the-Fly,” in *CVPR*, 2012.

[10] R. A. Newcombe, S. Lovegrove, and A. J. Davison, “DTAM: Dense tracking and mapping in real-time.” in *ICCV*, 2011.

[11] H. Strasdat, J. M. M. Montiel, and A. J. Davison, “Scale drift-aware large scale monocular SLAM.” in *Robotics : Science and Systems VI*, 2010.

[12] C. Beder and R. Steffen, “Determining an initial image pair for fixing the scale of a 3d reconstruction from an image sequence.” in *DAGM*, 2006.

[13] S. Wenhardt, B. Deutsch, E. Angelopoulou, and H. Niemann, “Active Visual Object Reconstruction using D-, E-, and T-Optimal Next Best Views,” in *CVPR*, 2007.

[14] A. J. Davison and D. W. Murray, “Mobile Robot Localization using Active Vision,” in *ECCV*, 1998.

[15] S. Haner and A. Heyden, “Optimal view path planning for visual SLAM,” in *Scandinavian Conference on Image Analysis*. Springer-Verlag, 2011.

[16] E. Dunn and J.-M. Frahm, “Next best view planning for active model improvement,” in *BMVC*, 2009.

[17] C. Hoppe, A. Wendel, S. Zollmann, K. Pirker, A. Irschara, H. Bischof, and S. Kluckner, “Photogrammetric camera network design for micro aerial vehicles,” in *Computer Vision Winter Workshop*, 2012.

[18] H. Strasdat, J. M. M. Montiel, and A. J. Davison, “Real-time monocular SLAM: Why filter?” in *ICRA*, 2010.

[19] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, “OctoMap: An efficient probabilistic 3D mapping framework based on octrees,” *Autonomous Robots*, vol. 34, 2013.

[20] S. Weiss, M. W. Achtelik, S. Lynen, M. Chli, and R. Siegwart, “Real-time onboard visual-inertial state estimation and self-calibration of MAVs in unknown environments,” in *ICRA*, 2012.

[21] A. Wendel, M. Maurer, M. Katusic, and H. Bischof, “Fuzzy visual servoing for micro aerial vehicles.” in *Austrian Robotics Workshop*, 2012.