19th International Conference on Information Fusion
Heidelberg, Germany - July 5-8, 2016

Scaling the World of Monocular SLAM with INS-Measurements for UAS Navigation

Daniel Bender*, Fahmi Rouatbi*, Marek Schikora*, Daniel Cremers† and Wolfgang Koch*
*Department Sensor Data and Information Fusion, Fraunhofer FKIE, Wachtberg, Germany
Email: {daniel.bender, fahmi.rouatbi, marek.schikora, wolfgang.koch}@fkie.fraunhofer.de
†Computer Vision Group, Technical University of Munich, Garching, Germany
Email: cremers@in.tum.de

***Abstract*—The usage of a global positioning system (GPS) corrected inertial navigation system (INS) seems to be advantageous to camera-based pose estimation algorithms for outdoor navigation. The GPS signals lead to a geographical location with an accuracy of a few meters or even up to some centimeters in a setup utilizing correction data. Performing the pose estimation with a camera system has several disadvantages. The high amount of data has to be processed and the robustness may vary because of different environment and weather conditions. Nevertheless, cameras are the most used sensors for various tasks with unmanned aerial systems (UAS) because of their low cost and weight. The research results for vision-based simultaneous localization and mapping (SLAM) algorithms show the great capabilities of vision-based navigation. Therefore, the possibilities of cameras should be investigated for outdoor navigation in conjunction with an INS or as standalone backup solution in the case of an INS failure. The vision-based pose estimation using a SLAM algorithm provides considerable additional value by providing a map of the observed area. In contrast to keypoint-based approaches, the usage of a semi-dense formulation leads to 3D information for all image regions with gradients and results in a detailed 3D reconstruction. However, the main problem of the monocular SLAM remains the same. The visual pose estimations, and consequently the generated world, suffer from the unknown scale and the corresponding scale drift. In this work we develop a large scale semi-dense SLAM algorithm for the outdoor navigation of UAS equipped with a camera and an INS. This sensor combination increases the robustness of the visual pose estimation process and enables the generation of a very detailed, metric scaled map.**

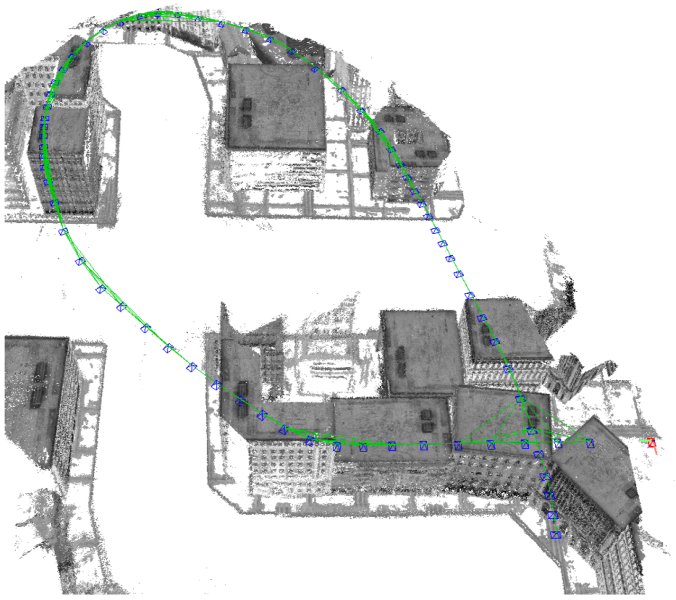


Fig. 1. The proposed approach creates 3D models by performing pixelwise multi-stereo comparisons in images captured with a monocular camera. To achieve a metric scale and to minimize the drift we utilize measurements from an INS. The flight path of the UAS is depicted by the blue camera poses and the image constraints between them. The current camera pose with its coordinate system axes is depicted in red at the lower right of the visualization.

I. INTRODUCTION

The most used sensor combination for the outdoor navigation of unmanned aerial systems (UAS) consists of a global positioning system (GPS) antenna and inertial measurement units (IMU). Their measurements are combined and filtered in a so called inertial navigation system (INS) which results in pose information describing the position and orientation of the device at a hight frequency.

An alternative to the navigation with INS is the usage of camera data. This can be realized by a real-time and robust simultaneous localization and mapping (SLAM) system. Camera-based navigation systems have the advantage of low cost and weight, but suffer from the processing power needed for the calculation of pose information out of pixel measurements. Engel *et al.* presented an approach called large-scale direct monocular SLAM (LSD-SLAM), which runs in real-time on a CPU [1]. In contrast to previous mono camera-based SLAM systems, it uses all image regions with gradients for pixelwise stereo comparisons over multiple frames. This leads to a very robust tracking and detailed maps. However, utilizing only cameras causes a drift in the estimated poses while moving the camera in new, previously not visited, regions. The reason for this is the workflow of all SLAM algorithms, which create pose information out of chained relative transformations between previous estimations. This results in summing up small errors over time, similar to the integration drift for processing the inertial sensors of an INS. The latter get usually corrected by independent GPS measurements, which have an accuracy in the range of a few centimeters by using real time kinematic (RTK) information.

Consequently, the INS leads to very accurate pose estimations, but provides no information about the environment. The LSD-SLAM generates very detailed maps of the observed

978-0-9964527-4-8

area, but the calculated poses have a hight tendency to drift in large scenarios. Furthermore, the created map has no reference to the metric or any geodetic system. A combination of these two approaches should be able to create very detailed, metric scaled and moreover georeferenced maps of the observed area in real time (Figure 1). The algorithm for the creation of these maps is the main contribution of the presented work.

II. RELATED WORK

Using a sensor combination of an INS and a camera for UAS navigation was examined by various authors. The integration of the optical flow as vision component into the navigation with an INS is one possibility of this type of sensor fusion [2], [3]. Besides the optical flow, various approaches use landmarks to estimate the movement of the camera. For example, [4] combined inertial measurements of an IMU with point landmarks tracked by stereo visual odometry in an extended Kalman filter framework. This enables the navigation of large distances without GPS. Likewise, most approaches utilizing a SLAM algorithm work with a small set of keypoints selected by a feature detector. These points are tracked over time and the corresponding 3D points are optimized in a bundle adjustment. A famous example is the Parallel Tracking and Mapping (PTAM) framework [5]. The original purpose was the realization of a small Augmented Reality workspace, but the algorithm was also applied in robotics. There are various approaches using PTAM for the navigation of aerial platforms. Using its pose estimation for a controller-based stabilization of an autonomous UAS allows the navigation without additional sensors in small indoor environments [6]. A more accurate and robust approach was presented by Weis *et al.* [7]. They realized a navigation algorithm combining a camera and an IMU. The focus, besides the integration of PTAM into the navigation framework of the UAS, were various adaptations to work on an on-board processor with limited processing power. One of the main drawbacks of PTAM and its application in the UAS navigation is the scalability for large areas. While the number of keyframes grows, the mapping threads can only perform updates for a subset of keyframes and observed 3D points. The tracker scales better but is also influenced by the high number of 3D points. To overcome this issue, the multiple map adaption of PTAM from Castle *et al.* [8] was adapted to realize a fully autonomous UAS navigation in large areas [9]. They used the pose estimation of the monocular SLAM approach on a horizontally viewing camera to correct the drifting IMU measurements. A more robust tracking can be realized by using all image information instead of a subset of keypoints. It was shown, that the convex optimization of a photometric dataterm in an energy functional results in superior tracking performance [10]. On the other hand, a dense volumetric reconstruction with an UAS was presented by Wendel *et al.* [11]. Both approaches achieve real-time performance with a graphics processing unit (GPU).

In contrast to this research, where keypoint-based or dense approaches were used, we investigate and adapt semi-dense LSD-SLAM using all image regions with gradients [1]. This promises a robust tracking and more detailed mapping compared to the keypoint-based approaches, while being less computationally intensive than the dense algorithms. To overcome the drift of the LSD-SLAM, we utilize INS measurements. By integrating filtered INS pose information in the mathematical formulation of the LSD-SLAM, we improve the pose estimations and map quality without the need of additional computational power. Quiet the contrary, some parts like the identification of loop closure becomes less expensive because of a smaller search region on the basis of more accurate pose estimations.

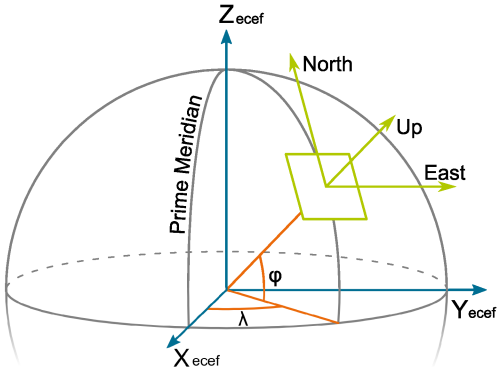


Fig. 2. Both, the navigation systems of the INS and the world reference frame are based on East, North, Up (ENU) coordinate systems (green). They are local Cartesian coordinate systems with the origin tangential to the earth ellipsoid. Furthermore, we use the global earth-centered, earth fixed (ECEF) coordinate system (blue), to convert between ENU-coordinates and GPS measurements taken in latitude (φ) and longitude (λ) as polar coordinates (orange).

III. DEFINITIONS

We define the world frame W as an ENU coordinate system with its origin being approximately in the middle of our flight area (Figure 2).

The mathematical concept of Lie groups, which represent smooth topological groups, is used to describe transformations. Examples are the 3-dimensional special orthogonal group $SO(3)$ for all 3D rotations and the special Euclidean group $SE(3)$ representing rigid body motions in the 3D space. The algebraic structures $so(3)$ and $se(3)$, named Lie-Algebras, are vector spaces which allow a minimal representation of the corresponding Lie-Groups. We will give some more insight of the associated theory in the following definition of rigid body motions, but refer the reader for a comprehensive explanation to [12].

The rigid body motion $g_{i\mathrm{W}}$ describes the camera pose at the middle exposure time $t_i, i = 1, 2, ..., n$ with regard to the world frame W. Likewise, g_{ji} specifies the rigid body motion of the camera pose from time t_i to t_j which describes the transformation between the corresponding frames.

In general, a rigid body motion $g \in SE(3)$ describes how the points of a rigid object change over time. Instead of considering the continuous path of the movement, we bring into focus the mapping between the initial and the final configuration of the rigid body motion. This movement can be

described by a rotation matrix $\mathbf{R} \in SO(3)$ and a translation vector $\mathbf{t} \in \mathbb{R}^3$. Consequently, the rigid body displacement $\boldsymbol{G}$ of a 3D point $\mathbf{p} \in \mathbb{R}^3$ can be performed by

$$\boldsymbol{G} : SE(3) \times \mathbb{R}^3 \to \mathbb{R}^3 , \qquad \boldsymbol{G}(g, \mathbf{p}) = \mathbf{R}\mathbf{p} + \mathbf{t} . \tag{1}$$

The representation of the rotational part in form of the overdetermined rotation matrix $\mathbf{R}$ is not suitable for the optimization performed in this work. A minimal representation is required. Being aware of the singularities which can occur by using Euler angles, we represent a rigid body motion by twist coordinates $\boldsymbol{\xi} = (\boldsymbol{t}, \boldsymbol{\omega})^\top \in \mathbb{R}^6$. Thereby, $\mathbf{t} \in \mathbb{R}^3$ describes the translational part, while the skew-symmetric matrix $\hat{\boldsymbol{\omega}} \in so(3)$ represents the rotational part of the full motion. The rotation angle in radians is encoded in the Euclidean norm $||\boldsymbol{\omega}||$. An element $\hat{\boldsymbol{\xi}} \in se(3)$ can be written in its homogeneous representation as

$$\hat{\boldsymbol{\xi}} = \begin{pmatrix} \hat{\boldsymbol{\omega}} & \mathbf{t} \\ 0 & 0 \end{pmatrix} . \tag{2}$$

Given $\hat{\boldsymbol{\xi}} \in se(3)$, we get a rigid body motion by the matrix exponential which is defined as the always converging power series:

$$\exp : se(3) \to SE(3) , \qquad \exp(\hat{\boldsymbol{\xi}}) = \sum_{k=0}^{\infty} \frac{\hat{\boldsymbol{\xi}}^k}{k!} . \tag{3}$$

The inverse of this exponential map is defined as the logarithm:

$$\exp(\log(g)) = g . \tag{4}$$

This leads to an alternative formulation of (1) using twist coordinates to describe the displacement of a point $\mathbf{p} \in \mathbb{R}^3$ according to the rigid body motion g as follows:

$$\boldsymbol{G} : SE(3) \times \mathbb{R}^3 \to \mathbb{R}^3 , \qquad \boldsymbol{G}(g, \mathbf{p}) = \exp(\hat{\boldsymbol{\xi}})\mathbf{p} . \tag{5}$$

The concatenation of two rigid body displacements is a simple multiplication in the $SE(3)$ space and allows us to define the pose concatenation operator:

$$\circ : SE(3) \times SE(3) \to SE(3) , \tag{6}$$

$$g_{kj} \circ g_{ji} = g_{kj} \cdot g_{ji} = g_{ki} . \tag{7}$$

This relationship can easily be utilized for the twist coordinate representation by applying the exponential mapping stated in (3).

To describe an image, we define the mapping from the image plane Ω to the intensity values as the function $I : \Omega \to \mathbb{R}$. Likewise, we specify the inverse depth map $D : \Omega \to \mathbb{R}^+$ and the corresponding depth variance $V : \Omega \to \mathbb{R}^+$.

IV. LSD-SLAM

This section gives a brief overview of the LSD-SLAM algorithm, which is mandatory to understand the INS-based LSD-SLAM introduced in this work. For a more detailed description we refer to the original publication of Engel *et al.* [1].

The LSD-SLAM algorithm computes a real time estimation of the position and orientation of the camera, also known as pose, in an unknown scene. At the same time, the approach generates a map of the observed environment. The algorithm is divided in a map building and a tracking part which are executed on multiple CPU cores to achieve real time performance. Both parts operate on keyframes, created from new frames based on a weighted combination of distance and angle to the previous keyframes.

A. Tracking

The tracker estimates the pose transformation g_{ji} from the latest keyframe I_i to a new camera frame I_j by the direct minimization of the photometric error. This is achieved by evaluating the intensity difference of the keyframe pixels $\mathbf{x} \in \Omega_i$ and their mapped position in the image plane Ω_j of the new frame:

$$r : \mathbb{R}^2 \times SE(3) \to \mathbb{R} , \tag{8}$$

$$r(\mathbf{x}, g_{ji}) = (I_i(\mathbf{x}) - I_j(w(\mathbf{x}, D_i(\mathbf{x}), g_{ji})))^2 , \tag{9}$$

whereby the warp function $w : \mathbb{R}^2 \times \mathbb{R} \times SE(3) \to \mathbb{R}^2$ performs the mapping of a pixel $\mathbf{x}$ from the keyframe I_i to the new image I_j. This is realized by using the inverse depth estimate of the keyframe $D_i(\mathbf{x})$ to generate a 3D point $\mathbf{p}$ which is transformed to the coordinate system of the new camera by $\boldsymbol{G}(g_{ji}, \mathbf{p})$ and then projected into the image plane I_j.

Thus, the overall energy function that is minimized in the tracking process is given by the sum of all residuals for pixels with an inverse depth:

$$E(g_{ji}) = \sum_{\mathbf{x} \in \Omega | D(\mathbf{x}) > 0} s_{\mathbf{x}} \cdot r(\mathbf{x}, g_{ji}) , \tag{10}$$

where $s_{\mathbf{x}}$ defines a robustness increasing weighting scheme based on the size and smoothness of the residuum $r(\mathbf{x}, g_{ji})$ as well as the inverse depth variance of the pixel $V(\mathbf{x})$ as defined in [1]. This least square problem is solved with an iteratively reweighed Gauss-Newton approach.

The resulting pose estimation is compared to a threshold defined for a weighted combination of position and angle differences. If it is lower, the image information of the current frame is used to refine the inverse depth map of the keyframe it was tracked on, otherwise, it will become a new keyframe. In this case its inverse depth map gets initialized by the propagation of information from the previous keyframe.

B. Mapping

After a new keyframe is generated by the tracking process, the previous keyframe gets finalized and consequently processed by the mapping. The mapping only operates on finalized keyframes, which implies that the inverse depth maps of these frames are not updated anymore. To find loop closure candidates, direct image alignments between the new finalized keyframe and spatial close keyframes are performed. In addition to the photometric error defined in (9), a depth residual, penalizing incompatible inverse depth values, is evaluated. This is possible because, in contrast to the tracking, both frames have an inverse depth map at this stage of the algorithm. As a consequence, the scale between the frames is observable and optimized as additional parameter in the direct

image alignment. Thus, the estimated pose transformations g_{ji} between keyframes are estimated as elements of $Sim(3)$, which adds the scale as an additional degree of freedom.

The global map containing the pose information of all keyframes is continuously optimized within g^{2}o, the general graph optimization framework [13]. Therefore, the problem is embedded in a graph by introducing the camera poses $g_{i\mathrm{W}} \in Sim(3)$ as nodes, representing variables to optimize, and the tracked constraints $g_{ji} \in Sim(3)$ between them as edges. An error function for these constraints connecting two camera poses can be defined as follows:

$$e_{ji}(g_{i\mathrm{W}}, g_{j\mathrm{W}}, g_{ji}) = \log(g_{j\mathrm{W}}^{-1} \circ g_{ji} \circ g_{i\mathrm{W}}) \ , \qquad (11)$$

whereby the logarithmic map transforms the rigid body motion to its twist coordinates. This results in an error vector of dimension seven, which is $\mathbf{0}$ if the image constraint perfectly describes the transformation between the two camera poses. The final energy that is minimized can be stated as follows:

$$E(g_{i\mathrm{W}}, ..., g_{n\mathrm{W}}) = \sum_{g_{ji}} e_{ji}^{\top} \Sigma_{ji}^{-1} e_{ji} \ , \qquad (12)$$

whereby Σ_{ji}^{-1} represents the inverse covariance matrix of the twist coordinates. The g^{2}o framework uses the Levenberg-Marquardt algorithm to compute a numerical solution of (12) and therefore needs a good initial guess of the state vector.

V. INS Based LSD-SLAM

The following section describes our adaptations to the LSD-SLAM framework, which utilize measurements from an INS to make the camera-based navigation more robust. The original LSD-SLAM implementation offers clear benefits over the feature point approaches in terms of map details and scalability for large regions. Nevertheless, it suffers from the fundamental disadvantage of a monocular SLAM approach, alias the unknown scale drift. Small tracking errors sum up over time and lead to a drift in the calculated camera pose. In comparison to a stereo camera setup, this drift is even bigger because of the unknown scale of the scene. Engel *et al.* apply an appearance-based image comparison in addition to the spatial distance between keyframes to detect regions already observed and perform a loop closure. This approach costs additional computing power and is likely to fail in similar-looking outdoor regions.

The usage of an additional sensor in form of a GPS-based INS leads to independent pose measurements in outdoor environments. To achieve an easy fusion with an INS, the coordinate system of the camera should be adapted to the navigational frame of the INS. Therefore, each camera pose g_{ji} is initialized with the composition of the associated INS pose and the mounting offsets between the devices, in the following referred to as g_{ji}^{INS}. The initialization of the original LSD-SLAM algorithm is performed by generating a random depth map with a mean depth of one and a high depth variance for all image regions with gradients larger than a defined threshold. The alternative approach realized in our work is based on a stereo initialization with pose information from the associated INS measurements. This allows the generation of an initial depth map in metric scale, by calculating depth estimates for all pixels with gradients above a threshold based on the INS transformation between two frames (Figure 3).

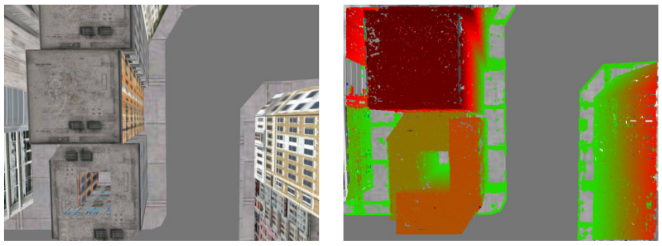


Fig. 3. The input image on the left is tracked with respect to the inverse depth map of the latest keyframe displayed on the right. The metric scaling of the proposed approach leads to absolute depth information. The applied color coding fades from green (30 m) over red (15 m) to black ($\leq$7.5 m). Image regions without gradients are represented in their original greyscale intensity.

A. Modified Tracking

The INS measurements provide an upper bound for the camera-based pose estimation, and thus are able to minimize the drift of the LSD-SLAM poses. Therefore, we perform the pose optimization stated in (10) while constraining the pose update g_{ji} to be in the standard deviation range of the corresponding INS pose composition:

$$g_{ji}^{\mathrm{INS}} = g_{j\mathrm{W}}^{\mathrm{INS}} \circ (g_{i\mathrm{W}}^{\mathrm{INS}})^{-1} \ , \qquad (13)$$

$$g_{ji} \in [g_{ji}^{\mathrm{INS}} - 4\sigma^{\mathrm{INS}}, g_{ji}^{\mathrm{INS}} + 4\sigma^{\mathrm{INS}}] \ . \qquad (14)$$

Using $4\sigma^{\mathrm{INS}}$ corresponds to the fact that we compare our tracking update to two independent INS poses and for both, the error is for more than 95% of the samples in the range of two times the standard deviation. Thus, we limit the difference between the pose update and the corresponding INS measurements to be smaller than $2 \cdot 2\sigma^{\mathrm{INS}} = 4\sigma^{\mathrm{INS}}$. This constraint is in some rare cases too strict. Nevertheless, the produced results are more accurate compared to the usage of $6\sigma^{\mathrm{INS}}$ as maximum distance for the calculated update.

B. Modified Mapping

The change in the tracking part can only limit the drift, but not eliminate it. In order to achieve this goal, we add INS measurements as unary edges to each graph node representing a camera pose. An error function for these constraints on the camera poses can be defined as follows:

$$e_i(g_{i\mathrm{W}}, g_{i\mathrm{W}}^{\mathrm{INS}}) = \log((g_{i\mathrm{W}}^{\mathrm{INS}})^{-1} \circ g_{i\mathrm{W}}) \ . \qquad (15)$$

This results in an error vector, which is $\mathbf{0}$ if the camera pose estimation can be perfectly described by the corresponding INS measurement. Adding this constraint to the pose graph extends (12) to the following energy minimized in our approach:

$$E(g_{i\mathrm{W}}, ..., g_{n\mathrm{W}}) = \sum_{g_{ji}} e_{ji}^{\top} \Sigma_{ji}^{-1} e_{ji} + \sum_{g_{i\mathrm{W}}} e_i^{\top} \Sigma_i^{-1} e_i \ . \qquad (16)$$

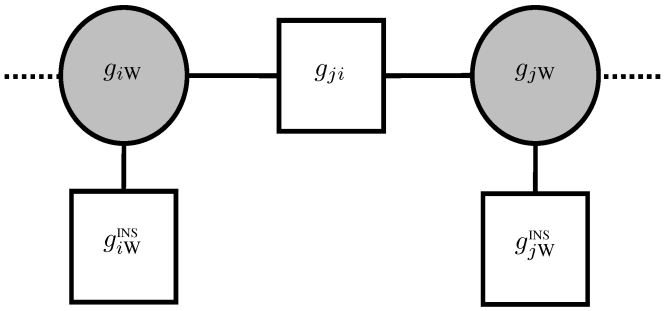


Fig. 4. The objective function of the mapping stage can be illustrated by a graph. The measurements (boxes) are connected to our variables (circles). In our setup, the latter are the camera poses. In addition to constraints based on image information between the camera poses, we add one unary edge to each camera pose representing the corresponding INS measurement.

whereby Σ_{ji}^{-1} and Σ_{i}^{-1} represent the inverse covariance matrices of the twist coordinates. Adding these unary edges eliminates the scale drift and makes the scale estimation performed in the LSD-SLAM algorithm superfluous. This allows us to use $SE(3)$ instead of $Sim(3)$ constraints in the pose graph (Figure 4).

VI. Evaluation setup

In the following, we describe our setup for the numerical studies performed to validate the proposed approach.

Using the robot simulation Gazebo [14] allows testing our approach in a realistic scenario and gather valuable information for real world experiments. The modeling, control and simulation of a quadcopter UAS within Gazebo were developed by Meyer *et al.* [15]. The generated pose information are on the one hand used as ground truth in the evaluation and on the other hand the basis for the generation of the noisy INS data. For the latter, we add white Gaussian noise to the poses using a standard deviation of $\sigma_{\mathrm{t}} = (0.02\,\mathrm{m}, 0.02\,\mathrm{m}, 0.04\,\mathrm{m})$ for the position and $\sigma_{\mathrm{r}} = (0.1°, 0.1°, 0.2°)$ for the rotational components represented by Euler angles. Thereby the altitude as well as the rotation around the z-axis are chosen twice as big as the other components to model the standard error behavior of an INS. The modeled values represent the accuracies of a small INS with RTK corrected satellite navigation measurements [16].

Furthermore, we use the camera simulation provided by Gazebo to generate images from a downward facing camera with 640×480 pixels at a frame rate of 30 Hz. The chosen aperture angle of the camera of $100°$ corresponds to a standard wide angle lens. The mounting offset between the camera and the INS is known in our simulation. They can be estimated for real world experiments by performing a boresight calibration of the sensor setup with data collected in a previous flight [17]. We used the 3D model 'The City' created by Herminio Nieves as environment in our evaluation (Figure 5). He published this [18] and other models free for commercial and non commercial use.

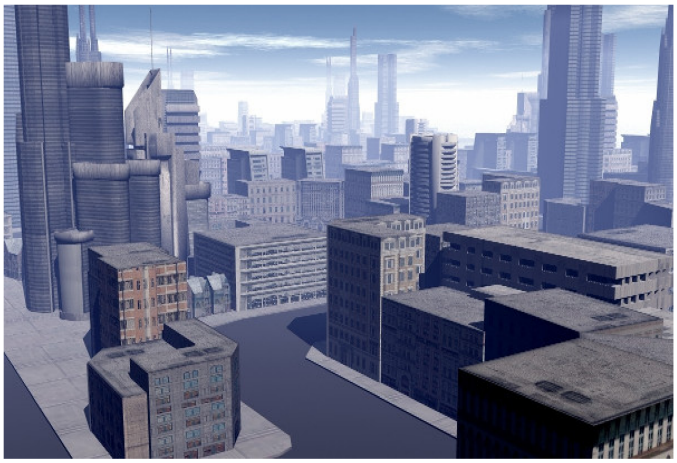


Fig. 5. Rendered image of the 3D model 'The City' from Herminio Nieves. We used the textured mesh as environment in our Gazebo simulations. Captured images of this model contain, a mixture of regions with a lot of gradients (most houses), some gradients (sidewalks) and without gradients (streets).

VII. Performance measures

We evaluate our results in two different ways. On the one hand we compare the pose estimation with the ground truth and on the other hand we perform a matching between the generated point cloud and the 3D mesh used as environment in our simulation.

Thereby, the pose comparison is realized separately for the translational and rotational part. Comparing two positions using the Euclidean norm is straightforward:

$$\Phi_t : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}^+ \,, \tag{17}$$

$$\Phi_t(\mathbf{t}_1, \mathbf{t}_2) = \sqrt{\langle \mathbf{t}_1 - \mathbf{t}_2, \mathbf{t}_1 - \mathbf{t}_2 \rangle} \,, \tag{18}$$

whereby $\langle \cdot, \cdot \rangle$ defines the scalar product.

The rotational part of the tracked poses is compared according to the angle difference between the unit quaternions $\mathbf{q}_1$ and $\mathbf{q}_2$ by:

$$\Phi_r : SO(3) \times SO(3) \to [0, \pi] \,, \tag{19}$$

$$\Phi_r(\mathbf{q}_1, \mathbf{q}_2) = 2 \arccos(|\langle \mathbf{q}_1, \mathbf{q}_2 \rangle|) \,. \tag{20}$$

Using the absolute value of the scalar product is necessary due to the ambiguity in the quaternion representation as $\mathbf{q}$ and $-\mathbf{q}$ describe the same rotation [19]. In contrast to comparing Euler angles with the Euclidean norm, the used metric reflects the distance. A smaller value always corresponds to a smaller rotation. Furthermore, the metric is geometrically meaningful as the angle between two orientations is exactly twice the angle between the corresponding unit quaternions.

Our second performance measure is the difference between the generated point clouds and the 3D mesh used as environment. For each point, the smallest distance to the mesh is determined. The visualization of the point to mesh distances allows an easy interpretation and comparison of the achieved results.

VIII. Evaluation

In this section, we present results from numerical studies to validate the proposed approach. The main purpose of this is to analyze the effect of the described integration of INS measurements in the SLAM process by comparing the estimations for the camera poses and the created map with the known ground truth. We performed Monte Carlo simulation tests with 100 trials for the original LSD-SLAM, versions integrating only our tracking or mapping modification and the presented approach using both modifications. The latter will in the following be referred to as INS-based.

Our first evaluated data set was created from a straight flight path at a constant altitude. This simple path without any maneuvers or course corrections shows the main problem of the monocular SLAM approach. Even without large scale changes, the tracked poses show a clear drift. The usage of the proposed approach using INS measurements as additional information prevents the pose drift.

We visualized the Euclidean difference of the translation error over time (Figure 6). The results of the LSD-SLAM grow linearly. At the end of the dataset, after 60 seconds, the mean error is approximately 5 meters. Using the proposed tracking modification limits the error, but cannot eliminate the drift. At the end of the dataset we have a mean error of roughly half as large as using LSD-SLAM without this modification. The exclusive usage of our mapping modification eliminates the drift, but leads to a sawtooth wave. This pattern is based on the workflow of the mapping modification which only operates on keyframes. Between these, the tracking estimates the current pose relative to the latest keyframe. One interesting fact is that the combination of the tracking and mapping modification in the INS-based SLAM seems to eliminate the sawtooth wave completely. Furthermore, this leads to an overall smaller error. The mean values of the Monte Carlo trials for the proposed approach are smooth and the corresponding standard deviations are small (Figure 6).

We use the metric defined in (20) to analyze the difference between the rotational part of the pose estimations and the ground truth (Figure 7). The mean value of the total angular error for the original LSD-SLAM grows slowly after the first few seconds and is at the end of the sequence still below two degree. This states, that the orientation estimation is more reliable compared to the positional results produced by the monocular SLAM system. Again, the proposed tracking modification reduces the error of the LSD-SLAM. The mapping modification shows in the beginning the characteristic sawtooth wave and becomes a lot smoother afterwards. In contrast to the position difference, the combination of these modifications does lead to a smaller error in the first few seconds and is nearly the same afterwards. The results show that we also eliminated the drift in the rotational part of the pose estimation.

Our proposed approach limits the position error to the centimeter range and produces also very small errors in the rotational component. This allows the creation of maps in a metric scale with only small deviations from the ground truth model (Figure 8). As expected, the LSD-SLAM point cloud shows small differences to the ground truth directly after the initialization and higher differences in the upper region of the visualization, corresponding to the end of the flight path. For the INS-based results some points in the outer region reveal a higher distance to the mesh. These points were most likely created by only one image correspondence and not updated by additional observations. It must be noted that the visualized error of the point cloud seems to be a lot smaller than the positional error of the camera poses produced by the original

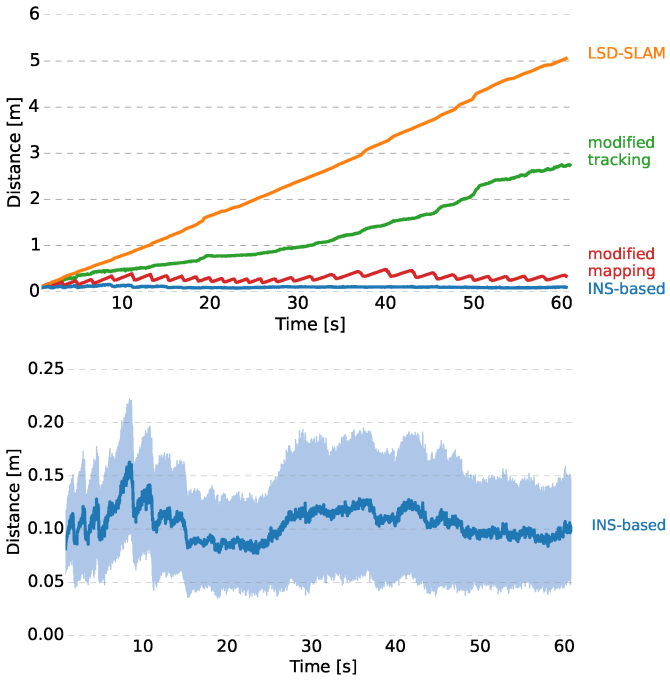


Fig. 6. Mean Euclidean norm of the translation difference to the ground truth data for a straight flight path from 100 Monte Carlo trials. The tracking modification limits the position drift of the SLAM algorithm to roughly one half of the LSD-SLAM results. Our mapping modification leads to a sawtooth wave, which is smoothed in the INS-based SLAM. The latter combines both modifications. For clarification, its mean error in combination with the standard deviation (colored region) is depicted in the lower plot.

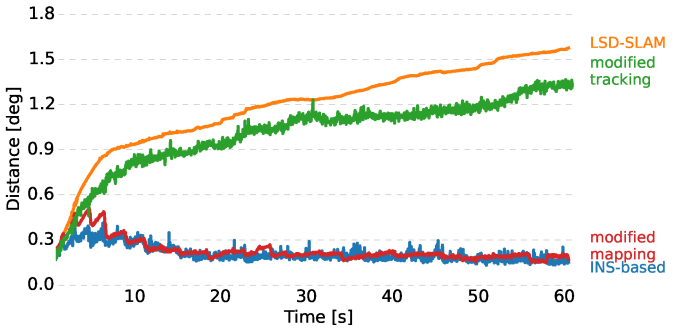
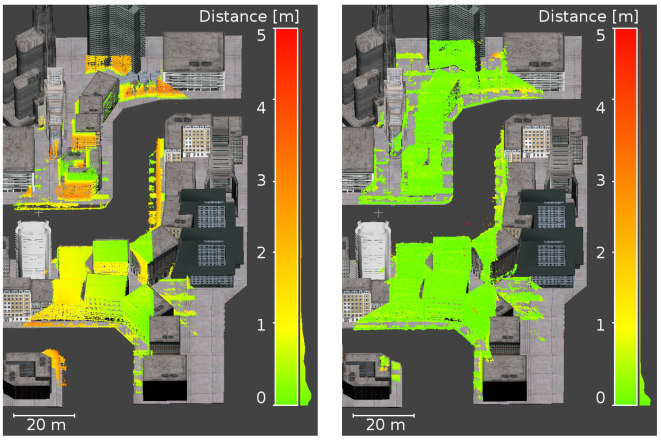


Fig. 7. Mean angular error between the estimations and the ground truth data for a straight flight path from 100 Monte Carlo trials. The results show that the influence of the tracking modification is rather small. The combination of both modifications in the INS-based approach results only in the first few seconds in more accurate pose estimations compared to the modified mapping.



(a) LSD-SLAM (b) INS-based

Fig. 8. Absolute distance of the generated maps, represented as point clouds, to the ground truth mesh. The straight flight from bottom to the top of the visualized section leads to bigger errors in the end region of the LSD-SLAM results, due to the pose drift (left). Using INS information eliminates the drift, which leads to a much more accurate point cloud (right). A histogram of the distances is displayed to the right of the color bar.

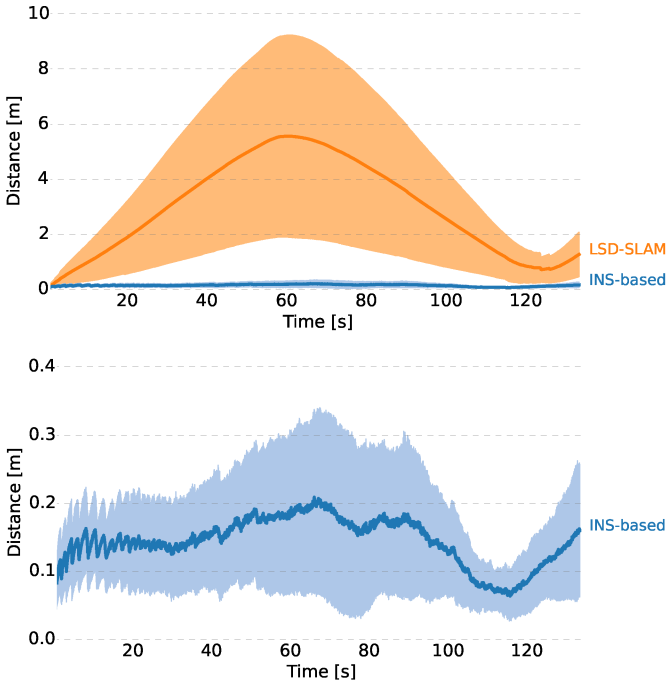


Fig. 9. Mean Euclidean norm of the translation difference to the ground truth data for a circular flight course from 100 Monte Carlo trials. The mean and the standard deviation (colored region) of the LSD-SLAM pose estimations are improving in second half of the dataset and the error starts to grow again after the final loop closure at 120 seconds. The same behavior is, to a lesser extent, also recognizable in the INS-based results.

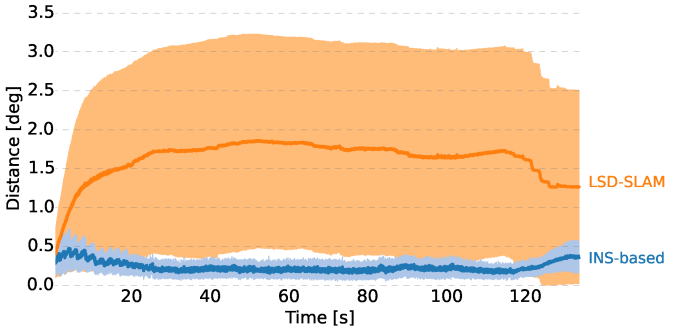


Fig. 10. Mean angular error between the estimations and the ground truth data for a circular flight course from 100 Monte Carlo trials. The loop closure at 120 seconds reduces the error for the LSD-SLAM estimations, but leads to slightly worse poses for the INS-based SLAM. The standard deviations (colored regions around the mean curves) shows the great improvement of the INS-based approach regarding the robustness.

LSD-SLAM. This inconsistency is explained by comparing the distance of each point to its closest point in the mesh instead of its real correspondence. Especially in the evaluated scenario with mainly flat roofs, this has a positive effect on the distance of the pointcloud to the 3D model.

Using a gamepad to control the quadcopter in the simulation environment leads to some rough but realistic movements in form of swings while performing course corrections in our second data set. These movements were tracked without any problems and show the high performance of the LSD-SLAM algorithm. The flight course of this dataset is a circular route at nearly constant altitude (Figure 1). This course allows the algorithm to perform a loop closure and propagate correction to the preceding pose estimations. Nevertheless, the visualizations in all our plots show the poses at the time they were tracked and not the final estimations of the pose graph framework. Thus, the slowly decreasing error in the translational part of the pose estimations (Figure 9) is not the result of the back propagated information from the final loop closure. A closer inspection of the visualization of one realization on this dataset (Figure 1) reveals many connections between the camera poses in the second half of the dataset. We conclude that these constraints have a strong positive effect on the pose graph optimization and result in decreasing the error. Also, the final loop closure occurs slowly by adding more and more constraints between the camera poses to the graph and, as a consequence, the loop closure does not result in a big jump (Figure 9). Applying the presented modifications limits also the rotation errors to very small values (Figure 10). These plots show the high variance of the original LSD-SLAM approach.

The accurate pose estimations have a direct influence on the generated world map. Visualizing the differences between the point cloud and the ground truth depicts the great results produced in real time by the proposed approach (Figure 11).

IX. Conclusion and Future Work

In this work we presented an approach which fuses measurements from an INS into a semi-dense monocular SLAM algorithm. We showed how INS measurements can be used within the tracking and mapping parts of the SLAM workflow to generate metric depth maps and eliminate the drift.

As a consequence, the generated point clouds of the observed areas are also metric and furthermore, due to the utilized GPS measurements, georeferenced. On the basis of RTK corrections, the created maps are accurate in the cen-

(a) LSD-SLAM

(b) INS-based

Fig. 11. Absolute distance of the generated maps, represented as point clouds, to the ground truth mesh. The loop course starting in the lower right leads to large errors in the upper left region, where the distance to the starting point is largest. Using INS information eliminates the drift, which leads to a much more accurate point cloud. A histogram of the distances is displayed to the right of the color bar.

timeter range. In contrast to other computational intensive camera-based algorithms, the results of the presented approach are generated in real time and thus suitable for the UAS navigation. The created maps describe the environment with a lot of details and can be used for various tasks. For example they should allow to perform a change detection or can be used as reference data for the robust navigation in a previous visited area without GPS corrections.

Future work will investigate the proposed procedure in more detail. We will evaluate how to adapt the approach to be suitable for GPS measurements without RTK corrections. Moreover, instead of the filtered INS pose, we plan to integrate the inertial measurements in the form of accelerations and angular rates in LSD-SLAM approach without the usage of GPS measurements. This would be a nice application for GPS denied or distorted environments if the produced navigation solutions show less drift compared to INS or camera only approaches and results additionally in detailed semi-dense maps of the environment. Based on the gathered insight, we will perform real flight experiments and evaluate the approach further.

References

[1] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European Conference on Computer Vision (ECCV)*. Springer, 2014, pp. 834–849.

[2] J. Wang, M. Garratt, A. Lambert, J. J. Wang, S. Han, and D. Sinclair, "Integration of gps/ins/vision sensors to navigate unmanned aerial vehicles," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 37, pp. 963–970, 2008.

[3] F. Kendoul, I. Fantoni, and K. Nonami, "Optic flow-based vision system for autonomous 3d localization and control of small aerial vehicles," *Robotics and Autonomous Systems*, vol. 57, no. 6, pp. 591–602, 2009.

[4] J. Kelly, S. Saripalli, and G. Sukhatme, "Combined visual and inertial navigation for an unmanned aerial vehicle," in *Field and Service Robotics*, 2008, pp. 255–264.

[5] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, 2007, pp. 225–234.

[6] M. Blösch, S. Weiss, D. Scaramuzza, and R. Siegwart, "Vision based mav navigation in unknown and unstructured environments," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 21–28.

[7] 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 *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, 2012, pp. 957–964.

[8] R. Castle, G. Klein, and D. W. Murray, "Video-rate localization in multiple maps for wearable augmented reality," in *12th IEEE International Symposium on Wearable Computers*, 2008, pp. 15–22.

[9] M. Jama and D. Schinstock, "Parallel tracking and mapping for controlling vtol airframe," *Journal of Control Science and Engineering*, vol. 2011, p. 26, 2011.

[10] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "Dtam: Dense tracking and mapping in real-time," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, 2011, pp. 2320–2327.

[11] A. Wendel, M. Maurer, G. Graber, T. Pock, and H. Bischof, "Dense reconstruction on-the-fly," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 1450–1457.

[12] B. C. Hall, *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction (Graduate Texts in Mathematics)*. Springer, 2015.

[13] R. Kümmerle, B. Steder, C. Dornhege, A. Kleiner, G. Grisetti, and W. Burgard, "Large scale graph-based slam using aerial images as prior information," *Autonomous Robots*, vol. 30, no. 1, pp. 25–39, 2011.

[14] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *Intelligent Robots and Systems (IROS), 2004 IEEE/RSJ International Conference on*, vol. 3, 2004, pp. 2149–2154.

[15] J. Meyer, A. Sendobry, S. Kohlbrecher, U. Klingauf, and O. von Stryk, "Comprehensive simulation of quadrotor uavs using ros and gazebo," in *Simulation, Modeling, and Programming for Autonomous Robots*. Springer, 2012, pp. 400–411.

[16] SBG Systems, "Ellipse series: Miniature high performance inertial sensors." [Online]. Available: http://sbg-systems.com/docs/Ellipse_Series_Leaflet.pdf

[17] D. Bender, M. Schikora, J. Sturm, and D. Cremers, "A graph based bundle adjustment for ins-camera calibration," *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci*, pp. 39–44, 2013.

[18] H. Nieves, "The city: 3d model," 2015. [Online]. Available: http://sharecg.com/v/79711/gallery/5/3D-Model/The-City

[19] D. Q. Huynh, "Metrics for 3d rotations: Comparison and analysis," *Journal of Mathematical Imaging and Vision*, vol. 35, no. 2, pp. 155–164, 2009.