Precise Positioning of Robots with Fusion of GNSS, INS, Odometry, LPS and Vision

Patrick Henkel,
Technische Universität München
Theresienstrasse 90
80333 München
+49-89-289-23462
patrick.henkel@tum.de

Andreas Sperl, Ulrich Mittmann, Robert Bensch and Paul Färber
ANavS GmbH - Advanced Navigation Solutions
Gotthardstrasse 40
80686 München, Germany
+49-89-89056721
{andreas.sperl, ulrich.mittmann, robert.bensch, paul.faerber}@anavs.de

Abstract— **The autonomous driving of robots is coming and requires precise and reliable positioning information with low-cost sensors for the mass market. In this paper, we propose a tightly coupled sensor fusion of multiple complementary sensors including Global Navigation Satellite System (GNSS) receivers with Real-Time Kinematics (RTK), Inertial Measurement Units (IMUs), wheel odometry, Local Positioning System (LPS) and Visual Positioning.**
The focus of this paper is on the integration of LPS and vision since the coupling of GNSS-RTK, INS and wheel odometry is already state of the art. We include the positions of the LPS anchors and the bearing vectors and distances from the robot's camera towards the patch features as state vectors in our Kalman filter, and show the achievable positioning accuracies.

TABLE OF CONTENTS

1. INTRODUCTION

In this section, we briefly introduce the complementary properties of various positioning sensors including GNSS receivers, IMUs, wheel odometry sensors, a Local Positioning Systems (LPS), and camera-based Visual Positioning. Tab. 1 lists these 5 positioning sensors and the favourable conditions for these sensors.

In this section, we also briefly introduce the ANavS Multi-Sensor RTK module, that is used in this work. The module is shown in Fig. 1 and carries multiple low-cost sensors, communication interfaces and a processor for performing the sensor fusion. The key features of the ANavS Multi-Sensor RTK module are provided in the following list.

- 1 to 3 integrated GNSS receivers (u-blox LEA M8T) for RTK positioning and attitude determination
- integrated inertial sensors (MPU 9250 from Invensense and optionally also ADIS 16460 from Analog Devices) and barometer for robust positioning
- integrated CAN-bus interface for odometry and CSI-interface for camera
- integrated LTE module for reception of RTK corrections
- integrated processor for Multi-Sensor, Multi-GNSS tightly coupled RTK positioning
- integrated USB, WiFi and Ethernet interfaces

Table 1. Comparison of complementary positioning sensors: description of conditions resulting in high performance for each individual sensor.

Sensor	Conditions enabling a high positioning accuracy
GNSS receiver	open-sky conditions with at least 4 visible satellites with continuous phase tracking
Inertial Measurement Unit (IMU)	any area for a few seconds after initialization
wheel odometry	any area with paved roads
Local Positioning System (LPS)	any area with line of sight to *at least* 3 anchors
Visual positioning	any area with clear textures, e.g. road markings and road signs, trees and houses, parked cars, etc.

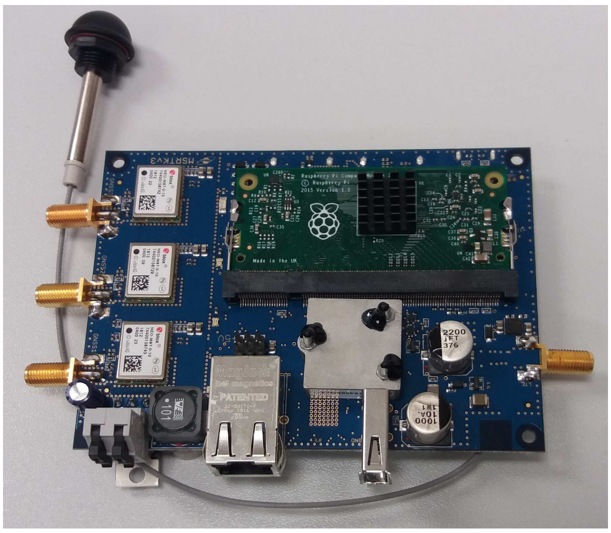

Figure 1. Multi-Sensor RTK module of ANavS: On the left side, there are 3 GNSS receivers with SMA antenna connectors. A commercial-grade IMU is in the middle. The processor for running the sensor fusion is plugged-in on the top. The SMA connector on the right side is used for the GSM/ LTE antenna. The antenna on the upper left side is a WiFi antenna that is integrated into the casing.

2. Local Positioning System

In this section, we describe the LPS and its integration into the sensor fusion similar to [3]. There are two types of LPS range measurements: the first type of range measurements refers to the range between a certain anchor (with index k) and the user/ robot (with index u), and is modeled as

$$\begin{aligned} r_u^k &= \|\vec{x}_u - \vec{x}^{\,k}\| + \Delta r^k_{\mathrm{MP}_u} + \eta^k_u \\ &= \left(\vec{e}^{\,k}_u\right)^{\mathrm{T}} \left(\vec{x}_u - \vec{x}^k\right) + \Delta r^k_{\mathrm{MP}_u} + \eta^k_u, \end{aligned} \quad (1)$$

with the following notations:

$\vec{x}_u$	user/ robot position
$\vec{x}^{\,k}$	anchor position
$\vec{e}^{\,k}_u = \frac{\vec{x}_u - \vec{x}^{\,k}}{\|\vec{x}_u - \vec{x}^{\,k}\|}$	normalized direction vector between anchor and robot
$\Delta r^k_{\mathrm{MP}_u}$	multipath error of LPS range meas.
η^k_u	noise of LPS range measurement

The second type of range measurements refers to the anchor-to-anchor measurements. The range measurement between anchors k and l is modeled similar to Eq. (1) as

$$\begin{aligned} r^{kl} &= \|\vec{x}^{\,k} - \vec{x}^{\,l}\| + \Delta r^{kl}_{\mathrm{MP}_u} + \eta^{kl}_u \\ &= \left(\vec{e}^{\,kl}\right)^{\mathrm{T}} \left(\vec{x}^{\,k} - \vec{x}^{\,l}\right) + \Delta r^{kl}_{\mathrm{MP}_u} + \eta^{kl}_u. \end{aligned} \quad (2)$$

We determine the positions of the robot and all anchors jointly in a Kalman filter. The anchor-to-robot and anchor-to-anchor measurements are stacked in a single column vector as

$$z = \left(r^1_u, \ldots, r^K_u, r^{12}, \ldots, r^{1K}, \ldots, r^{(K-1)K}\right)^{\mathrm{T}}, \quad (3)$$

which includes $K + \frac{(K-1)K}{2}$ independent measurements.

The LPS can not provide a unique solution, i.e. the positions of all anchors and the robot can be shifted by a common, arbitrary vector and be rotated by a common, arbitrary rotation matrix without affecting the range measurements. This leaves 6 degrees of freedom. We use them

- to set the coordinate center of the Local Positioning System (LPS) to the position of the first LPS anchor
- to define the x-axis of the coordinate frame, such that it points from the coordinate center towards the second LPS anchor
- to define the y-axis of the coordinate frame, such that it lies in the plane spanned by the first two anchors and the third anchor
- to define the z-axis of the coordinate frame, such that it complements a right-hand coordinate frame

Fig. 2 shows the LPS coordinate frame based on the positions of three anchors.

The $\{x, y, z\}$ coordinates of the anchor positions in the LPS coordinate frame are noted as

$$\vec{x}^{\,1} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \; \vec{x}^{\,2} = \begin{pmatrix} d^2_x \\ 0 \\ 0 \end{pmatrix}, \; \vec{x}^{\,3} = \begin{pmatrix} d^3_x \\ d^3_y \\ 0 \end{pmatrix},$$

$$\vec{x}^{\,4} = \begin{pmatrix} d^4_x \\ d^4_y \\ d^4_z \end{pmatrix}, \ldots, \vec{x}^{\,K} = \begin{pmatrix} d^K_x \\ d^K_y \\ d^K_z \end{pmatrix}, \quad (4)$$

where 6 coordinates are 0. The remaining $3(K-2)$ coordinates are unknown and have to be estimated in the Kalman filter.

The state vector comprises the unknown position coordinates of the robot and all anchors, and the velocity $\vec{v}_u$ of the robot, i.e.

$$x = \left(\vec{x}_u^{\,\mathrm{T}}, \vec{v}_u^{\,\mathrm{T}}, d^2_x, d^3_x, d^3_y, (\vec{x}^{\,4})^{\mathrm{T}}, \ldots, (\vec{x}^K)^{\mathrm{T}}\right)^{\mathrm{T}}. \quad (5)$$

The coordinates of the anchor points are assumed to be constant and the robot is assumed to move with a low and almost constant speed. Thus, the state space model is straightforward, i.e. we assume constant coordinates for the anchor points and very small process noise for the robot's acceleration.

We a use a standard Kalman filter for the LPS, and consider the following aspects:

- iterative approach required for state update due to linearization of range measurements
- certain movement required for convergence of positions of anchors and robot

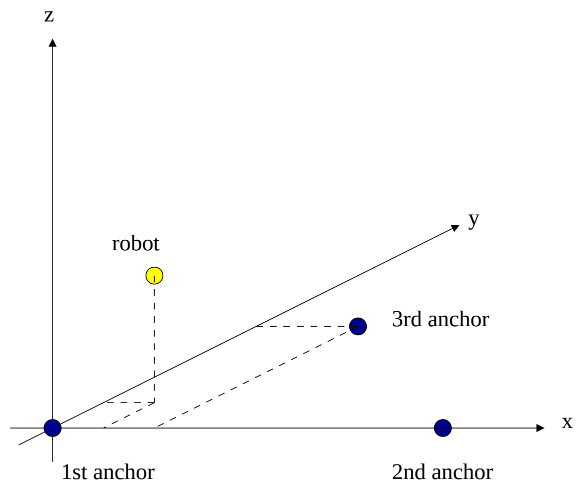

Figure 2. Local Positioning System (LPS): The right-hand coordinate frame is spanned by the locations of three anchors.

3. Integration of Local Positioning System into Sensor Fusion

In this section, we briefly describe the sensor fusion of the Local Positioning System (LPS), the IMU, and wheel odometry. We perform a tightly coupled sensor fusion as shown in Fig. 3, i.e. the raw measurements of all sensors are directly used to estimate the state vector comprising the position, velocity, acceleration, attitude angles and angular rates of the robot, the anchor positions, and the IMU and odometry biases. A standard extended Kalman filter is used for the sensor fusion as described by Brown and Hwang in [4].

4. Visual-Inertial Odometry

In this section, we describe some fundamentals for visual-inertial odometry with a monocular camera. We use the Robust Visual-Inertial Odometry (ROVIO) method of Blösch et al. and closely follow their description in [1] [2].

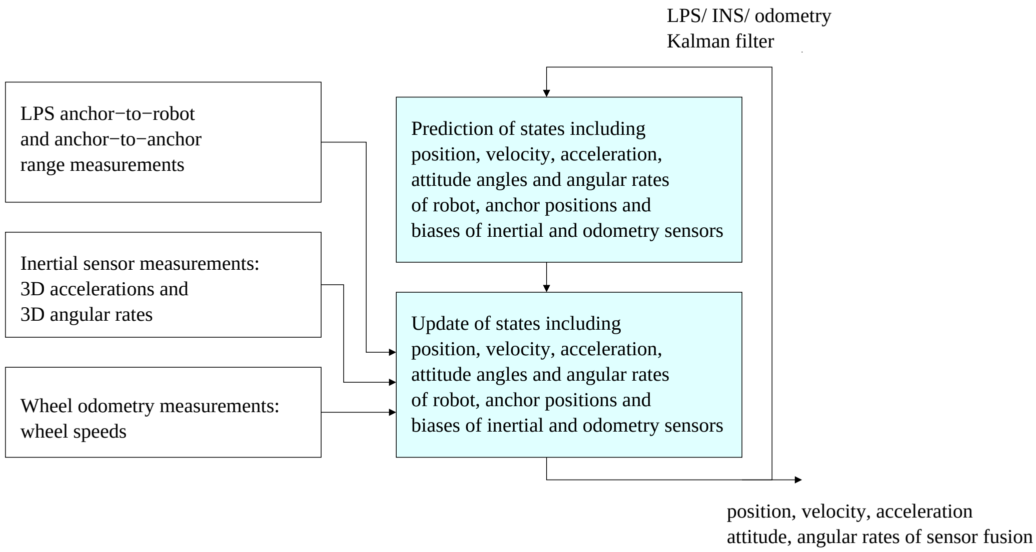

Figure 3. Sensor Fusion of LPS, INS and Odometry: A Kalman filter is used to predict and update the state vector at every measurement epoch. The state vector includes the position, velocity, acceleration, attitude angles and angular rates of the robot, the anchor positions, and the IMU and odometry biases.

Projection model and linear warping

In this subsection, we discuss the relationship between the pixel coordinates of a landmark in two subsequent image frames. The pixel coordinates $\vec{p}_n^{\,l}$ of landmark l in frame n can be expressed in terms of the camera model π with known intrinsic calibration, and the bearing vector $\vec{\mu}_n^{\,l}$ pointing from the camera to the landmark:

$$\vec{p}_n^{\,l} = \pi\left(\vec{\mu}_n^{\,l}\right). \tag{6}$$

Solving this equation for $\vec{\mu}_n^{\,l}$ yields

$$\vec{\mu}_n^{\,l} = \pi^{-1}\left(\vec{p}_n^{\,l}\right). \tag{7}$$

The bearing vector is changing with the movement of the robot/ camera and, therefore, predicted to the next camera frame with a certain process model, i.e.

$$\vec{\mu}_{n+1}^{\,l} = f(\vec{\mu}_n^{\,l}), \tag{8}$$

and then re-projected to pixel coordinates:

$$\vec{p}_{n+1}^{\,l} = \pi\left(\vec{\mu}_{n+1}^{\,l}\right). \tag{9}$$

Concatenating the projections of Eq. (7) to (9) relates the pixel coordinates of a certain landmark in two subsequent frames:

$$\vec{p}_{n+1}^{\,l} = \pi\left(f\left(\pi^{-1}\left(\vec{p}_n^{\,l}\right)\right)\right) \tag{10}$$

We linearize these projections for the Kalman filter and obtain the following linear warping matrix:

$$D = \frac{\partial \vec{p}_{n+1}^{\,l}}{\partial \vec{p}_n^{\,l}} = \frac{\partial \pi(\vec{\mu}_{n+1}^{\,l})}{\partial \vec{\mu}_{n+1}^{\,l}} \frac{\partial f(\vec{\mu}_n^{\,l})}{\partial \vec{\mu}_n^{\,l}} \frac{\partial \pi^{-1}(\vec{p}_n^{\,l})}{\partial \vec{p}_n^{\,l}}. \tag{11}$$

Photometric error

The photometric error is defined as the pixel-wise intensity difference between a patch feature (as extracted from a previous image frame) and the patch feature at the predicted location of the current image. The patch distortion due to the movement of the camera/ robot is considered by the warping matrix of Eq. (11). Blösch et al. also take changes in illumination between the different image frames into account by introducing a scaling factor a and bias b. Thus, the photometric error follows as

$$\varepsilon_{n,j}^{l}(\vec{p}^{\,l}, P_n, I_n, D) = P_n(\vec{p}_j^{\,l}) - a I_n(\vec{p}^{\,l} s_n^l + D\vec{p}_j^{\,l}) - b, \tag{12}$$

with the following notations:

P_n	intensity of patch feature at frame n
$\vec{p}_j^{\,l}$	coordinates of patch pixel of patch feature relative to center of patch feature
$\vec{p}^{\,l}$	predicted coordinates of centre of l-th patch feature relative to center of image
I_n	intensity of image at frame n
s_n^l	scaling factor accounting for downsampling
a	intensity model parameter to account for changes in illumination
b	intensity model parameter to account for changes in illumination

This photometric error is used directly as measurement to update the state vector in our tightly-coupled sensor fusion with a Kalman filter.

5. Integration of Visual-Inertial Odometry into Sensor Fusion

In this section, we describe the integration of the visual-inertial odometry into the sensor fusion. Fig. 4 shows the architecture for the sensor fusion of GNSS, INS, wheel-odometry and visual-inertial odometry. The LPS measurements are not considered in this section since both GNSS and LPS provide position information.

The visual-inertial odometry uses a Kalman filter that processes the images from a monocular camera and the measurements from an inertial sensor. Our implementation is

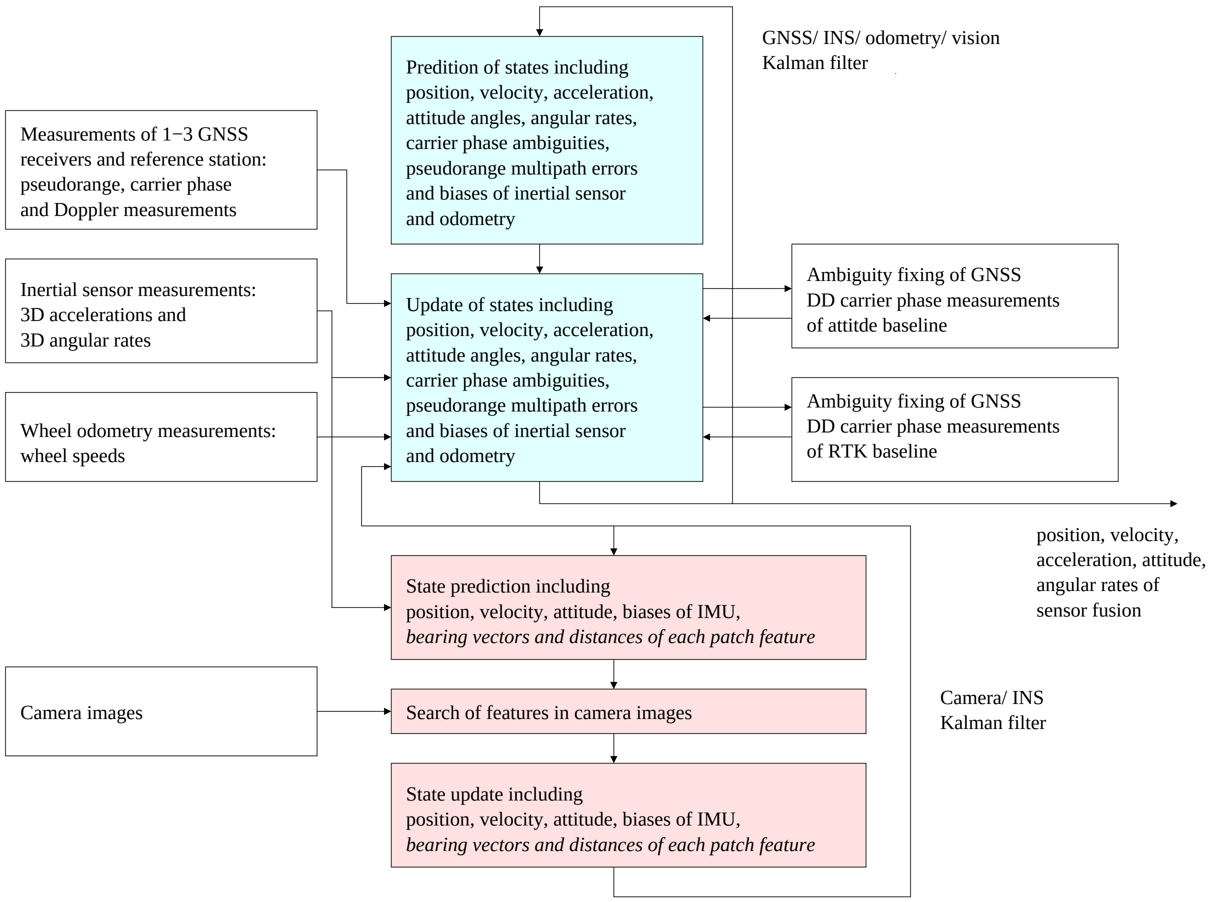

Figure 4. Architecture for Sensor Fusion of GNSS, INS, wheel-odometry and visual-inertial odometry in Kalman filter.

based on the ROVIO (RObust Visual-Inertial Odometry)-framework of Blösch et al. [1] and [2], that tracks the *bearing vector* and *distance* of *each patch feature* as state parameter besides the position, velocity, attitude and biases of the inertial sensor. The individual steps of the visual-inertial odometry are highlighted in red. The first step includes the prediction of the state parameters using inertial measurements. Subsequently, the locations of the feature patches are searched in the new camera image around the predicted locations of the feature patches. Finally, the state vector is updated based on the found feature patches.

The obtained position, velocity and attitude estimates serve as measurements for the main Kalman filter, that also uses the GNSS-, INS- and wheel-odometry measurements to update its state vector. The state vector of the main Kalman filter includes the position, velocity, acceleration, attitude angles, angular rates, carrier phase ambiguities, pseudorange multipath errors, and biases of the inertial sensor and wheel odometry. A standard Kalman filter is used for this overall sensor fusion and the respective state prediction and state update steps are highlighted in blue in Fig. 4.

6. Measurement results

In this section, we describe the measurement results. We start with the LPS/ INS/ ODO tightly coupled positioning system using the TREK 1000 from Decawave as LPS and the IMU MPU 9250 from Invensense. The performance is tested with two set-ups: a model train (without odometry) and an autonomous lawnmower (with odometry). Subsequently, we present the benefit of integrating visual-inertial odometry into the GNSS/ INS/ ODO tighly coupled RTK positioning with an autonomous lawnmower. The sensors of the ANavS Multi-Sensor RTK module and the Raspberry Pi camera are used.

Fig. 5 shows the performance of the LPS with a model train. The positions of three anchors and the robot are jointly estimated. The track of the model train is a closed loop, which enables an analysis of the repeatability of the position solution. The enlarged view provides two insights: First, the point cloud at (3.3 m, 1.95 m) refers to the initial static position, and has a standard deviation of a few centimeters only. Second, the multiple parallel lines refer to different rounds of the model train and indicate a consistent position solution.

Fig. 6 includes a comparison between the tightly coupled LPS/ INS and the tightly Multi-GNSS/ INS RTK positioning. The closed-loop track is installed at the roof-top of ANavS with open-sky conditions, i.e. both satellite signals and anchor signals are received without obstructions. Both systems are coupled with an IMU and provide consistent solutions with an uncertainty of less than 10 cm for most epochs. The systematic offsets between both positioning solutions around the lower left part and also at the rightmost part of the track are LPS errors that occur if the angle between an LPS antenna plane and the signal path is very small.

Now, the performance of the LPS/ INS/ ODO-tightly coupled

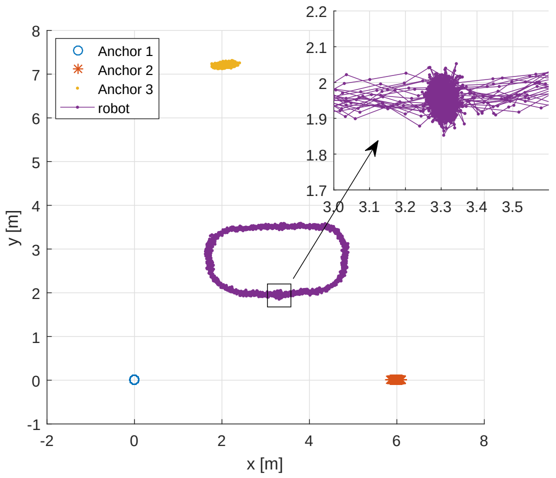

Figure 5. Performance analysis of Local Positioning System (LPS) with a model train.

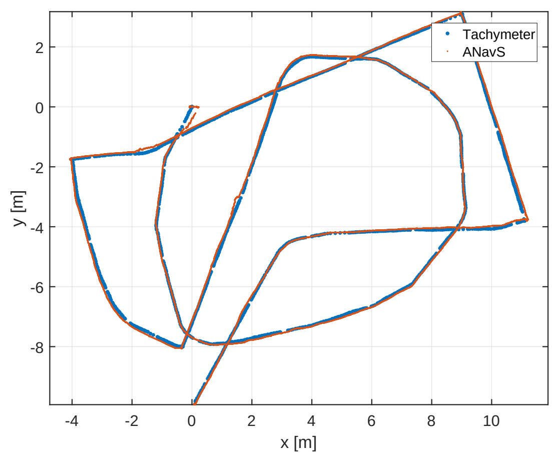

Figure 7. Comparison of horizontal position estimates between the ANavS Multi-Sensor Fusion of Local Positioning System, wheel odometry and inertial sensor with the tachymeter-based reference solution.

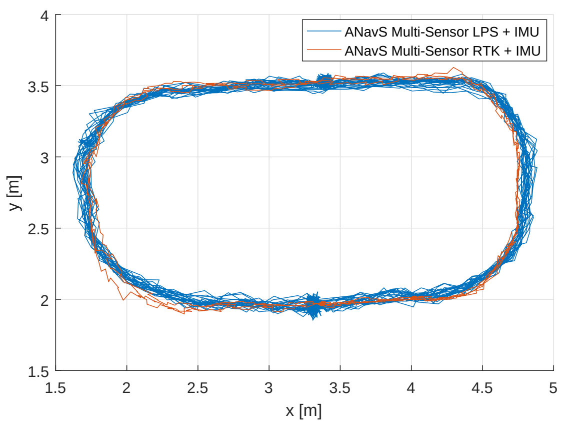

Figure 6. Comparison between ANavS tightly coupled Multi-GNSS/ INS RTK positioning and ANavS tightly coupled LPS/ INS positioning with model train.

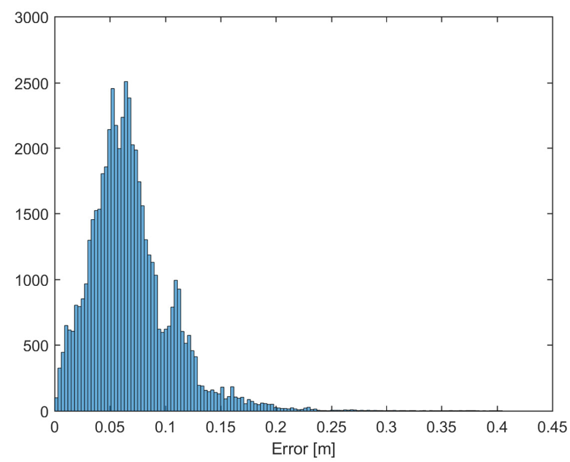

Figure 8. Histogram of horizontal position deviation between ANavS Multi-Sensor Fusion of Local Positioning System, wheel odometry and inertial sensor and tachymeter-based reference.

system is analyzed in a typical outdoor environment with surrounding trees. Fig. 7 includes a comparison of horizontal position estimates between the ANavS Multi-Sensor Fusion of LPS, wheel odometry and inertial sensor with the reference solution of a tachymeter from Leica: In principle, both solutions are well aligned for almost all epochs. A slight offset of the LPS/Odo/IMU solution can be observed near the start at (0,0) since the Kalman filter needs some time to converge. The tachymeter solution has occasional gaps due to the lack of a line of sight between tachymeter and robot. Moreover, a temporary reduction of accuracy can be observed for the LPS/ODO/IMU solution in areas where the LPS signals from at least one anchor point were shadowed or blocked, e.g. around (1.5 m, -3.0 m).

Fig. 8 includes a histogram of the horizontal position deviation between the ANavS Multi-Sensor Fusion of Local Positioning System, wheel odometry and inertial sensor and the tachymeter-based reference: The position offset remains below 15 cm for 95 % of the epochs.

Fig. 9 shows a comparison of the positioning trajectories obtained with and without visual odometry. The trajectory starts with a rectangular, repetitive pattern at an open field. The initial convergence of the RTK float solution is also shown. The position estimates with and without visual positioning are well-aligned. This indicates the correctness of positioning with and without visual odometry. After the rectangular pattern, the robot drove towards trees and bushes (upper part of trajectory) to test the positioning performance in more challenging conditions. We can observe a certain deviation between the position trajectories with and without visual odometry. The benefit of the visual odometry becomes apparent at the RTK refixing after passing the sections with trees and bushes: The position correction is only 20 cm with visual odometry compared to 30 cm without visual odometry. The diagram also shows three highlighted locations. The respective camera images are provided in Fig. 10. The illumination slightly varies between the images. The multilevel patch features are determined by ROVIO, and represented by squares. Green color denotes successfully tracked patch features and red color denotes rejected patches, whereas a feature patch is rejected if the innovation residuals at more than 2 of the 4 checked surrounding locations (labeled by dots) are not higher than for the predicted location.

The final (i.e. after iterative convergence) location of each landmark is shown with a small red dot surrounded by 4 green or red dots. The surrounding locations are checked for higher innovation residuals to keep (green) or reject (red) the patch features. The estimated uncertainty of each landmark location is shown by yellow ellipses. The largest patch feature uncertainty of the first image has the patch feature in the upper right part, where the image is very dark. We can observe that almost all patch features are in green, which indicates that grass patches can be tracked well.

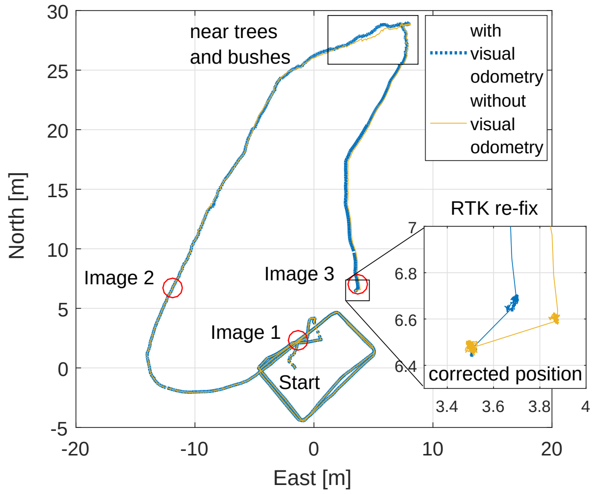

Figure 9. Comparison of Multi-GNSS/ wheel odometry/ IMU tightly coupled RTK positioning with and without integrated visual odometry.

7. Conclusion

The autonomous driving of robots requires a precise and reliable positioning. In this paper, we analyzed the tightly-couled sensor fusion of GNSS-RTK, INS, odometry, Local Positioning System (LPS) and visual positioning. The focus was put on the LPS and visual positioning, and their integration into the sensor fusion. The paper provided a quantitative performance analysis with real measurements, and showed that centimeter-level positioning accuracy is feasible with low-cost sensors.

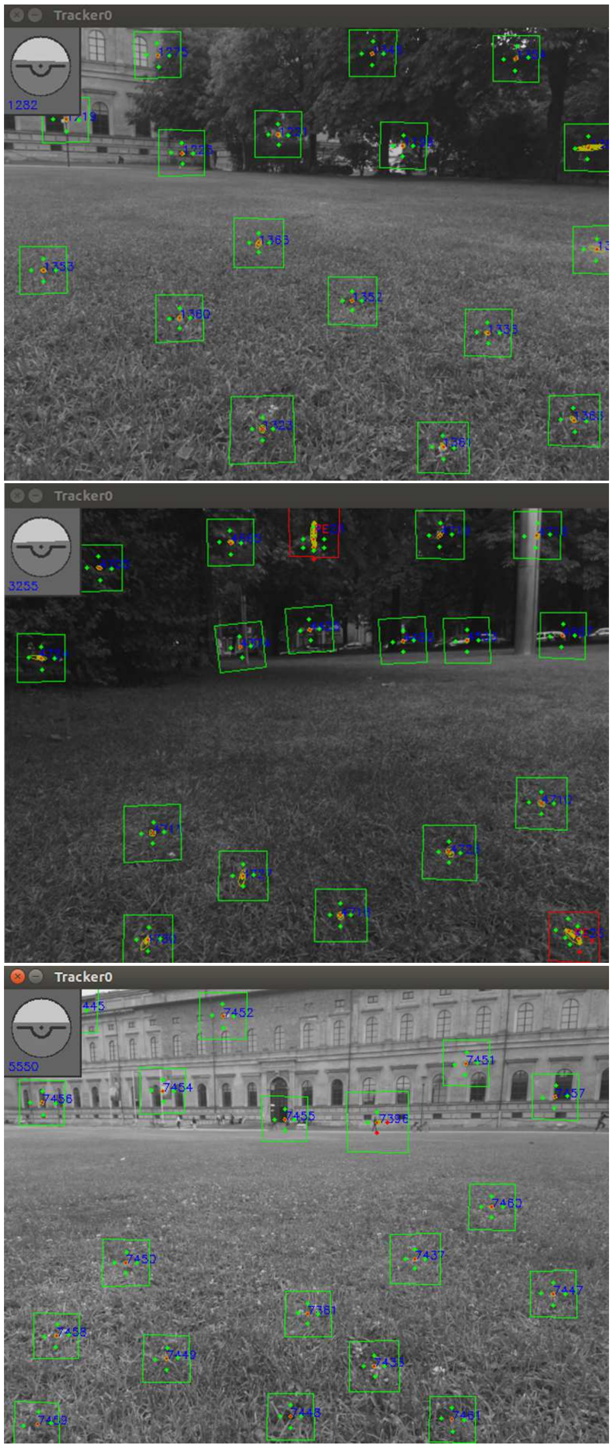

Figure 10. Camera images with ~ 20 patch features on the grass at Pinakothek, Munich, with trees in the background.

References

[1] M. Blösch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, *IEKF-based Visual-Inertial Odometry using direct Photometric Feedback*, Intern. Journal of Robotics Research, vol. 36, issue 10, pp. 1053 – 1072, Sep. 2017.

[2] M. Blösch, S. Omari, M. Hutter, and R. Siegwart, *Robust Visual Inertial Odometry Using a Direct EKF-Based Approach*, Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, Sep. 2015.

[3] P. Henkel, A. Sperl, U. Mittmann, R. Bensch, P. Färber and C. Günther, *Precise Positioning of Robots with Fusion of GNSS, INS, Odometry, Barometer, Local Positioning System and Visual Localization*, Proc. of ION GNSS+, pp. 3078 – 3087, Sep. 2018.

[4] R.G. Brown and P.Y.C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering with Matlab Exercises*, 4th edition, Wiley, 400 pages, Feb. 2012.