Localization Method for Autonomous Vehicles with Sensor Fusion Using Extended and Unscented Kalman Filters

2021-01-5089 Published 15 Sep 2021

Localization Method for Autonomous Vehicles with Sensor Fusion Using Extended and Unscented Kalman Filters

Stefano Feraco, Stefano Favelli, Andrea Tonoli, Angelo Bonfitto, and Nicola Amati Politecnico di Torino

Citation: Feraco, S., Favelli, S., Tonoli, A., Bonfitto, A. et al., "Localization Method for Autonomous Vehicles with Sensor Fusion Using Extended and Unscented Kalman Filters," SAE Technical Paper 2021-01-5089, 2021, doi:10.4271/2021-01-5089.

Abstract

This paper presents the design and experimental validation of a localization method for autonomous driving. The investigated method proposes and compares the application of the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF) to the sensor fusion of onboard data streaming from a Global Positioning System (GPS) sensor and an Inertial Navigation System (INS). In the paper, the design of the hardware layout and the proposed software architecture is presented. The method is experimentally validated in real time by using a properly instrumented all-wheel-drive electric racing vehicle and a compact Sport Utility Vehicle (SUV). The proposed algorithm is deployed on a high-performance computing platform with an embedded Graphical Processing Unit that is mounted on board the considered vehicles. The reported experimental results include the outcomes of the localization algorithm at submeter accuracy and the estimated vehicle's states for the retained single-track vehicle model that is exploited for further control strategies. The experimental results show a substantial equivalence of the application of the two filters. Nevertheless, the UKF-based method is characterized by a significantly lower estimation variance in the localization task, thus providing more robust results.

Keywords

Localization, Sensor fusion, Vehicle dynamics, Autonomous vehicles, Extended Kalman filter, Unscented Kalman filter

1. Introduction

Autonomous vehicles are experiencing an increasing interest worldwide during the last years, which has constantly motivated huge research efforts to continuously address design challenges related to safety and performances of the next generation of automated cars [1, 2]. Many Advanced Driving Assistance Systems are already present in the majority of vehicles of the recent mass production, thanks to recent developments in engineering in the real-time assessment of vehicle dynamics and passengers comfort, as well as in the development of robust safety systems [3, 4, 5]. However, fully autonomous commercial vehicles are still far from being the common means of transport, although they are retained to change the whole mobility panorama in the next decades, thanks to the contribution of Artificial Intelligence, as stated in [6] and [7].

In this framework, localization represents one of the most relevant issues for an autonomous vehicle that aims to navigate without any human support, thus being a key enabler for the development of any technology devoted to self-driving cars [8, 9]. Considering the self-driving vehicle as a robotic system, the localization problem arises with the vehicle's motion in an unknown environment [10]. This issue is even more critical in autonomous driving when the effectiveness of path planning or vehicle's dynamics control algorithms is tested in the limited conditions presented by challenging driving scenarios, such as racetracks or urban areas. Strictly linked to localization, the precise real-time vehicle's state estimation is another crucial topic to be addressed by any autonomous driving system because it directly impacts the performance of every control algorithm devoted to driverless systems. An accurate and robust estimation of the vehicle's pose (position and orientation) and states is the goal of the method presented in this paper, which has been motivated by the need for an accurate and robust real-time positioning system to deploy an effective local path planner [11, 12].

The investigated method is addressed to a properly instrumented all-wheel-drive electric racing vehicle and a

commercial compact Sport Utility Vehicle (SUV). In the retained vehicles, the choice of a local path planner for autonomous driving is given by the unavailability of an a priori knowledge of the driving environments [13, 14]. The considered racing vehicle aims to participate in Formula Student Driverless Competition. In this paper, the racing vehicle is retained as a development platform for the proposed algorithm that is validated on a commercial compact SUV to prove the responsiveness, accuracy, and scalability of the proposed algorithms. The problem of finding the best trade-off between those features is a common design challenge and has been tackled with different approaches, as investigated by [15, 16, 17, 18]. Nevertheless, all the aforementioned research works solve the localization problem together with the mapping problem by means of Simultaneous Localization and Mapping (SLAM) algorithms, embedding different state estimation techniques and hardware setups. In general, SLAM algorithms are a well-documented solution in literature [19] and have proven to be a robust and reliable source of state estimation also for racing applications. However, the main drawback of the SLAM approach when dealing with unknown racetracks or urban maps is the need for a software stack that is able to run a global mapping routine during a first low-pace lap to achieve a good level of details of the map. Then the vehicle can navigate for the remaining laps in a known environment in a pure autonomous roaming condition. This process is poorly applicable to commercial vehicles in urban areas. Moreover, it can sacrifice the dynamic performance of a racing vehicle during the initial lap, while building the environment for a global path planner. Conversely, the proposed localization method aims to provide an accurate real-time localization to enable a local path planner that emulates the behavior of a human driver in ensuring the needed short-term and high-end performance, even in the case of facing the track for the first time, i.e., the common situation encountered by a commercial vehicle in urban areas. Therefore, the localization algorithm has a pivotal role in assuring the responsiveness of the entire autonomous system because it enables the ability of the vehicle to be self-aware of its position on the track and its dynamic conditions at any time with the highest level of accuracy.

In this research work, the performance of two renowned filters for nonlinear models, namely, the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF), are compared in estimating the position and orientation of the retained vehicles. Furthermore, the vehicle's states are also estimated for the real-time assessment of the retained single-track vehicle model that is exploited for further control strategies. The two investigated filters fuse the measurements taken by a Global Positioning System (GPS) sensor and an Inertial Navigation System (INS) mounted on board the retained vehicles.

Although the application of the EKF and UKF with sensor fusion of GPS and INS sensors have been individually validated in the recent literature for the assessment of vehicle localization and navigation algorithms [20, 21, 22], an effective comparison of the two filters and the consequent study of the potential application to commercial vehicles is still missing. To the best of the authors' knowledge, such studies are limited to indoor autonomous robot applications [23] and autonomous underwater vehicles [26]. This paper aims to fill this gap in the literature by experimentally testing the proposed algorithms on an autonomous racing prototype and by validating them on a properly instrumented commercial compact SUV. Different driving maneuvers are studied for both the considered vehicles.

Furthermore, a proper localization algorithm must guarantee a low computational effort for the devoted computational platform, while assessing the estimation of the vehicle's position and orientation and the estimation of the vehicle dynamics states for the local path planner. In this paper, it is also shown how to partially decentralize the sensor fusion routine at the hardware level using an industrial-grade navigation sensor, combined with the aforementioned filtering techniques deployed on board for redundancy. The proposed decentralized approach is novel at the hardware level, and it has been validated experimentally.

Moreover, the choice of exploiting the robot localization package capabilities for the main software architecture has constrained the kind of filters considered in this paper. The package embeds only two different implementations of nonlinear state estimators for a three-dimensional motion assessment of mobile robots, namely, the EKF and UKF algorithms, for their similar configuration and tuning procedures. Both the EKF and UKF belong to the class of Gaussian filtering techniques, which guarantee a parametric approach to the measurement noise description, limiting the overall computational burden of the package, which is designed to run on low-cost computing platforms. Despite being a general-purpose software component, it can be adapted to the needs of a specific application by properly tuning the process noise and the initial estimate covariance matrices, respectively. In the considered case, the package has been initialized to privilege computational efficiency over final estimate accuracy, finding the best trade-off among the requirements.

Furthermore, Particle Filters and Graph-based methods have proven to guarantee a more stable solution to the state estimation problem with respect to Gaussian techniques because of the lower level of approximation introduced in measurement noise. Nevertheless, the computational complexity rises with the adoption of nonparametric noise descriptions, especially for nonlinear systems, as written in [23], [24], and [25]. Thus, considering the requirements of the retained vehicles and the adoption of a general-purposed software component, only EKF and UKF have been designed because of the best trade-off reachable between the computation effort required and the estimates' accuracy.

The advantage of this approach is to increase the robustness of the localization pipeline that can be reached with the hardware components present on board. Eventually, localization robustness can be further refined and improved by adding measurements from other sensors if available, such as vision-based or odometry sensors. However, this procedure can cause a reasonable degradation of responsiveness and an increased level of complexity, thus being beyond the scope of the paper. The performance of the investigated algorithms has been evaluated in terms of analysis of the covariance matrices of both the filters during the performed experimental tests. Moreover, results have been quantified in terms of position error.

In detail, the main contributions of this paper are (1) The comparison and performance assessment of EKF and UKF for localization purposes and the potential application to commercial vehicles. To this end, different driving maneuvers are studied for both the considered vehicles; (2) The partial decentralization of the sensor fusion routine at hardware level using an industrial-grade navigation sensor, which is combined with the mentioned filtering techniques. The proposed decentralized approach is novel at the hardware level, and it has been validated experimentally.

This paper is structured as follows: Section 2 presents the experimental vehicle setup and modelling for each of the two considered vehicles, as well as the outline of both the hardware and software architectures chosen for the localization method. Moreover, the design of both the investigated EKF and UKF is presented. Then, the experimental validation process is discussed in Section 3 using two different datasets acquired on board the considered vehicles, representing the different driving scenarios. Results include the estimated vehicle's states during the experimental validation, such as the vehicle's position and orientation, roll, pitch, yaw (RPY) angles, velocity, and acceleration measurements.

2. Method

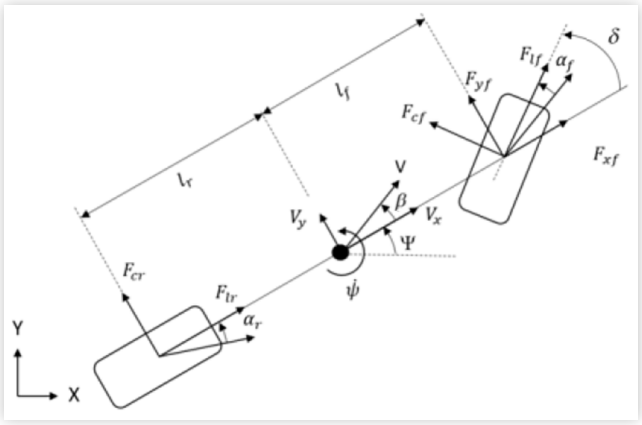
In this section, the modelling approach for both the considered vehicles is illustrated, along with the hardware setup and software architecture. The main parameters of the two retained vehicles are also presented first. Then, the design of the investigated EKF and UKF for localization purposes and vehicle states estimation is discussed.

2.1. Vehicle Modelling

The proposed localization algorithm is tested and validated on two different vehicles: the first one is an all-wheel-drive electric racing vehicle used as a prototype in this research and the latter is a commercial passenger car (compact SUV class). For both the vehicles, the longitudinal and lateral dynamics are modelled by means of a single-track linear three degrees-of-freedom (3-DoF) model that is used for further autonomous vehicles control strategies as described in [27]. This simplified model is proven to be effective among the different mathematical formulations, which are present in the literature [28, 29, 30]. The relevant states of the vehicle model are represented in Figure 1. The considered formulation features a rigid two-axle vehicle model and accounts for the linear motion in the xy-plane and the rotation about the z-axis.

In Figure 1, two different reference frames are considered: the inertial reference frame (XY) and the vehicle reference frame (xy). The considered DoF are the lateral and longitudinal positions of the vehicle and the yaw angle. A canonical state feedback formalism is used for the modelling definition, as in [5] and [31]. In the state-space model of Equation 1, the lower scripts $(\cdot)_f$ and $(\cdot)_r$ denote the variable $(\cdot)$ at the front and rear wheels, respectively. The state-space model is obtained considering a first-order transfer function with time constant

FIGURE 1 Single-track linear 3-DoF vehicle model.



$\tau = 0.5$ s for the longitudinal dynamics and the Euler equation for the lateral and the yaw motion:

$$\begin{bmatrix} \ddot{V}_x \\ \dot{V}_x \\ V_y \\ \dot{V}_y \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau} & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\frac{2C_{af}+2C_{ar}}{mV_x} & 0 & -V_x - \frac{2C_{af}l_f - 2C_{ar}l_r}{mV_x} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\frac{2C_{af}l_f - 2C_{ar}l_r}{I_z V_x} & 0 & -\frac{2C_{af}l_f^2 + 2C_{ar}l_r^2}{I_z V_x} \end{bmatrix} \begin{bmatrix} \dot{V}_x \\ V_x \\ y \\ V_y \\ \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} \frac{1}{\tau} & 0 \\ 0 & 0 \\ 0 & \frac{2C_{af}}{m} \\ 0 & \frac{2l_f C_{af}}{I_z} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{V}_x \\ \delta \end{bmatrix} \quad \text{Eq. (1)}$$

where x and y are the positions along the x-axis and y-axis, respectively; V_x, V_y are the longitudinal and the lateral vehicle velocities, respectively; ψ is the yaw angle; and δ is the front wheel steering angle.

The equations of motion for the considered vehicle model are developed in the reference frame xy fixed to the Center of Mass (CoM) of the vehicle:

$$ma_x = mV_y\dot{\psi} + F_{xf,w}\cos\delta + F_{yf,w}\sin\delta + F_{xr,w} \quad \text{Eq. (2)}$$

$$ma_y = -mV_x\dot{\psi} + F_{yf,w}\cos\delta + F_{xf,w}\sin\delta + F_{yr,w} \quad \text{Eq. (3)}$$

$$I_{zz}\ddot{\psi} = a\left(F_{yf,w}cos\delta + F_{xf,w}\sin\delta\right) - bF_{yr,w} \quad \text{Eq. (4)}$$

$F_{yf,w}$ and $F_{yr,w}$ are the lateral tires forces applied to front and rear wheels, respectively.

The relation between the inertial reference frame XY and the vehicle-fixed reference frame xy is then described by the following equations:

$$X = x\cos\psi - y\sin\psi \quad \text{Eq. (5)}$$

$$Y = x\sin\psi + y\cos\psi \quad \text{Eq. (6)}$$

$$\Psi = \psi \quad \text{Eq. (7)}$$

The lateral tire forces at the front and rear wheels are considered perpendicular to the rolling direction of the tire and proportional to the lateral slip angle, α_f and α_r. Considering the assumption of small slip angles, the lateral tire forces are modelled as

$$F_{yf,w} = 2C_{\alpha f}\alpha_f \quad \text{Eq. (8)}$$

$$F_{yr,w} = 2C_{\alpha r}\alpha_r \quad \text{Eq. (9)}$$

where $C_{\alpha f}$ and $C_{\alpha r}$ are the tire stiffness and α_f and α_r are the tire slip angles of the front and rear tires, respectively. Therefore, the tire slip angles are defined as follows:

$$\alpha_f = \text{atan}\left(\frac{V_y + \dot{\psi} a}{V_x}\right) - \delta \quad \text{Eq. (10)}$$

$$\alpha_r = \text{atan}\left(\frac{V_y - \dot{\psi} b}{V_x}\right) \quad \text{Eq. (11)}$$

where a and b are the distance from the *CoM* and the front and rear axles, respectively.

The electric racing vehicle is properly instrumented for driving autonomously in the racetrack and features an integral carbon fiber chassis with honeycomb panels, double-wishbone push-rod suspensions, an on-wheel planetary transmission system, and a custom aerodynamic package. The vehicle can reach a maximum speed of 120 km/h with longitudinal acceleration peaks up to 1.6 g. The main vehicle's parameters are listed in Table 1.

The commercial passenger car (compact SUV class) has been used in parallel to the racing vehicle for the experimental validation of the proposed localization technique. The main parameters of the retained commercial vehicle are listed in Table 2. The parameters of the retained compact SUV have been identified from a time-frequency analysis in a prior research work by the authors [5].

TABLE 1 Main parameters of the retained electric racing vehicle (driver not included).

Parameter	Symbol	Value	Unit
Wheelbase	b	1525	mm
Overall length	l	2654	mm
Track width	t	1400	mm
Maximum height	h	1225	mm
Ground clearance	h_g	60	mm
Tire outer diameter	d	13	inch
Mass	m	185	kg
Moment of inertia about the z-axis	I_{zz}	95.81	N · m^2
Distance from *CoM* to front/rear axle	$[l_f; l_r]$	[839; 686]	mm
Front/rear cornering stiffness	$[C_{\alpha f}, C_{\alpha r}]$	[44222; 44222]	N/rad
Vehicle top speed	v_{max}	120	km/h

TABLE 2 Main parameters of the retained commercial vehicle (driver not included).

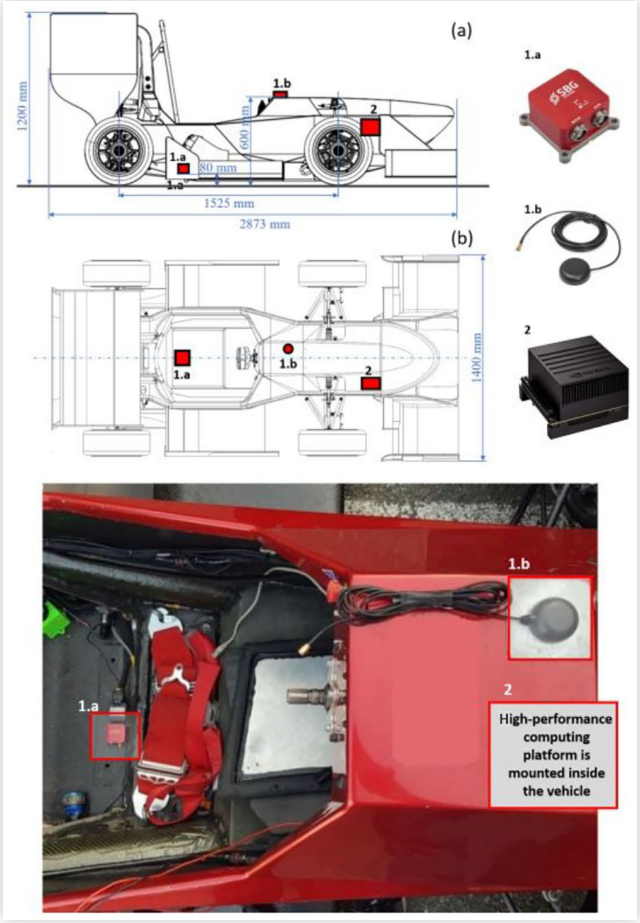
Parameter	Symbol	Value	Unit
Wheelbase	b	2570	mm
Overall length	l	4236	mm
Track width	t	1805	mm
Maximum height	h	1697	mm
Ground clearance	h_g	170	mm
Tire outer diameter	d	28.1	inch
Mass	m	1270	kg
Moment of inertia about the z-axis	I_{zz}	1550	N · m^2
Distance from *CoM* to front/rear axle	$[l_f; l_r]$	[1020; 1550]	mm
Front/rear cornering stiffness	$[C_{\alpha f}, C_{\alpha r}]$	[65765; 49517]	N/rad
Vehicle top speed	v_{max}	185	km/h

2.2. Hardware Layout and Software Architecture

Local path planning algorithms for assisted and autonomous driving require precise and real-time assessment of the main vehicle's states and position. This requirement can be met with different hardware layouts. In this research, the retained solution exploits an INS. INS are self-contained, non-radiating, dead-reckoning navigation systems that provide dynamic information through direct measurements and, if integrated with absolute location-sensing mechanisms (such as GNSS receivers), can provide accurate information about the vehicle's position at the centimeter level [32].

The retained INS sensor is the SBG Systems Ellipse-N, which embeds a 64-bit microprocessor that is able to process sensors fusion algorithms in real time. This sensor is small sized and features an integrated dual-band antenna GNSS receiver. It provides orientation angles and accurate GNSS position, thus being most suited for dynamic environments and harsh GNSS conditions. Nevertheless, it can also operate in lower dynamics applications with a magnetic heading. The sensor is fully compatible with the computational platform used on board the considered vehicles since it supports both serial communication at a high baud rate with a Linux machine running Robotic Operating System (ROS) and CAN bus communication with any real-time automotive Electronic Control Unit. The retained onboard computing platform is an NVIDIA Jetson AGX Xavier featuring Linux Ubuntu 18.04 and ROS Melodic, which has been chosen for its high computational power. Moreover, the retained INS sensor can also reduce the computational effort required by the embedded onboard computer because it integrates inertial measurement unit (IMU) and GNSS receivers with a 64-bit microprocessor capable of real-time signal pre-processing and data fusion. Specifically, a first stage of sampling and calibration of the IMU via a coning and sculling algorithm is performed at a frequency of 1 kHz by the sensor itself. The IMU inputs are then processed by an embedded data fusion routine, which integrates global navigation satellite system (GNSS) and barometer data by means of an embedded EKF with an output frequency of

FIGURE 2 Vehicle layout with hardware positions: (a) side view; (b) top view; (c) actual vehicle and hardware. (1.a) INS sensor; (1.b) GPS antenna; (2) high-performance computing platform.



200 Hz. The embedded EKF can be properly tuned on the specific application and performs a first pre-processing stage at the sensor level [33]. The hardware layout and the location of the hardware in the retained racing vehicle are illustrated in Figure 2. The computing platform is connected to a devoted rechargeable lithium battery with a proper custom wiring system, as well as the INS sensor is connected to a GPS antenna. An equivalent layout is used on the commercial vehicle but is not reported in the paper for confidentiality issues. The employment of other sensors has been also evaluated in the design phase, such as visual odometry from stereocameras or wheels odometers. Nevertheless, the purpose of having a lean hardware architecture has motivated the choice to reduce the number of the considered sensors.

TABLE 4 Accuracy performance of the INS complete module.

Parameters	Performance
Roll and pitch	0.1° SP 0.05° RTK 0.03° PPK
Heading	0.2° Dual antenna 2 m, 0.2° Single antenna with dynamics 0.1° PPK
Velocity	0.03 m/s
Navigation	1.2 m single point 1 m SBAS 1 cm RTK/PPK + 1 ppm
Heave	5 cm or 5%—valid for marine version
Heave period	Up to 15 s—automatically adjusts to the wave period

The main parameters of the retained inertial sensing units and GNSS module are reported in Table 3. Table 4 illustrates the accuracy performance of the INS complete module.

The proposed localization pipeline is fully developed by means of properly designed Python and C++ nodes in ROS. The main nodes composing the localization stack are

- ***sbg_driver***

This first node is an SBG Systems proprietary C++ node, which publishes the filtered sensor data at a custom rate that is decided for each signal on the basis of the required robustness. The standard output rate for the Ellipse-N sensor is 200 Hz, which is the one actually used in the proposed system. This node publishes proprietary ROS messages, which are used as raw data for post-processing by means of a dedicated ROS topic called */imu*, as represented in Figure 2. The standard ROS messages include the following topics: */imu/data* (sensor_msgs/Imu), */imu/velocity*,

TABLE 3 Main parameters of the retained inertial sensing units and GNSS module.

	Accelerometer	Gyroscope	Magnetometer	GNSS module	
Gain stability	1000 ppm	500 ppm	<0.5%	**Features**	SBAS, RTK, RAW
Nonlinearity	1500 ppm	50 ppm	<0.1% FS	**Signals**	GPS: L1C/A, L2C GLONASS: L1OF, L2OF GALILEO: E1, E5b, BEIDOU: B1/B2
Bias stability	±5 mg	±0.2°/s	±1 mGauss	**Update rate**	5 Hz
Random walk noise density	57 μg/$\sqrt{Hz}$	0.15°/$\sqrt{hr}$	3 mGauss	**Start time (cold and hot start)**	<24 s and <2 s
Bias in-run instability	14 μg	7°/h	1.5 mGauss		
Vibration rectification error	50 μg/g^2 RMS	1°/h/g^2 RMS	—		
Alignment error	<0.05°	<0.05°	<0.1°		
Bandwidth	390 Hz	133 Hz	22 Hz		

(geometry_msgs/TwistStamped), */imu/pos_ecef* (geometry_msgs/PointStamped), and */imu/nav_sat_fix* (sensor_msgs/NavSatFix). All the messages refer to the main reference frame of the IMU and are reported in the Earth-Centered, Earth-Fixed (ECEF) world frame, without any transform between frames published directly by the node.

- ***odometry_publisher***

A custom Python-based ROS node oversees the filling of a standard ROS message called */odometry*, which is part of the navigation messages library */nav_msgs/Odometry*. The message is composed of two geometry messages: */pose*, which refers to the position and orientation of the robot in its frame with respect to a parent frame, and */twist*, which contains the information about linear and angular velocities. The resulting odometry message is published in a ROS topic called */odometry* and reports the position, orientation, and speed of the robot in the East-North-Up (ENU) frame, which is the standard frame in ROS frames conventions.

- ***robot_localization***

The last node of the proposed localization pipeline exploits the *robot_localization* package [34], i.e., a standard ROS C++ software package. This package is properly adapted to the needs of the application in order to run the two different Kalman Filter instances improving the quality of the */odometry* topic and publishing the transforms between map, odometry, and base frames of the system. The *robot_localization* package embeds both an implementation of an EKF and a UKF, which are used to fuse the data coming from the INS solution with the raw data of the GNSS.

The software architecture of the proposed localization method is represented in Figure 3, illustrating the main hardware components, the retained operating system, and the software components.. The INS sensor routine embeds the internal EKF with the proper prediction and correction tasks, retrieving data from the IMU and GPS. The resulting output from the sensor is used for implementing the investigated EKF and UKF via a robot_localization package in the ROS environment. Eventually, all the considered ROS topics and messages are listed in Table 5.

FIGURE 3 Block scheme of the proposed software architecture. Hardware components are defined by a blue box. The operating system is indicated by a dashed box. The software components are indicated via gray boxes.

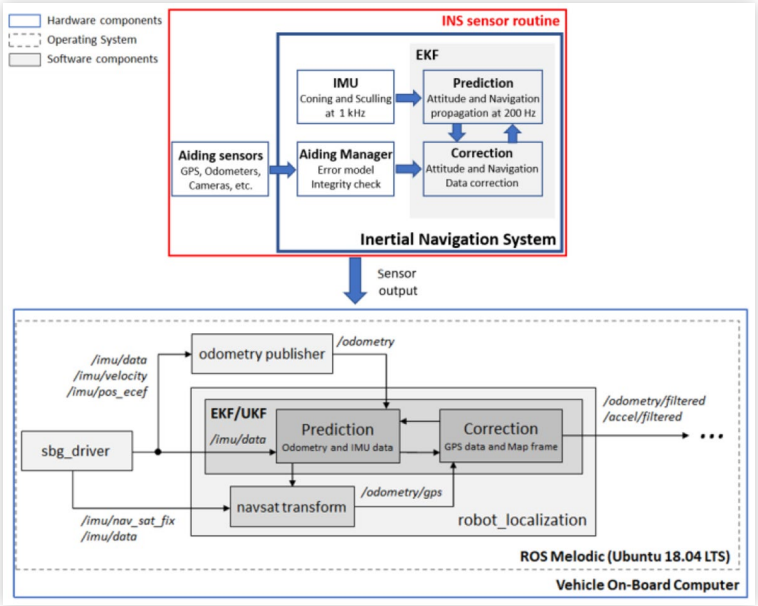


TABLE 5 Retained ROS topics and ROS message type.

ROS topic	ROS message type	Message content
/imu/data	sensor_msgs/Imu	IMU measurements from INS data (orientation and acceleration)
/imu/nav_sat_fix	sensor_msgs/NavSatFix	Raw GPS data from GNSS receiver
/imu/velocity	geometry_msgs/TwistStamped	Linear and angular twist from INS data
/imu/pos_ecef	geometry_msgs/PointStamped	ECEF position from INS data
/odometry	nav_msgs/Odometry	ECEF pose and orientation
/odometry/gps	nav_msgs/Odometry	GPS pose and orientation
/odometry/filtered	nav_msgs/Odometry	Final pose and orientation values filtered by EKF/UKF
/accel/filtered	geometry_msgs/AccelWithCovarianceStamped	Final acceleration values filtered by EKF/UKF

2.3. Vehicle Localization and States Estimation

The vehicle localization and states estimation are the main tasks of the proposed method where a pivotal role is played by sensor fusion algorithms allowing to solve the problem of estimating the states of a stochastic system from noisy measurements. As a matter of fact, real sensors observations are prone to distortions and fluctuations due to measurement noise, linked to the hardware used, and to process noise, related to inaccuracies of the retained models. Furthermore, single sensors may not cover the whole range of the physical properties under study, and usually, sensors work in an unpredictable environment and with limited computational power, which raise the uncertainty of the observed states of the process [35]. Therefore, sensor fusion is the most common way to overcome the addressed problems by combining data from multiple sensors to achieve improved accuracies and more specific inferences than what could be achieved by the usage of a single sensor. Filtering techniques are then used to estimate the state of a process directly from raw sensor measurements. As a consequence of the system modelling presented in Subsection 2.1, a nonlinear approach to the filtering problem has been considered in the paper, thus estimating the state of a nonlinear stochastic system.

For a generic discrete-time nonlinear system dealing with noisy measurements, the presented framework is represented by the following equations:

$$x_t = g\left(u_t, x_{t-1}\right) + \varepsilon_t \qquad \text{Eq. (12)}$$

$$z_t = h\left(x_t\right) + \delta_t \qquad \text{Eq. (13)}$$

where x_t is the state to be estimated, u_t is the control input, z_t is the measured output, and ε_t and δ_t are respectively the process and measurements noises. Moreover, an initial state estimate $\hat{x}_{0|0}$ and the initial estimation error covariance matrix $\Sigma_{0|0}$ are given, and the initial estimate is assumed to be uncorrelated with both ε_t and δ_t. Even under the assumption of Gaussian noises affecting the system, the filtering problem rises with complexity when dealing with nonlinearities because nonlinear transformations of Gaussian variables, such as g and h in Equations 12 and 13, do not return Gaussian variables as a result.

Among the most common state estimators (or filters) present in literature, the EKF and the UKF are considered in this paper. Both the retained filters are Bayes filters, which are also referred to as Gaussian Filters, because they deal with Gaussian Random Variables (GRV). Those filters are recursive state estimators, which are widely used in literature for their generally good performance and ease of implementation [36]. The proposed filtering methods are designed for the estimation of 15 states of the vehicle, namely, linear positions (x, y, z), Euler angles (φ, θ, ψ), linear speeds (v_x, v_y, v_z), angular rates $\left(\dot{\varphi}, \dot{\theta}, \dot{\psi}\right)$, and linear accelerations $\left(\dot{v}_x, \dot{v}_y, \dot{v}_z\right)$. Nevertheless, the retained filters are initialized in the two-dimensional mode for the considered linear 3-DoF bicycle model; thus, all the linear components along the Z-axis can be neglected. Also roll and pitch angles are excluded from the relevant states tracked. Therefore, the considered state vector is

$$x_t = \left[x \;\; y \;\; \psi \;\; v_x \;\; v_y \;\; \dot{\psi} \;\; \dot{v}_x \;\; \dot{v}_y\right]^T \qquad \text{Eq. (14)}$$

2.3.1. EKF Design The EKF has been introduced to overcome the assumption of linearity in state transition in the standard Kalman Filter. Its working principle is based on the linearization of the nonlinear functions (g, h) in Equations 12 and 13 with a GRV, i.e., the state distribution, which is then propagated through a first-order linearization of the nonlinear system via Taylor expansion, as in Equations 15 and 16. Hence, the main assumption that the initial state and measurement noises are Gaussian and uncorrelated with each other still holds. The formulation of the investigated EKF algorithm is presented in Algorithm 1 [37]. The output matrices G_t and H_t are given by

$$G_t := g'\left(u_t, \hat{x}_{t|t-1}\right) = \frac{\partial g\left(u_t, \hat{x}_{t|t-1}\right)}{\partial \hat{x}_{t|t-1}} \qquad \text{Eq. (15)}$$

$$H_t := h'\left(\hat{x}_{t|t}\right) = \frac{\partial h\left(\hat{x}_{t|t}\right)}{\partial \hat{x}_{t|t}} \qquad \text{Eq. (16)}$$

The process noise covariance and measurements noise covariance are denoted respectively as R_t and Q_t in Algorithm 1, while the matrix Q_t is assumed constant and will estimate its entries from measurements; the process noise covariance R_t is used as a tuning variable according to the application (e.g., dynamic racing situation or urban environment scenario). The output of the EKF algorithm is the sequence of state estimates $\hat{x}_{t|t}$ and of matrices $\Sigma_{t|t}$, starting from the given $\hat{x}_{0|0}$ and $\Sigma_{0|0}$.

A similar implementation of the discussed EKF algorithm is on the basis of both the sensor fusion routine performed at a sensor level by the retained INS device and the proposed EKF-based method. Specifically, the INS sensor pre-processes the IMU data and raw GNSS data to lower the computational burden on the main state estimation EKF routine running on board. While the INS filter runs at 200 Hz and is tuned according to the motion profile given by the manufacturer, the developed EKF instance running on board has an update frequency of 30 Hz and its process noise covariance has been heuristically tuned based on the experimentally collected data. Nevertheless, high nonlinearities in the filtering process could affect the EKF. The introduction of a first-order propagation through the system dynamics has proven to introduce large uncertainty in the estimate of parameters, thus leading

ALGORITHM 1 Extended Kalman filter.

$\left(\hat{x}_{t-1|t-1}, \Sigma_{t-1|t-1}, u_t, z_t\right)$
$\hat{x}_{t|t-1} = g\left(u_t, \hat{x}_{t-1|t-1}\right)$
$\hat{\Sigma}_{t|t-1} = G_t \Sigma_{t-1|t-1} G_t^T + R_t$
$K_t = \hat{\Sigma}_{t|t-1} H_t^T \left(H_t \hat{\Sigma}_{t|t-1} H_t^T + Q_t\right)^{-1}$
$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t\left(z_t - h\left(\hat{x}_{t|t-1}\right)\right)$
$\Sigma_{t|t} = (I - K_t H_t)\hat{\Sigma}_{t|t-1}$
return $\hat{x}_{t|t}, \Sigma_{t|t}$

to suboptimal performance and eventually to filter's divergence [37]. This tendency will be analyzed for the EKF deployed on board in the results presented in Section 3.

2.3.2. UKF Design

The driving scenario of the considered vehicles is characterized by strong nonlinearities. Moreover, the usage of a local path planner imposes the need for the highest accuracy to avoid any possible source of divergence of the measurement in real-time localization and states estimation. In this framework, the implementation of a UKF is investigated. UKF has proven in the literature to guarantee improved performances in terms of accuracy and robustness of the estimate with respect to a well-tuned EKF. On the other side, UKFs have a level of complexity that is comparable with EKFs; thus, the required computational effort is expected to be lower than the one required by non-Gaussian Filters, i.e., nonparametric filters [23]. Nevertheless, considering the computational cost analysis performed in [38] and [39], it can be easily concluded that UKF is characterized by a higher computational cost with respect to EKF. However, this aspect has not limited in the present work the applicability of the investigated algorithms. In this paper, a UKF has been developed to properly address the limitations of the EKF by using a deterministic sampling approach. The purpose is to better linearize the nonlinear functions (g, h) in Equations 12 and 13, improving the linearization performed by the EKF and the propagation of a GRV through the system dynamics. Since the UKF is a Gaussian Filter, the state distribution can still be approximated as a GRV, but in the UKF it is represented by a set of $2n + 1$ points based on its original mean and variance, called $\sigma - points$. A robust theoretical background can be found in [40]. When $\sigma - points$ are propagated through the nonlinear system by means of the Unscented transformation, it has been proven that the filter is able to capture the state estimate mean and covariance accurately to the third order of the Taylor series expansion for any nonlinearity; while for reference, the EKF estimate is accurate only to the first order [41]. In the proposed approach, the implemented UKF formulation is reported in Algorithm 2. The choice of the sigma points and the filter weights are reported in the following equations, as discussed in [40]. UKF σ-points are defined in Equations 17 and 18, while UKF weights are defined in Equations 19-21.

$$\chi^{[0]} = \mu \chi^{[i]} = \mu + \left(\sqrt{(n+\lambda)\Sigma}\right)_i \quad \text{for } i = 1,\ldots,n \qquad \text{Eq. (17)}$$

$$\chi^{[i]} = \mu - \left(\sqrt{(n+\lambda)\Sigma}\right)_{i-n} \quad \text{for } i = n+1,\ldots,2n \qquad \text{Eq. (18)}$$

$$w_m^{[0]} = \frac{\lambda}{n+\lambda} \qquad \text{Eq. (19)}$$

$$w_c^{[0]} = w_m^{[0]} + \left(1 - \alpha^2 + \beta\right) \qquad \text{Eq. (20)}$$

$$w_m^{[i]} = w_c^{[i]} = \frac{1}{2(n+\lambda)} \quad \text{for } i = 1,\ldots,2n \qquad \text{Eq. (21)}$$

where the parameters are chosen as follows: $\alpha = 0.001$, $\beta = 2$, $\kappa = 3 - n$, and $\lambda = \alpha^2(n + \kappa) - n$. According to [40], the choice of $\beta = 2$ minimizes the error in the fourth-order moment of the a posteriori covariance when the state distribution is a GRV. As already retained for the EKF, the matrix Q_t representing the measurements noise covariance is assumed to be constant, while the process noise covariance R_t is used as a tuning variable. Although the UKF brings in principle a heavier computational effort with respect to the EKF, a proper tuning of the process noise covariance can guarantee a good performance of the filter in terms of posterior estimate covariance while having the same responsiveness of the EKF instance, as stated by [23]. The complete UKF pipeline runs on the onboard computing platform at a frequency of 30 Hz and can deliver an accurate estimate of the vehicle localization and states.

ALGORITHM 2 Unscented Kalman filter.

$(\hat{x}_{t-1|t-1}, \Sigma_{t-1|t-1}, u_t, z_t)$
$\chi_{t-1|t-1} = \left(\hat{x}_{t-1|t-1} \quad \hat{x}_{t-1|t-1} + \sqrt{(n+\lambda)\Sigma_{t-1|t-1}} \quad \hat{x}_{t-1|t-1} - \sqrt{(n+\lambda)\Sigma_{t-1|t-1}}\right)$
$\hat{x}^*_{t|t-1} = g(u_t, \chi_{t-1|t-1})$
$\hat{x}_{t|t-1} = \sum_{i=0}^{2n} w_m^{[i]} \hat{x}^{*[i]}_{t|t-1}$
$\hat{\Sigma}_{t|t-1} = \sum_{i=0}^{2n} w_c^{[i]} \left(\hat{x}^{*[i]}_{t|t-1} - \hat{x}_{t|t-1}\right)\left(\hat{x}^{*[i]}_{t|t-1} - \hat{x}_{t|t-1}\right)^T + R_t$
$\hat{x}_{t|t-1} = \left(\hat{x}_{t|t-1} \quad \hat{x}_{t|t-1} + \sqrt{(n+\lambda)\hat{\Sigma}_{t|t-1}} \quad \hat{x}_{t|t-1} - \sqrt{(n+\lambda)\hat{\Sigma}_{t|t-1}}\right)$
$\hat{z}_{t|t-1} = h(\hat{x}_{t|t-1})$
$\hat{z}_{t|t-1} = \sum_{i=0}^{2n} w_m^{[i]} \hat{z}^{*[i]}_{t|t-1}$
$S_{t|t-1} = \sum_{i=0}^{2n} w_c^{[i]} \left(\hat{z}^{[i]}_{t|t-1} - \hat{z}_{t|t-1}\right)\left(\hat{z}^{[i]}_{t|t-1} - \hat{z}_{t|t-1}\right)^T + Q_t$
$\hat{\Sigma}^{x,z}_{t|t-1} = \sum_{i=0}^{2n} w_c^{[i]} \left(\hat{x}^{[i]}_{t|t-1} - \hat{x}_{t|t-1}\right)\left(\hat{x}^{[i]}_{t|t-1} - \hat{x}_{t|t-1}\right)^T$
$K_t = \hat{\Sigma}^{x,z}_{t|t-1} {S_{t|t-1}}^{-1}$
$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t\left(z_t - \hat{z}_{t|t-1}\right)$
$\Sigma_{t|t} = \hat{\Sigma}_{t|t-1} - K_t S_{t|t-1} {K_t}^{-1}$
return $\hat{x}_{t|t}, \Sigma_{t|t}$

3. Results and Discussion

In this section, the results obtained with the two investigated state estimation methods are presented and discussed. The testing phase has been carried out in Italy (Piedmont region) on both the considered vehicles (namely, the racing prototype and the compact SUV) in the same driving scenario, but on different paths. The driving environment is represented in Figure 4 from an aerial view. The performed maneuvers are representative of many common situations during urban routes, which are challenging cases for a localization algorithm in autonomous driving.

The performance of the investigated algorithms is evaluated on the racing prototype during five consecutive laps, in which the vehicle starts at standstill and finishes its route again with a null longitudinal speed. The same approach has been used in the dataset recorded on the commercial vehicle in which the performance of the pipeline is tested on an open lap. Both the EKF and the UKF algorithms are properly initialized at the beginning of the acquisition when the INS sensor output is already computed in full alignment mode, neglecting the motion along the z-axis. This assumption is consistent

FIGURE 4 Aerial views of the driving scenario (experimental tests). Intended paths for the retained racing prototype (red) and compact SUV. Start of the track (black cross), end of the track (black dot). The actual path can vary, as it is not perfectly reproducible.

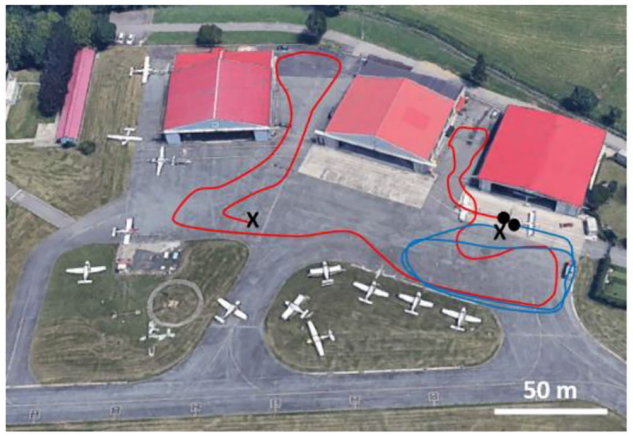
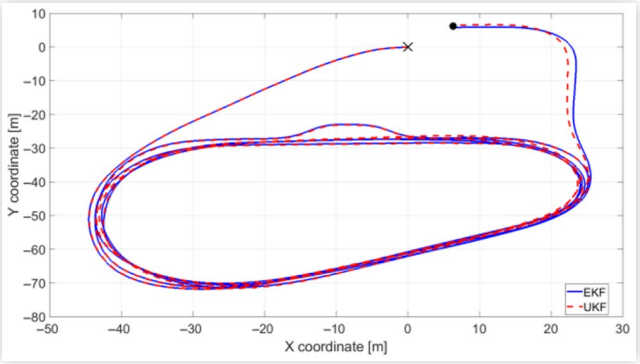


FIGURE 5 Dataset 1: racing prototype—Computed localization paths with EKF (blue, solid) and UKF (red, dashed). Start of the track (black cross), end of the track (black dot).



with the adopted vehicle's dynamics model, i.e., planar motion, and is also compatible with the ground flatness of the chosen testing environment. The measurements are recorded in the messages */odometry/filtered* and */accel/filtered* listed in Table 5, which report the complete odometry of the vehicle (orientation with covariance and pose with covariance) and the set of its linear accelerations with covariance, respectively.

Table 6 reports the maximum and mean values of the vehicle's states estimated during the two performed tests. Both the proposed filters achieve similar numerical results in the main vehicle's state estimation, with no significant oscillations of the values.

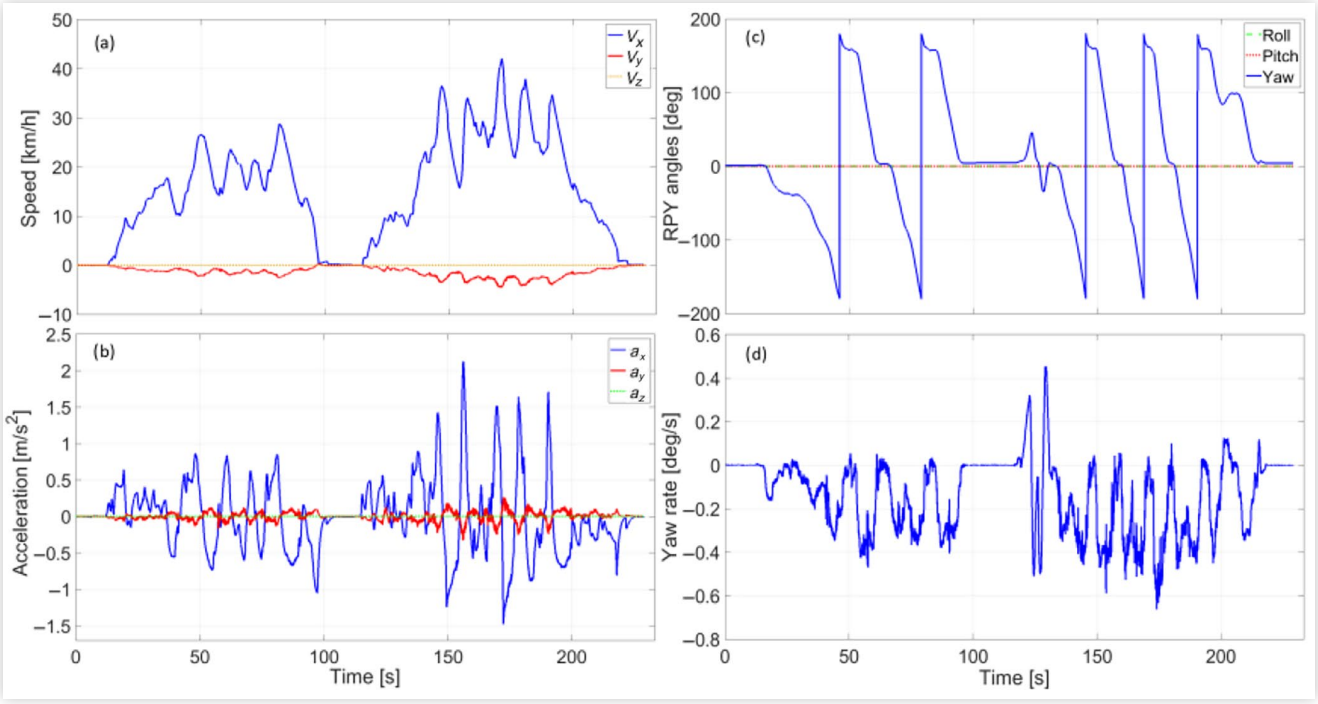
The computed localization paths for Dataset 1 (Table 6) recorded with the racing prototype are presented in Figure 5. The paths represented in Figure 5 show the vehicle's positions on the xy-plane computed by the EKF and the UKF, respectively. Specifically, the localization path computed with UKF is less prone to divergence with respect to the EKF one, thanks to the improved estimation of the uncertainty in the unscented transform. Nevertheless, the proposed filters have comparable performance in estimating the vehicle's position on the track having a submeter difference in the xy-coordinate estimation. The estimated vehicle states are illustrated in Figure 6 for Dataset 1 (Table 6) recorded on the racing prototype. Figure 6 reports all the relevant states estimated by the two realizations of the localization pipelines, namely, the linear speeds, the linear accelerations, RPY angles, and the yaw rate. The motion along the *z*-axis is neglected in the proposed localization pipeline; thus, speed and acceleration along this axis have null values, and only the yaw dynamics is analyzed in Subfigures 6.c and 6.d. The investigated UKF and EKF can compute equivalent values for the analyzed states; thus, Figure 6 reports data computed by one of the filters only, i.e., UKF.

The performance of the two methods can be evaluated by means of analyzing the covariance matrices of both the filters, as written in [23] and [42]. Specifically, the analysis has been carried out on the computed variances of the filtered states: the considered messages (*/odometry/filtered* and */accel/filtered* defined in Table 5) deliver the complete covariance matrix of the pose, twist, and acceleration of the vehicle. However, only the position measurements (namely, the x and y coordinates) are characterized by a non-null behavior of the associated variance. For all the other retained states (speeds, accelerations, and RPY angles), both the EKF and UKF have consistent performance and are characterized by a low and constant variance while assuring the uncorrelation of the estimated parameters. This result is obtained, thanks to both the accuracy of the measurements delivered by the INS sensor and to the de-centralized EKF embedded in the sensor. Therefore, the achieved results have been evaluated in terms of uncertainty determined by the covariance matrices and Horizontal Position Error (HPE). The true values about the

TABLE 6 Maximum and mean values of the estimated vehicle's states in the considered datasets.

		Dataset 1: Closed laps—prototype				Dataset 2: Open lap—compact SUV			
Estimated vehicle's states		Mean		Maximum		Mean		Maximum	
State	Unit	EKF	UKF	EKF	UKF	EKF	UKF	EKF	UKF
Yaw angle	[deg]	13.61	13.19	179.98	179.87	−17.69	−17.65	179.58	179.96
Longitudinal speed	[m/s]	4.13	4.12	11.68	11.68	3.57	3.57	6.33	6.32
Lateral speed	[m/s]	−0.37	−0.37	0.08	0.08	−0.04	−0.04	0.15	0.18
Yaw rate	[°/s]	−0.14	−0.14	0.45	0.46	−0.11	−0.11	0.92	0.92
Longitudinal acceleration	[m/s^2]	0.00	0.00	2.13	2.13	0.00	0.00	1.00	1.00
Lateral acceleration	[m/s^2]	−0.33	−0.33	0.26	0.26	0.48	0.48	0.15	0.15

FIGURE 6 Dataset 1: racing prototype—(a) Linear speeds: V_x (blue, solid); V_y (red, solid); V_z (orange, dotted). (b) Linear accelerations: a_x (blue, solid); a_y (red, solid); a_z (green, dotted). (c) RPY angles: roll (green, dashed); pitch (red, dotted); yaw (blue, solid). (d) Yaw rate $\dot{\psi}$.



vehicle's state variables could have been obtained via expensive integrated optical sensors. Nevertheless, due to the huge cost of those sensors, the performance of the proposed method has been evaluated by the comparison of the proposed EKF and UKF.

In detail, Figure 7 represents the variance of the estimated vehicle's position for the EKF-based method and the UKF-based method in the case of the racing prototype (Dataset 1).

An equivalent discussion about the results obtained with the compact SUV can be carried out. The results obtained with the compact SUV in the same driving environment are illustrated in Figures 8-10 during a different set of handling maneuvers.

Indeed, the filtering process of X and Y coordinates suffers from a consistent divergence of their associated variance for both the investigated filters since they depend mostly on the GNSS acquisitions. Even in presence of a clear view of the sky during the entire acquisitions, the GPS signal has the lowest acquisition rate among the used sensors; thus, it is more prone to divergence. Figure 11 represents the HPE for the positioning measurements of the EKF-based method in comparison with the UKF-based method, for both the considered datasets. HPE has been chosen as a method of comparison because it can represent the information

FIGURE 7 Dataset 1: racing prototype—Variance of the estimated vehicle's position for the EKF-based method (x: solid, blue; y: solid, red) and the UKF-based method (x: dashed, green; dotted, orange).

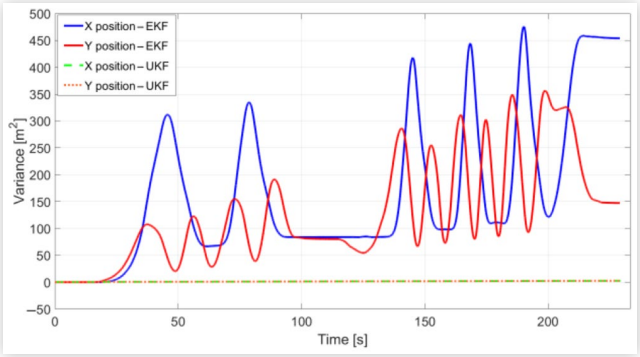


FIGURE 8 Dataset 2: compact SUV—Computed localization paths with EKF (blue, solid) and UKF (red, dashed). Start of the track (black cross), end of the track (black dot).

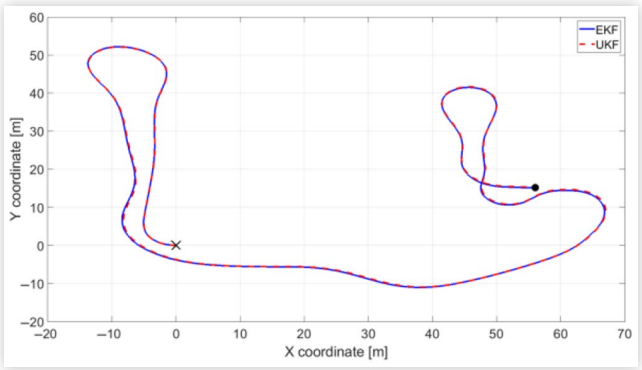


FIGURE 9 Dataset 2: compact SUV—(a) Linear speeds: V_x (blue, solid); V_y (red, solid); V_z (orange, dotted). (b) Linear accelerations: a_x (blue, solid); a_y (red, solid); a_z (green, dotted). (c) RPY angles: roll (green, dashed); pitch (red, dotted); yaw (blue, solid). (d) Yaw rate $\dot{\psi}$.

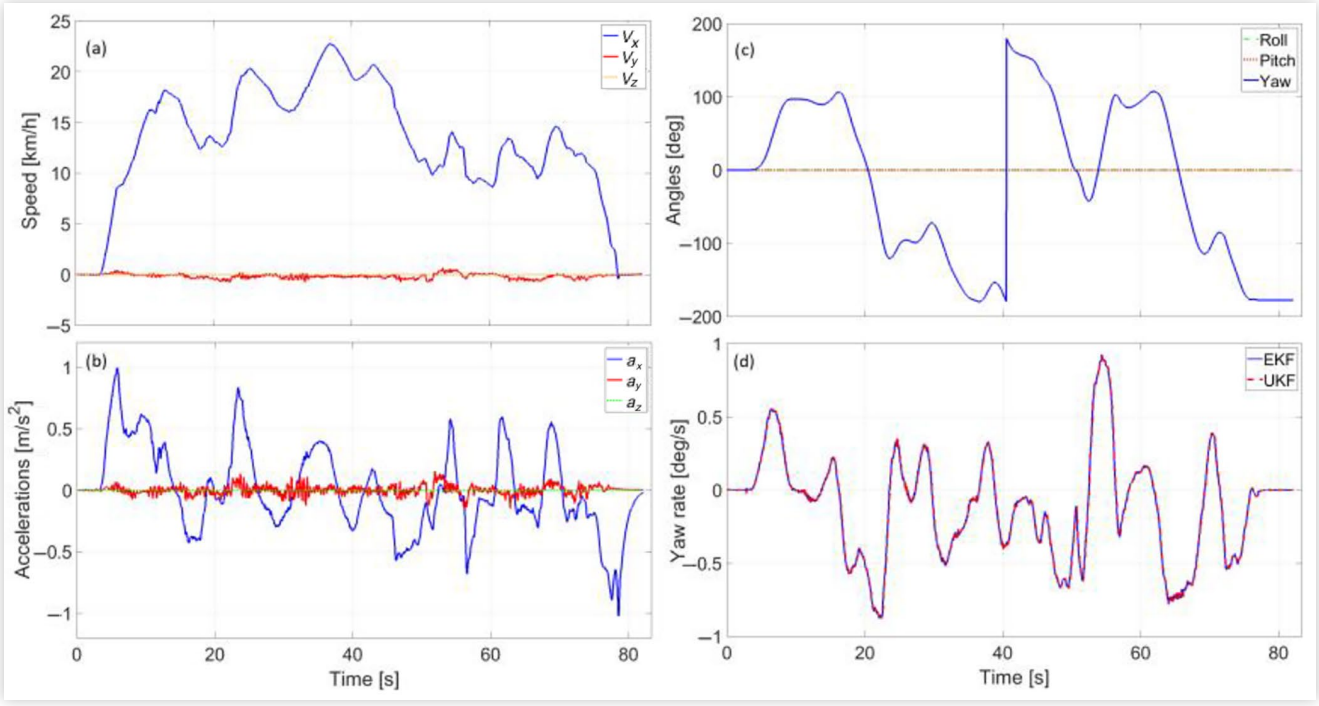
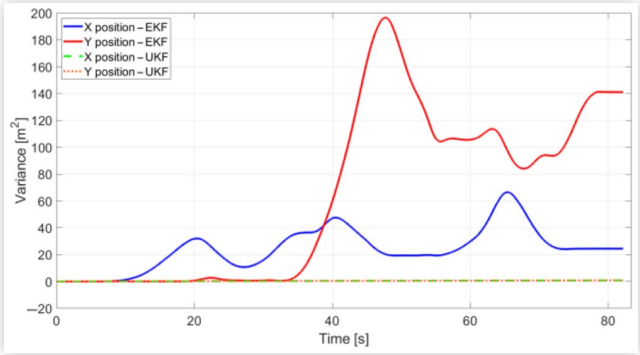


FIGURE 10 Dataset 2: compact SUV—Variance of the estimated vehicle's position for the EKF-based method (x: solid, blue; y: solid, red) and the UKF-based method (x: dashed, green; dotted, orange).



associated with the position error on the local map. To this end, HPE can be interpreted as the uncertainty on the current position on the map expressed in terms of mean estimation error on the two coordinates. HPE is defined from the knowledge of the variances σ_{xx}^2 and σ_{yy}^2 associated to the X and Y coordinates in the map, as follows:

$$HPE = \sqrt{\sigma_{xx}^2 + \sigma_{yy}^2} \qquad \text{Eq. (22)}$$

The UKF-based method is characterized by a better performance in containing the growth of the uncertainty, delivering a consistent and non-oscillating behavior during the entire acquisition. Therefore, the overall localization performance on vehicle position on the map is better when implementing the UKF-based method with respect to the EKF routine running at the sensor level.

The performance analysis on the variance between EKF and UKF algorithms has been quantified in terms of minimum, mean, and maximum absolute difference values between the two algorithms for X and Y positions along the travelled paths. In Table 7, also the abovementioned values for the HPE have been reported for completeness with reference to Figure 11.

4. Conclusion

Research interest toward autonomous vehicles has recently motivated design challenges related to the safety and performances of the next generation of automated cars. Vehicle localization represents one of the most relevant issues for autonomous vehicles. In this paper, a localization and states estimation method for autonomous driving has been presented. The investigated approach proposed and compared the application of the EKF and UKF to the sensor fusion of onboard data streaming from a GPS sensor and an INS. The purpose of having a lean hardware architecture reduced the number of considered sensors in the localization pipeline. The solution is presented as a reliable alternative to existing methods since it can accurately estimate vehicle's states and provide a submeter localization accuracy. The performance of the method was evaluated experimentally during handling

FIGURE 11 (a) Dataset 1: racing prototype—HPE for EKF (blue) and UKF (red) methods. (b) Dataset 2: compact SUV—HPE for EKF (blue) and UKF (red) methods.

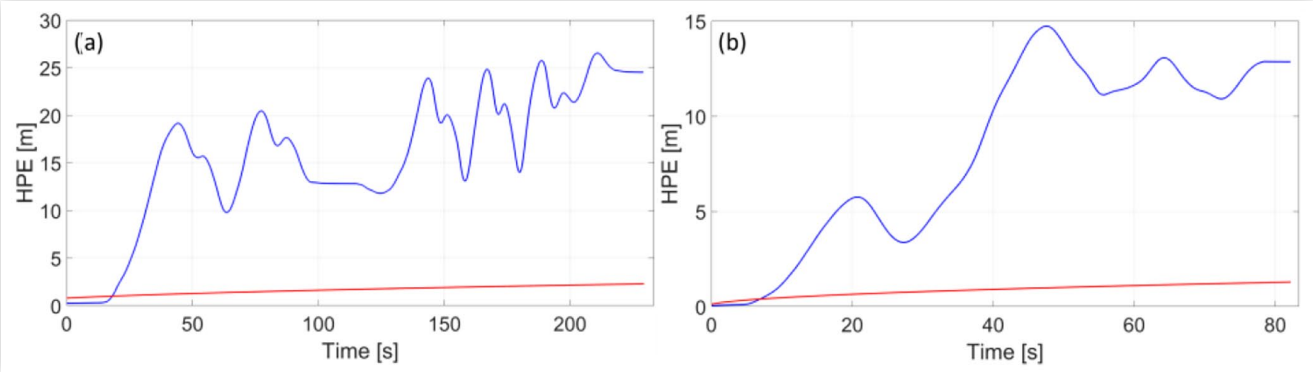


TABLE 7 Performance analysis in terms of variance (X and Y coordinates) and HPE.

		Dataset 1: Racing prototype						Dataset 2: Compact SUV					
Performance		Min		Mean		Max		Min		Mean		Max	
Parameter	Unit	EKF	UKF	EKF	UKF	EKF	UKF	EKF	UKF	EKF	UKF	EKF	UKF
Variance on X position	[m^2]	0.03	0.31	169.89	1.46	475.89	2.60	0.01	0.01	24.35	0.42	66.49	0.83
Variance on Y position	[m^2]	0.03	0.31	127.52	1.46	356.21	2.60	0.01	0.01	64.55	0.42	196.56	0.83
Horizontal position error	[m]	0.25	0.79	15.77	1.66	26.53	2.28	0.04	0.13	8.16	0.87	14.73	1.29

maneuvers in real driving environments, both on a racing prototype and on a commercial compact SUV. The performance of the algorithm is validated in real time while performing common maneuvers during urban driving, showing a substantial equivalence of the application of the two filters. Nevertheless, better results were achieved with the UKF-based method that is characterized by a lower estimation variance in the localization task.

After a further extensive on-field validation stage considering highways, parking, and other driving scenarios, which has not been conducted in the present work, the localization method described in this paper could provide a reliable pipeline for autonomous vehicles in actual industrial applications. Eventually, as the problem of providing ground truth for position data is widely discussed in the recent literature, it was beyond the scope of the present work to design and provide a ground truth algorithm for position data, although the paper has highlighted the applicability of EKF and UKF filters for position and state estimation and compared results between the two filters in different driving situations. Thus the achieved results have been evaluated in terms of uncertainty determined by the covariance matrices and HPE. The true values about vehicle's state variables could have been obtained from accurate dynamic measurement technology acceleration applications, which usually come from expensive integrated optical sensors. However, due to the huge cost of those sensors, the performance of the proposed method has been evaluated by the comparison of the two investigated techniques.

Declaration of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was developed in the framework of the activities of the Interdepartmental Center for Automotive Research and Sustainable Mobility (CARS) at Politecnico di Torino (www.cars.polito.it).

Contact Information

Stefano Feraco
stefano.feraco@polito.it

References

1. Luettel, T., Himmelsbach, M., and Wuensche, H.J., "Autonomous Ground Vehicles—Concepts and a Path to the Future," *Proceedings of the IEEE* 100, no. Special Centennial Issue (2012): 1831-1839.

2. Fraedrich, E. and Lenz, B., "Automated Driving: Individual and Societal Aspects," *Transportation Research Record* 2416, no. 1 (2014): 64-72.
3. Ziebinski, A., Cupek, R., Grzechca, D., and Chruszczyk, L., "Review of Advanced Driver Assistance Systems (ADAS)," *AIP Conference Proceedings* 1906, no. 1 (2017): 120002.
4. Bonfitto, A., Feraco, S., Tonoli, A., and Amati, N., "Combined Regression and Classification Artificial Neural Networks for Sideslip Angle Estimation and Road Condition Identification," *Vehicle System Dynamics* 58, no. 11 (2020): 1766-1787.
5. Luciani, S., Bonfitto, A., Amati, N., and Tonoli, A., "Comfort-Oriented Design of Model Predictive Control in Assisted and Autonomous Driving," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, St. Louis, MO, 2020, Vol. 83938, V004T04A008, American Society of Mechanical Engineers
6. Hörl, S., Ciari, F., and Axhausen, K.W., "Recent Perspectives on the Impact of Autonomous Vehicles," *Arbeitsberichte Verkehrs-und Raumplanung* 1216 (2016): 5-18.
7. Kaur, K. and Rampersad, G., "Trust in Driverless Cars: Investigating Key Factors Influencing the Adoption of Driverless Cars," *Journal of Engineering and Technology Management* 48 (2018): 87-96.
8. Kuutti, S., Fallah, S., Katsaros, K., Dianati, M. et al., "A Survey of the State-of-the-Art Localization Techniques and Their Potentials for Autonomous Vehicle Applications," *IEEE Internet of Things Journal* 5, no. 2 (2018): 829-846.
9. Woo, A., Fidan, B., and Melek, W.W. "Localization for Autonomous Driving," Wiley, *Handbook of Position Location: Theory, Practice, and Advances*, 2nd ed. (Hoboken, NJ: Wiley, 2018), 1051-1087.
10. Leonard, J.J. and Durrant-Whyte, H.F., "Mobile Robot Localization by Tracking Geometric Beacons," *IEEE Transactions on Robotics and Automation* 7, no. 3 (1991): 376-382.
11. Feraco, S., Bonfitto, A., Khan, I., Amati, N. et al., "Optimal Trajectory Generation Using an Improved Probabilistic Road Map Algorithm for Autonomous Driving," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Anaheim, CA (USA), August 2020, Vol. 83938, V004T04A006, American Society of Mechanical Engineers.
12. Feraco, S., Luciani, S., Bonfitto, A., Amati, N. et al., "A Local Trajectory Planning and Control Method for Autonomous Vehicles Based on the RRT Algorithm," in *2020 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)*, Torino, Italy, 2020, 1-6.
13. Formula Student Germany, "FSG Competition Handbook 2019," 2019, https://www.formulastudent.de/fileadmin/user_upload/all/2019/rules/FSG19_Competition_Handbook_v1.0.pdf.
14. Marin-Plaza, P., Hussein, A., Martin, D., and Escalera, A.D.L., "Global and Local Path Planning Study in a ROS-Based Research Platform for Autonomous Vehicles," *Journal of Advanced Transportation* (2018), Article ID 6392697, 0197-6729.
15. Kabzan, J., Valls, M.I., Reijgwart, V.J., Hendrikx, H.F. et al., "Amz Driverless: The Full Autonomous Racing System," *Journal of Field Robotics* 37, no. 7 (2020): 1267-1294.
16. Nekkah, S., Janus, J., Boxheimer, M., Ohnemus, L. et al., "The Autonomous Racing Software Stack of the KIT19d," 2020, arXiv preprint arXiv:2010.02828.
17. Zeilinger, M., Hauk, R., Bader, M., and Hofmann, A. "Design of an Autonomous Race Car for the Formula Student Driverless (FSD) in *Oagm & Arw Joint Workshop*, Wien, Austria, 2017.
18. Chen, T., Li, Z., He, Y., Xu, Z. et al, "From Perception to Control: An Autonomous Driving System for a Formula Student Driverless Car," 2019 arXiv preprint arXiv:1909.00119.
19. Cadena, C., Carlone, L., Carrillo, H., Latif, Y. et al., "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Transactions on Robotics* 32, no. 6 (2016): 1309-1332.
20. Ferreira, J.V., "Sensor Fusion Tests for an Autonomous Vehicle, Using Extended Kalman Filter," *Journal of Engineering Science and Technology Review* 11, no. 3 (2018): 1-8.
21. Meng, X., Wang, H., and Liu, B., "A Robust Vehicle Localization Approach Based on gnss/imu/dmi/Lidar Sensor Fusion for Autonomous Vehicles," *Sensors* 17, no. 9 (2017): 2140.
22. Lin, M., Yoon, J., and Kim, B., "Self-Driving Car Location Estimation Based on a Particle-Aided Unscented Kalman Filter," *Sensors* 20, no. 9 (2020): 2544.
23. D'Alfonso, L., Lucia, W., Muraca, P., and Pugliese, P., "Mobile Robot Localization via EKF and UKF: A Comparison Based on Real Data," *Robotics and Autonomous Systems* 74 (2015): 122-127.
24. Konatowski, S., Kaniewski, P., and Matuszewski, J., "Comparison of Estimation Accuracy of EKF, UKF and PF Filters," *Annual of Navigation* 23 (2016): 69-87.
25. Xue, Z. and Schwartz, H., "A Comparison of Several Nonlinear Filters for Mobile Robot Pose Estimation," in *2013 IEEE International Conference on Mechatronics and Automation*, Takamatsu, Japan, August 2013, 1087-1094, IEEE.
26. Allotta, B., Chisci, L., Costanzi, R., Fanelli, F. et al., "A Comparison between EKF-Based and UKF-Based Navigation Algorithms for AUVs Localization in *OCEANS 2015*, Genova, May 2015, 1-5, IEEE.
27. Khan, I., Feraco, S., Bonfitto, A., and Amati, N., "A Model Predictive Control Strategy for Lateral and Longitudinal Dynamics in Autonomous Driving," in *ASME 2020 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Virtual Conference, August 2020, American Society of Mechanical Engineers Digital Collection.
28. Amer, N., Zamzuri, H., Hudha, K., and Kadir, Z., "Modelling and Control Strategies in Path Tracking Control for Autonomous Ground Vehicles: A Review of State of the Art and Challenges," *Journal of Intelligent & Robotic Systems* 86 (2017): 225-254.

29. Feraco, S., Bonfitto, A., Amati, N., and Tonoli, A. "Combined Lane Keeping and Longitudinal Speed Control for Autonomous Driving in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Anaheim, CA (USA), August 2019, Vol. 59216, V003T01A018, American Society of Mechanical Engineers.
30. Kiencke, U. and Nielsen, L., *Automotive Control Systems: For Engine, Driveline, and Vehicle* (Berlin: Springer, 2000)
31. Rajamani, R., *Vehicle Dynamics and Control* (New York: Springer Science & Business Media, 2011)
32. Barshan, B. and Durrant-Whyte, H.F., "Inertial Navigation Systems for Mobile Robots," *IEEE Transactions on Robotics and Automation* 11, no. 3 (1995): 328-342.
33. Lazarou, T. and Danezis, C., "Assessment of Modern Smartphone Sensors Performance on Vehicle Localization in Urban Environments," in *Fifth International Conference on Remote Sensing and Geoinformation of the Environment (RSCy2017)*, Paphos, Cyprus, September 2017, Vol. 10444, 104441S, International Society for Optics and Photonics.
34. Moore, T. and Stouch, D., "A Generalized Extended Kalman Filter Implementation for the Robot Operating System," Springer, *Intelligent Autonomous Systems 13* (Cham:Springer, 2016), 335-348.
35. Thrun, S., Burgard, W., and Fox, D., *Probabilistic Robotics*, Intelligent Robotics and Autonomous Agents Series (Cambridge, MA: The MIT Press, 2006)
36. Särkkä, S., *Bayesian Filtering and Smoothing*. Vol. 3 (Cambridge: Cambridge University Press, 2013)
37. Bishop, G. and Welch, G., "An Introduction to the Kalman Filter," in *Proceedings of SIGGRAPH*, Los Angeles, CA (USA), 2001, Course, 8(27599-23175), 41.
38. Zhang, B., Chu, H., Sun, T., Jia, H. et al., "Error Prediction for SINS/GPS after GPS Outage Based on Hybrid KF-UKF," *Mathematical Problems in Engineering* 2015 (2015): 1-9.
39. Ali, H.F., Mansour, N.A., and Kim, Y., "Comparative Study of Extended and Unscented Kalman Filters for Estimating Motion States of an Autonomous Vehicle-Trailer System," Springer, *Recent Advances in Mechanical Engineering* (Singapore: Springer, 2021), 165-173.
40. Julier, S.J. and Uhlmann, J.K., "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE* 92, no. 3 (2004): 401-422, 10.1109/JPROC.2003.823141.
41. Wan, E.A. and Van Der Merwe, R., "The Unscented Kalman Filter for Nonlinear Estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, Lake Louise, Alberta, Canada, 2000, 153-158, 10.1109/ASSPCC.2000.882463.
42. Higham, N.J., *Accuracy and Stability of Numerical Algorithms* (Society for Industrial and Applied Mathematics, Philadelphia, PA (USA), 2002)

ISSN 0148-7191