

3D Distance Filter for the Autonomous Navigation of UAVs in Agricultural Scenarios

Original

3D Distance Filter for the Autonomous Navigation of UAVs in Agricultural Scenarios / Donati, C., Mammarella, M., Comba, L., Biglia, A., Gay, P., Dabbene, F.. - In: REMOTE SENSING. - ISSN 2072-4292. - ELETTRONICO. - 14:6(2022), pp. 1-18. [10.3390/rs14061374]

Availability:

This version is available at: 11583/2958228 since: 2022-11-21T11:55:18Z

Publisher:

MDPI

Published

DOI:10.3390/rs14061374

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)



Article

3D Distance Filter for the Autonomous Navigation of UAVs in Agricultural Scenarios

Cesare Donati ¹, Martina Mammarella ^{1,*} , Lorenzo Comba ^{1,2} , Alessandro Biglia ² , Paolo Gay ^{2,†} and Fabrizio Dabbene ¹

¹ Institute of Electronics, Computer and Telecommunication Engineering (IEIIT), National Research Council of Italy, Corso Duca degli Abruzzi 24, 10129 Turin, Italy; cesare.donati@ieiit.cnr.it (C.D.); lorenzo.comba@unito.it (L.C.); fabrizio.dabbene@ieiit.cnr.it (F.D.)

² Department of Agricultural, Forest and Food Sciences (DiSAFA), Università degli Studi di Torino, Largo Paolo Braccini 2, 10095 Grugliasco, Italy; alessandro.biglia@unito.it (A.B.); paolo.gay@unito.it (P.G.)

* Correspondence: martina.mammarella@ieiit.cnr.it

† Co-last author.

Abstract: In precision agriculture, remote sensing is an essential phase in assessing crop status and variability when considering both the spatial and the temporal dimensions. To this aim, the use of unmanned aerial vehicles (UAVs) is growing in popularity, allowing for the autonomous performance of a variety of in-field tasks which are not limited to scouting or monitoring. To enable autonomous navigation, however, a crucial capability lies in accurately locating the vehicle within the surrounding environment. This task becomes challenging in agricultural scenarios where the crops and/or the adopted trellis systems can negatively affect GPS signal reception and localisation reliability. A viable solution to this problem can be the exploitation of high-accuracy 3D maps, which provide important data regarding crop morphology, as an additional input of the UAVs' localisation system. However, the management of such big data may be difficult in real-time applications. In this paper, an innovative 3D sensor fusion approach is proposed, which combines the data provided by onboard proprioceptive (i.e., GPS and IMU) and exteroceptive (i.e., ultrasound) sensors with the information provided by a georeferenced 3D low-complexity map. In particular, the parallel-cuts ellipsoid method is used to merge the data from the distance sensors and the 3D map. Then, the improved estimation of the UAV location is fused with the data provided by the GPS and IMU sensors, using a Kalman-based filtering scheme. The simulation results prove the efficacy of the proposed navigation approach when applied to a quadrotor that autonomously navigates between vine rows.

Keywords: precision farming; 3D crop modelling; sensor fusion; autonomous vehicles in agriculture; ellipsoid method; Kalman filter



Citation: Donati, C.; Mammarella, M.; Comba, L.; Biglia, A.; Gay, P.; Dabbene, F. 3D Distance Filter for the Autonomous Navigation of UAVs in Agricultural Scenarios. *Remote Sens.* **2022**, *14*, 1374. <https://doi.org/10.3390/rs14061374>

Academic Editor: Jorge Delgado García

Received: 12 February 2022

Accepted: 9 March 2022

Published: 11 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Autonomous vehicles such as unmanned aerial vehicles (UAVs) are systems that are designed to operate in an unpredictable and partially unknown environment. This requires the UAVs to be able to navigate without disruption and to avoid any obstacle placed within their movement boundaries. In this framework, two main concepts are of particular relevance: (i) perception, i.e., the ability to acquire information from the vehicle environment and to sense objects around it or their relative position; and (ii) navigation, i.e., the ability to locate the vehicle in the surrounding environment by the merging of the sensors' data, combined with the vehicle dynamics [1]. To achieve these tasks, information is perceived by the use of sensors in order to autonomously perform the UAV localisation and navigation [2].

Sensors are devices that map the detected events or changes in the surroundings to a quantitative measurement for further processing, useful for UAVs' path planning and decision making, and essential precursors for controlling the motion of the drone.

The sensors used for data collection and autonomous navigation are typically grouped into two major categories, (i) proprioceptive and (ii) exteroceptive sensors. Proprioceptive sensors measure values internal to the system (e.g., UAV), whereas exteroceptive sensors perceive and acquire information such as the distance measurements or the light intensity from the surroundings of the system. According to these definitions, inertial measurement unit (IMU) and global positioning system (GPS) devices are classified as proprioceptive sensors, while sonar, ultrasonic distance sensors, and infrared (IR) sensors are some examples of exteroceptive sensors [3]. Therefore, the sensing capability of a UAV that employs a diverse set of sensors is an essential element as the cooperation and performance of these sensors can directly determine the viability and safety of an autonomous vehicle. UAVs primarily utilise ultrasonic sensors to perceive the operative environment [4]. Additionally, other sensors, including GPS and IMU, are used to determine the relative and absolute positions of the vehicle. Therefore, the merging and integration of the data from several sources becomes crucial.

This task is typically identified as sensor fusion, whose definition is provided by the Joint Directors of Laboratories Workshop [5] as “a multi-level process dealing with the association, correlation, combination of data and information from single and multiple sources to achieve refined position, identify estimates and complete and timely assessments of situations, threats and their significance”. Therefore, sensor fusion techniques combine data from multiple sources to achieve better accuracy than the one obtained by the use of a single sensor. The sensor fusion framework also involves the development of consistent models that can accurately perceive the surroundings in various environmental conditions. The goal of using data fusion in multi-sensor frameworks is to reduce the detection error probability and to increase the information reliability by using data gathered from multiple distributed sources.

The available data fusion techniques can be classified into three main categories: (i) data association, (ii) decision fusion, and (iii) state estimation. Other classifications have been proposed over the years, as reported in [6]. The goal of the data association techniques is to establish the set of observations and/or measurements that are generated by the same target over time. On the other hand, for the decision fusion techniques, a decision is taken based on the knowledge of the perceived situation, provided by many sources. These techniques aim at making a high-level inference about the events and activities that are produced by the detected targets. Finally, the term state estimation identifies the techniques that determine the state of the moving target, given its observation or measurements. The final goal is to obtain a global target state from the observations coming from different sensors or sources.

The estimation problem can be defined as follows: given a set of redundant observations of the vector state (e.g., position, velocity, orientation), the goal is to find the set of parameters that provides the best fit to the observed data. Indeed, data acquisition is typically affected by errors and measurement noise. In the last fifty years, numerous state estimation techniques have been proposed, starting from the well-known Kalman filter [7]. Most of the state estimation methods are based on control theory and employ the laws of probability to compute a vector state from a vector measurement or a stream of vector measurements. The most common estimation methods, in addition to the Kalman-like filters [8,9], include the maximum likelihood [10] and maximum posterior approach [11], the particle filter [12–14], the distributed particle filter [15,16], and the covariance consistency methods [17].

UAV navigation systems significantly rely on knowledge of the environment, which needs to be reliably described to allow an accurate determination of both drone position and orientation [18]. In this sense, 3D high-accuracy maps can represent a valuable resource to improve the UAV's capability of autonomously navigating in the field [19]. This ability to correctly assess the location of UAVs becomes fundamental in modern applications requiring the autonomous and coordinated navigation of agricultural unmanned vehicles, both aerial and ground, in the agriculture 4.0 framework [20]. Such a 3D map can be derived

from several types of sources, such as LiDAR sensors [21], structured light cameras [22], or even standard cameras, by photogrammetric techniques [23] or stereo vision [24,25]. The raw data is usually a point cloud, which is a set of 3D coordinates of unlabelled points in the space representing the external surface of the detected objects [26,27].

On the other hand, retrieving 3D maps first and then exploiting them in real time for guidance and navigation tasks is challenging. This is mainly due to the required high data storage and computational demand with respect to the commonly available UAV hardware. Moreover, as highlighted in [3], to accurately determine the position and orientation of the UAV in its operative environment, it is better to exploit simple and understandable structures. The main technique for representing the environment is the geometric one, which is used to describe the environment by parameterising primitive geometric objects such as curves, lines, and points. The geometric representation of the environment is closer to the sensor and actuator world and it is the best one to perform local navigation.

Several triangulation techniques have been proposed to properly approximate surfaces or volumes in space by generating 2D or 3D meshes from sets of points [28]. Even if the most widely adopted techniques rely on Delaunay triangulations [29] and Voronoi diagrams [30], many other effective approaches have been developed for triangulated mesh generation, such as the Powell–Sabin-based methods (Barrera 2021) or the moving parabolic approximation approach [31]. The refinement of triangulated meshes is also a crucial task, essential for reducing the number of instances used to spatially describe an environment [32,33]. In addition, the semantic interpretation of a uniform dataset is essential to fully exploit the information potential of the 3D models of real scenarios, by assigning proper labels to each different element's category [34]. An innovative approach was proposed in [35] to generate low-complexity 3D mesh models of vine rows starting from a point cloud-processing pipeline. The main output of this processing flow was a 3D mesh, still properly describing the spatial layout and shape of the vine, but with a significant reduction of the required data storage demands. The efficacy of the so-called low-complexity georeferenced (LCG) maps was already validated in [36]. In this paper, the LCG maps were exploited to automatically build the obstacle/occupancy grid map used by the guidance and control schemes for allowing autonomous vehicles to navigate between vine rows. On the other hand, the authors assumed to know the vehicle's exact location, neglecting the sensor fusion task for autonomous navigation.

In agricultural scenarios, when relying on low-cost/low-accuracy sensors, the autonomous navigation of vehicles into narrow and complex environments could encounter several issues. Indeed, when operated within the vine rows, UAVs could be affected by the GPS signal being compromised or lost, which can jeopardise their navigation capabilities and lead to collisions. For this reason, in [37], we first proposed an innovative sensor fusion approach in which the data provided by GPS, IMU, and ultrasound sensors were fused together with the information provided by the LCG maps. In that paper, the ellipsoid method was applied to fuse the data collected by the ultrasonic sensors and the LCG map to improve the location estimation of an unmanned ground vehicle (UGV) navigating within the vine rows.

In this paper, we extend the aforementioned approach to the 3D case, thus allowing for the use of a similar filtering scheme for the autonomous navigation of UAVs into narrow spaces, such as vine rows. The main differences can be summarised as follows: (i) a real-time selection of the reference plane with respect to which the distance between the UAV center-of-mass (CoM), and assessment of the projected LCG map; (ii) the use of a Kalman-based filter to fuse the data from the combined ultrasound sensors–LCG maps and the data of the GPS and IMU with their correlated uncertainty; and (iii) the use of a moving-window approach to select the section of the 3D mesh of interest, complying with the estimated UAV location, thus allowing for a significant reduction of the algorithm's computational complexity. The efficacy of the proposed sensor fusion approach was validated in simulation by considering a quadrotor equipped with low-cost sensors and navigating within the rows of a vineyard in the Piedmont region of Italy.

The remainder of the paper is structured as follows. In Section 2, we present the selected framework and the modelling setup used to describe the main elements involved in the selected scenario. The proposed navigation scheme is detailed in Section 3, where the main features of the 3D distance filter are outlined together with some hints for the classic Kalman filtering scheme. The preliminary numerical results are presented in Section 4 and discussed in Section 5, while the main conclusions are drawn in Section 6.

2. Selected Framework

In this section, we provide an overview of the models exploited to describe the main elements involved in the selected navigation framework: (i) the quadrotor UAV; (ii) the LCG maps; and (iii) the ultrasound sensors.

2.1. Rotary-Wing UAV Modelling

The kinematics and dynamics of a rotary-wing UAV can be defined with respect to two main reference frames, as depicted in Figure 1: (i) an inertial frame $\mathcal{I} = (\hat{i}_{\mathcal{I}}, \hat{j}_{\mathcal{I}}, \hat{k}_{\mathcal{I}})$, with the z-axis pointing upward; and (ii) a moving (body) frame $\mathcal{B} = (\hat{i}_{\mathcal{B}}, \hat{j}_{\mathcal{B}}, \hat{k}_{\mathcal{B}})$, attached to the UGV with oriented axes, in which we envision the so-called “+” configuration, with the body axes aligned with the quadrotor arms. In this configuration, we have a pair of rotors (1 and 3) spinning counter-clockwise, whereas the other pair (2 and 4) spins clockwise with the angular speed of the i -th rotor denoted as ω_i , with $i \in \mathbb{N}_1^4$.

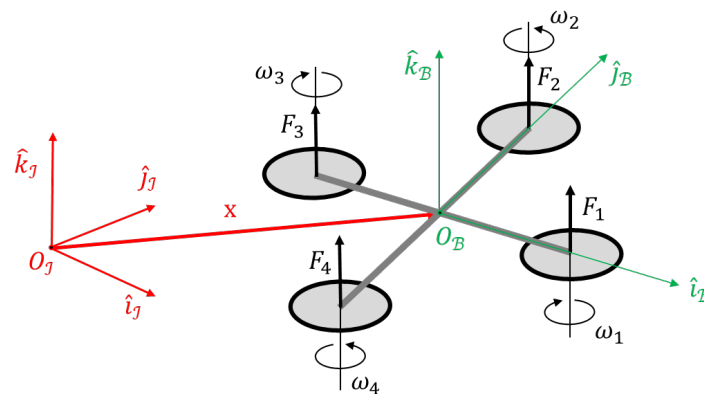


Figure 1. Quadrotor dynamics reference frames: inertial frame (red) and body frame (green).

The Euler equations are used to describe the quadrotor rotational dynamics, where the torques generated by the rotors are defined in the body frame as $\tau = [\tau_\phi, \tau_\theta, \tau_\psi]^\top$, and the contribution of the rotating blades is defined by the rotor moment of inertia I_r and the hover rotational speed Ω_r . Thus, neglecting the aerodynamics effects, the rotational dynamic model is given by:

$$I\dot{\omega}_{\mathcal{B}} = \tau - \omega_{\mathcal{B}} \times (I\omega_{\mathcal{B}} + I_r\Omega_r), \tag{1}$$

where $\omega_{\mathcal{B}} = [p, q, r]^\top$ is the UAV angular velocity expressed in the body frame and I its inertial tensor.

For the kinematic model, we exploited the quaternion-based formulation, thus avoiding singularity issues, i.e.:

$$\dot{q} = \frac{1}{2}\Sigma(q)\omega_{\mathcal{B}}, \quad \Sigma(q) = \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_s & -q_3 & q_2 \\ q_3 & q_s & -q_1 \\ -q_2 & q_1 & q_s \end{bmatrix} \tag{2}$$

where $[q_1, q_2, q_3]^\top$ are the vectorial components of the quaternion q while q_s is the scalar one.

Then, to model the translational motion of the UAV in the inertial frame $\mathbf{p}_{\mathcal{I}} = [x_{\mathcal{I}}, y_{\mathcal{I}}, z_{\mathcal{I}}]^{\top}$, we used the classical Newton's equations, expressed in the inertial frame as:

$$\ddot{\mathbf{p}}_{\mathcal{I}} = \frac{1}{m} \mathbf{\Xi}^{\top} T_B + \mathbf{g}, \quad (3)$$

where $\mathbf{\Xi}$ is the 3 – 1 – 2 inertial-to-body rotation matrix, m is the UAV mass, $T_B = [0, 0, F_z]^{\top}$ is the thrust vector expressed in the body frame, and $\mathbf{g} = [0, 0, -g]^{\top}$ is the gravitational acceleration defined in the inertial frame.

In terms of the generated thrust and torque, each rotor is able to generate a vertical force F_i and a moment M_i as:

$$F_i = k_F \omega_i^2, \quad M_i = k_M \omega_i^2, \quad (4)$$

where the proportional constants k_F and k_M can be determined by experimentation or by simulation (see, e.g., [38]). Then, it is possible to relate the control inputs, i.e., F_z , τ_{ϕ} , τ_{θ} , τ_{ψ} , to the quadrotor characteristic length L , the rotors' angular velocities ω_i , and the thrust F_i , knowing that:

- the thrust command F_z is the sum of the thrust contributions F_i generated by each rotor, i.e.,

$$F_z = \sum_{i=1}^4 F_i;$$

- the rolling torque τ_{ϕ} is produced by a different angular velocity variation of the rotors 2 and 4, i.e.,

$$\tau_{\phi} = L(F_2 - F_4) = k_F L(\omega_2^2 - \omega_4^2);$$

- the pitching torque τ_{θ} is produced instead by a different angular velocity variation of the rotors 1 and 3, i.e.,

$$\tau_{\theta} = L(F_3 - F_1) = k_F L(\omega_3^2 - \omega_1^2);$$

- the yawing torque τ_{ψ} derives from the drag generated by the propellers on the quadrotor itself, with a torque direction opposite to the one of the rotors' motion, such that:

$$\tau_{\psi} = \frac{k_M}{k_F} (F_1 - F_2 + F_3 - F_4) = \frac{k_M}{k_F} (\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2).$$

Further details can be found in [39].

2.2. Vine Row Modelling from LCG Maps

Contrary to the 2D case in [37], in which maps were processed offline to derive the single cutting plane, in this paper, we use the LCG maps in real time within the navigation loop of a quadrotor. We first assume that we can retrieve a 3D plane, defined as $a_T x + b_T y + c_T z + d_T = 0$, that approximates the terrain directly from the LCG map defined with respect to the body frame. Then, at each time step k , we need to properly roto-translate this plane to intersect the 3D mesh. In this way, we can obtain the 2D reference slice that will be exploited in the ellipsoid method, as described in Section 3.1 (see also Figure 2).

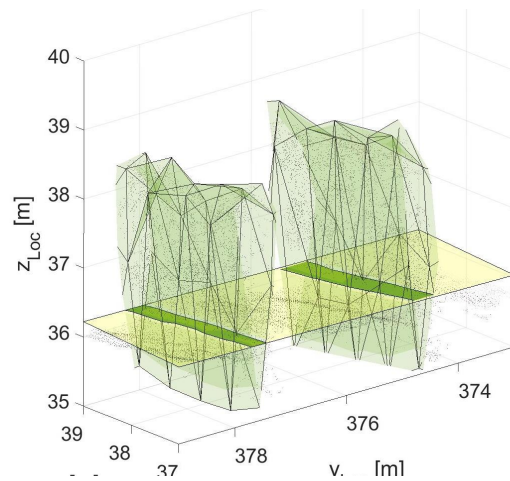


Figure 2. Example of roto-translated approximating plane (yellow plane), intersecting the 3D LCG map and generating a 2D slice (green polyhedron).

The required roto-translation depends on the UAV relative flight altitude with respect to the (approximated) terrain, i.e., z_k , as well as on its attitude. Assuming that, at time k , the UAV orientation with respect to the local/inertial frame is defined by the Euler angles $(\phi_k, \theta_k, \psi_k)$, we can obtain the normal vector to the new plane by a rotation Ξ_k , as:

$$n_k = \begin{bmatrix} a_k \\ b_k \\ c_k \end{bmatrix} \doteq \Xi_{k-1} n_T = [\Theta_k][\Phi_k][\Psi_k] \begin{bmatrix} a_T \\ b_T \\ c_T \end{bmatrix} \quad (5)$$

where $[\Phi_k]$, $[\Theta_k]$, and $[\Psi_k]$ are the rotation matrices, and $n_T = [a_T, b_T, c_T]^\top$ is the normal to the terrain approximating plane. Then, we translate this new plane by the vector $[x_k, y_k, z_k]^\top$, representing the UAV location at time k , in order to make it pass through the quadrotor CoM. In this way, we can identify the last parameter d_k to define the new plane at time k as:

$$a_k x + b_k y + c_k z + d_k = 0; \quad (6)$$

by intersecting this approximating plane with the 3D mesh, the resulting 2D slice (the dark green section in Figure 2) for the i -th row turns out to be composed by N_i vertices and the $N_i - 1$ segments connecting them. To model these slices, and to later use them in the ellipsoid method, we need to retrieve, for each pair of vertices (N_i, N_{i+1}) , the approximating line equation of the form $a_i x + b_i y + c_i = 0$ as follows:

1. Once the vertices (N_i, N_{i+1}) as $N_i = [x_i, y_i]^\top$ and $N_{i+1} = [x_{i+1}, y_{i+1}]^\top$ are defined in the body frame, we have:

$$\frac{x - x_i}{x_{i+1} - x_i} = \frac{y - y_i}{y_{i+1} - y_i};$$

2. We retrieve the coefficients a_i , b_i , and c_i as:

$$a_i = y_{i+1} - y_i, \quad b_i = x_i - x_{i+1}, \quad c_i = x_{i+1}y_i - x_i y_{i+1};$$

3. Once the straight line passing through the vertices (N_i, N_{i+1}) is identified, we compute its characteristic slope β_i , which will later be used in the ellipsoid method, as:

$$\beta_i = \arctan\left(-\frac{a_i}{b_i}\right).$$

2.3. Ultrasound Sensor Modelling

Ultrasonic sensors are based on the measured propagation time of an ultrasonic signal. They emit high-frequency sound waves which are reflected by an object, including transparent and other demanding objects where optical technologies fail. In particular, the distance measurement is based on a time-of-flight measurement. The sensor calculates the time between the sending and receiving of the reflected sound signal. In this work, we assume that the UAV is equipped with five ultrasonic sensors, one on each quadrotor arm, to measure the distance in the x–y-body plane with respect to the vine rows, and one at the bottom of the UAV, for measuring the relative flight altitude with respect to the terrain.

In particular, we assume that the US sensor returns a distance measurement d given by the sum of two contributions, i.e.:

$$d = d^\perp + e^d, \quad (7)$$

where e^d is an unknown-but-bounded error defined in the range $e^d \in [e^d, \bar{e}^d]$, whereas d^\perp is the distance measured from the UAV CoM to the LCG map on the intercepted 2D slice, as depicted in Figure 3.

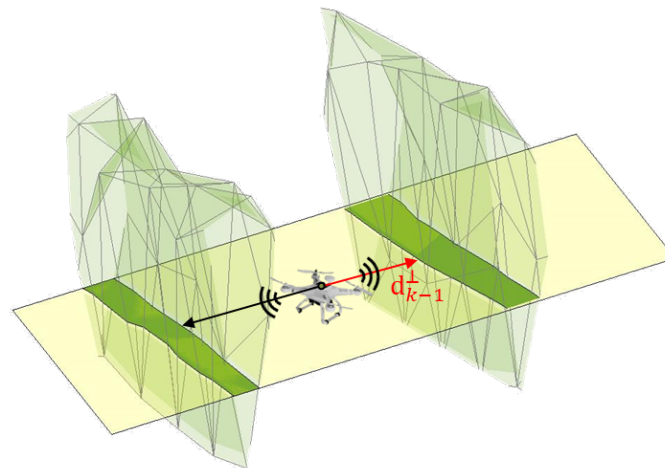


Figure 3. Ultrasound measurement d_{k-1}^\perp with respect to the 2D slice (green area) obtained by intersecting the 3D LCG map with the 2D (yellow) plane, passing through the UAV CoM.

It is important to remark that e^d is defined as $e^d = e^s + e^m$, where e^s is the ultrasound sensor accuracy error, i.e., $e^s \in [e^s, \bar{e}^s]$, and e^m is the LCG approximation error, i.e., $e^m \in [e^m, \bar{e}^m]$. Consequently, we have that $d \in [d, \bar{d}]$ and

$$d = d^r + e^s, \quad d^r = d^\perp + e^m, \quad (8)$$

where d^r is the real distance from the quadrotor CoM and the vine row.

The approximation error e^m takes into account the error introduced by the use of the map, which provides only an approximation of the crop location and shape. On the other hand, the UAV–LCG distance d^\perp represents the input to the row selection sub-phase, together with the information arising from the map and the position $p_{k|k-1}$ and attitude $\theta_{k|k-1}$ predicted at time k , given the measurement at time $k - 1$, as depicted in Figure 4 and described in detail in Section 3.1.

3. 3D Navigation Filtering Scheme

The work logic of the proposed sensor fusion approach is depicted in Figure 4, from which it is possible to recognise the classic two-phase approach of a Kalman-like filtering scheme (prediction and update), in between the so-called 3D distance filter, enclosed into the green box. Given the estimate of the state \hat{x}_{k-1} and the error covariance

matrix P_{k-1} at time $k-1$, we can obtain the state and covariance projection, i.e., \tilde{x}_k and \tilde{P}_k , at time k during the Kalman filter prediction phase. In the specific case, the state vector includes the UAV position $\tilde{p}_k = [\tilde{x}_k, \tilde{y}_k, \tilde{z}_k]^\top$ and attitude $(\tilde{\phi}_k, \tilde{\theta}_k, \tilde{\psi}_k)$. The heading angle $\tilde{\theta}_k$ and the flight altitude \tilde{z}_k represents the input to the 2D map update block, allowing for the definition of the intersecting plane and the 2D slice, as described in Section 2.2. Together with the UAV position \tilde{p}_k , the heading $\tilde{\theta}_k$, the measured distance d_k^\perp , and the estimation error e^d , the 2D slice represent the inputs to the row selection phase. This block leads to the identification of the reference segment for the calculation of the parallel cuts, defined by the parameters $a_k^O, \beta_k, \hat{\beta}_k$. The last step of the 3D distance filter allows for the propagation of the uncertain ellipsoid defined by \tilde{x}_k and \tilde{P}_k , obtaining the minimum volume ellipsoid \mathcal{E}'_k that contains the feasible point set defined by its center x'_k and covariance matrix P'_k . These data, together with those provided at time k by the GPS and IMU, i.e., (x_k^{GPS}, P_k^{GPS}) and (x_k^{IMU}, P_k^{IMU}) , respectively, represent the inputs to the last phase of the Kalman filter, i.e., the update phase. The main outputs are the estimate of the state \hat{x}_k and the error covariance matrix P_k at time k . Each block is detailed in the following sections, starting from the innovative part of this approach, i.e., the 3D distance filter.

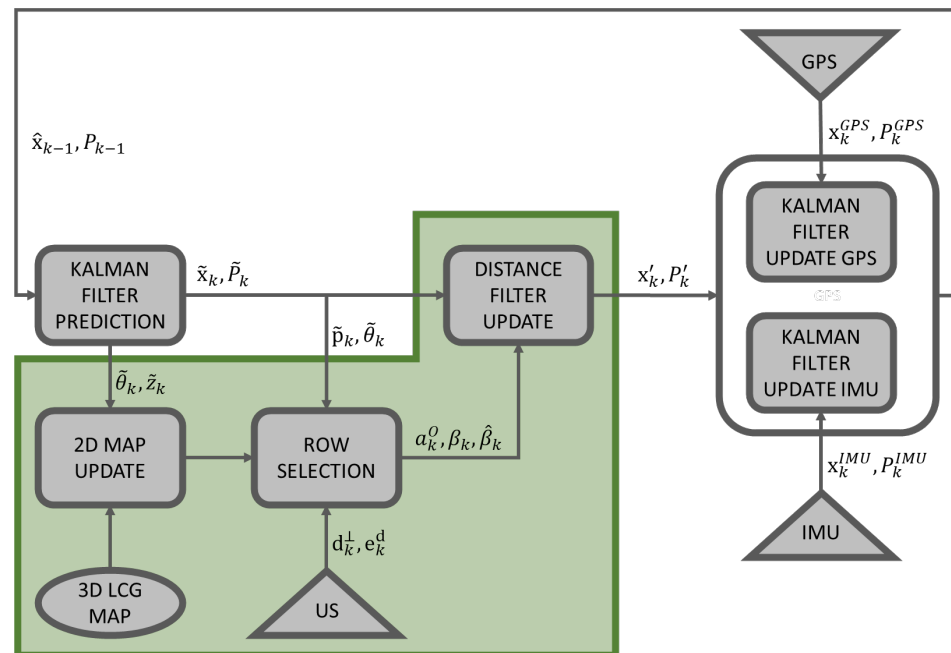


Figure 4. Work logic of the proposed sensor fusion strategy, in which the 3D distance filter is enclosed in the green area. In this scheme, the sensors' data are represented by triangular blocks, while the LCG maps are represented by an ellipsoidal block.

3.1. 3D Distance Filter

As represented in Figure 4, the 3D distance filter can be split into three main sub-phases: (i) the 2D map update, which includes the modelling of the vine row starting from the LCG maps, as described in Section 2.2; (ii) the row selection, a moving window approach that allows for the identification of the row section of interest, given the perceived UAV location (in this way, the number of segments of the 2D slice, among which the one complying with the measured distance is identified, is reduced, together with the computational burden of the algorithm); and (iii) the final distance filter update, which is based on the parallel-cuts ellipsoid method [40], first proposed in [37] as filtering scheme.

At each time step k , we first determine the reference plane $a_k x + b_k y + c_k z + d_k = 0$ following the approach described in Section 2.2, whose results coincide with the UAV body xy -plane. In this framework, the only relevant attitude angle is represented by the heading

(pitch) angle θ_k . Then, we assume that at time k , the quadrotor location (x_k, y_k) is affected by uncertainty, represented by an ellipsoid \mathcal{E}_k defined as

$$\mathcal{E}_k = \left\{ \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^2 \mid \left\| \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x_k \\ y_k \end{bmatrix} \right\|_{P_k^{-1}}^2 \leq 1 \right\} \tag{9}$$

where P_k is the ellipsoid shape matrix. Then, we collect the distance measurement d_k from the UAV and the row, and we select a subset of segments composing the 2D slice that are in the proximity of the UAV location. Then, among this subset, we identify the segment i -th whose distance from the UAV adheres more with the measured one d_k^\perp . Last, we retrieve the corresponding approximating equation $a_i x + b_i y + c_i = 0$.

Because all these measures are affected by errors, this implies that there are multiple points inside the uncertainty ellipsoid complying with the ultrasound measure. Hence, it is possible to define a lower and upper bound on the offset from the i -th approximating line to limit the points satisfying this constraint. In particular, given the UAV orientation θ_k , the line slope β_i , and the bounds on the sensor measurement, i.e., \underline{d} and \bar{d} , we can define the upper and lower offsets as

$$\bar{d}_k^O = \frac{\cos(\theta_k - \beta_i)}{\cos(\pi/2 - \beta_i)} \bar{d}, \quad d_k^O = \frac{\cos(\theta_k - \beta_i)}{\cos(\pi/2 - \beta_i)} d. \tag{10}$$

These offsets identify two straight-lines, parallel to the one representing the i -the segment and described as

$$-(a_k^O)^\top x \leq \hat{\beta}_k, \quad (a_k^O)^\top x \leq \beta_k, \tag{11}$$

where

$$a_k^O = -\frac{a_i}{b_i}, \quad \hat{\beta}_k = -\frac{c_i + a_i \bar{d}_k^O}{b_i}, \quad \beta_k = -\frac{c_i + a_i d_k^O}{b_i}. \tag{12}$$

The area of the uncertainty ellipsoid \mathcal{E}_k constrained between these two lines is the so-called feasible point set \mathcal{F} , defined as

$$\mathcal{F} \doteq \left\{ \begin{bmatrix} x \\ y \end{bmatrix} \in \mathcal{E}_k \mid \underline{d} \leq d\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) \leq \bar{d} \right\}, \tag{13}$$

and represented by the yellow area in Figure 5.

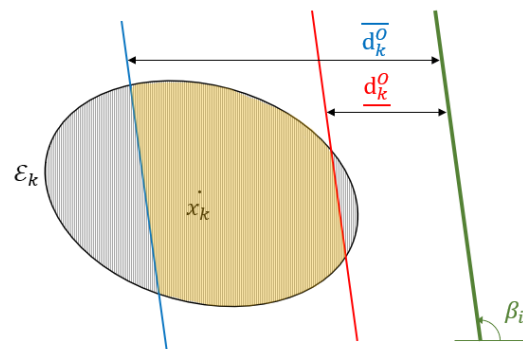


Figure 5. Identification of lower- d_k^O and upper- \bar{d}_k^O offsets, defining the parallel cuts (red and blue lines) with respect to the approximating plane (green line) with orientation β_i , thus identifying the section of the uncertain ellipsoid (yellow area) whose distance complies with the measured one.

The parallel-cuts ellipsoid method, described in [40], allows the simultaneously use of the constraints imposed by the pair of parallel cuts to generate the new ellipsoid \mathcal{E}'_k having minimum volume and containing all the points between the two half-spaces (see

Figure 6). Hence, let us compute the algebraic distance of each half-plane from the center of the ellipse (x_k, y_k) as

$$\hat{\alpha} = \frac{\hat{\beta} - a^T x_k}{\sqrt{a^T P_k a}}, \quad \alpha = \frac{a^T x_k - \beta}{\sqrt{a^T P_k a}}. \tag{14}$$

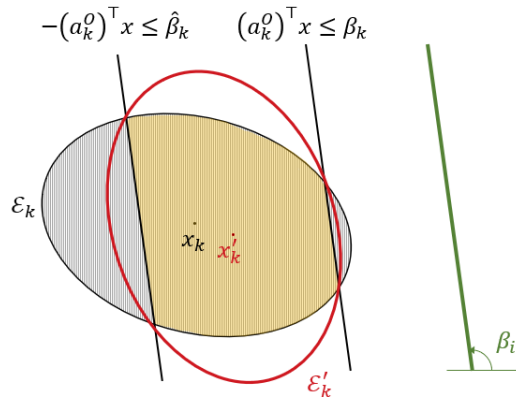


Figure 6. Propagation of the uncertainty ellipsoid \mathcal{E}_k at time $k + 1$, i.e., \mathcal{E}'_k , using the parallel-cuts method. \mathcal{E}'_k represents the ellipsoid at minimum volume containing the feasible point set (yellow area).

Then, we can compute the propagation of the ellipsoid \mathcal{E}'_k (see Figure 6) by its center x'_k and shape matrix P'_k as:

$$x'_k = x_k - \tau \frac{P_k a}{\sqrt{a^T P_k a}}, \quad P'_k = \delta \left(P_k - \sigma \frac{P_k a (P_k a)^T}{\sqrt{a^T P_k a}} \right), \tag{15}$$

where n is the space dimension, and σ , τ , and δ are the dilation, step, and expansion parameters, respectively. For the parallel-cuts approach, these parameters can be computed as [40]:

$$\sigma = \frac{1}{n + 1} \left[n + \frac{2}{(\alpha - \hat{\alpha})^2} \left(1 - \alpha \hat{\alpha} - \frac{\rho}{2} \right) \right], \quad \tau = \frac{\alpha - \hat{\alpha}}{2} \sigma, \quad \delta = \frac{n^2}{n^2 - 1} \left(1 - \frac{\alpha^2 + \hat{\alpha}^2 - \frac{\rho}{n}}{2} \right),$$

where $\rho = \sqrt{4(1 - \alpha^2)(1 - \hat{\alpha}^2) + n^2(\hat{\alpha}^2 - \alpha^2)^2}$.

3.2. Kalman Filter for Sensor Fusion

Since the early 1960s, much effort has been devoted to modelling processes $\{y_k\}$ in the state-space form, i.e.:

$$z_k = Hx_k + v_k, \quad k \geq 0, \tag{16}$$

where the state x_k obeys the recursion:

$$x_{k+1} = \Phi x_k + w_k, \quad k \geq 0. \tag{17}$$

The measurement noise v_k and the process noise w_k are assumed to be vector-valued zero-mean white-noise processes with known covariance, i.e., Q and R , respectively. On the other hand, the initial state x_0 is assumed to have zero-mean covariance matrix Π_0 , and to be uncorrelated with $\{v_k\}$ and $\{w_k\}$. We also assume that the matrices Φ , H , Q , R , S , and Π_0 are known a priori, following the classical assumptions of the Kalman filter (see also [41]). Then, we can define the estimation error e_k as the difference between x_k and its estimate \hat{x}_k , and the error covariance matrix P_k can be defined as:

$$P_k \doteq \mathbb{E}\{(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T\}. \tag{18}$$

Assuming that \tilde{x}_k is the prior estimate of \hat{x}_k , the new estimate can be updated by combining the old estimate with the measured data as:

$$\hat{x}_k = \tilde{x}_k + K_k(z_k - H\tilde{x}_k), \quad (19)$$

where K_k is the Kalman gain defined by:

$$K_k \doteq \tilde{P}_k H^\top \left(H \tilde{P}_k H^\top + R \right)^{-1}, \quad (20)$$

and \tilde{P}_k is the prior estimate of P_k , which can be updated as:

$$P_k = (\mathbb{I} - K_k H) \tilde{P}_k. \quad (21)$$

Then, the state and covariance projection can be achieved as follows:

$$\tilde{x}_{k+1} = \Phi \hat{x}_k, \quad \tilde{P}_{k+1} = \Phi P_k \Phi^\top + Q. \quad (22)$$

This classical filtering approach was used in the proposed navigation scheme to fuse the data coming from the sensors mounted on the UAV, as depicted in Figure 4.

4. Results

The proposed case study involves a DJI Matrice 100 (M-100) quadrotor that shall autonomously navigate between the vine rows of a vineyard located in Barolo (Piedmont, Italy). The selected field extends over a surface of about 0.7 hectares, with an elevation that ranges from 460-m to 490-m above sea level. The space between the vine plants and the inter-row space are about 0.9 m and 2.5 m, respectively. As is typical for precision agricultural applications involving drones [42,43], the UAV is equipped with one GPS module (CP.TP.000025 [44]), one IMU sensor (Vectornav VN-200 [45]), and ultrasound sensors (HC-SR04 [46]). The main features of the drone and of the equipped sensors are reported in Tables 1 and 2, respectively.

Table 1. List of DJI Matrice 100 geometrical parameters.

Parameter	Value	Parameter	Value
Diagonal Wheelbase	650 (mm)	x -axis inertia I_x	0.0617 (kg m ²)
Weight	2431 (g)	y -axis inertia I_y	0.0619 (kg m ²)
Max. Takeoff Weight	3600 (g)	z -axis inertia I_z	0.1231 (kg m ²)
Nominal rotor rate	580 (rad/s)	Rotor inertia I_r	0.001 (kg m ²)

Table 2. Features of the sensors equipped onboard the UAV.

Sensor	Parameter	Value
Ultrasound	Distance range	0.02–4.00 (m)
	Accuracy	3 (mm)
GPS	Position Accuracy (1σ)	1 (m)
	Horizontal position accuracy (1σ)	1 (m)
IMU	Vertical position accuracy (1σ)	1.5 (m)
	Velocity accuracy	0.05 (m/s)
	ϕ/ψ range	0–180 (deg)
	θ range	0–90 (deg)
	ϕ/θ accuracy (1σ)	0.03 (deg)
	ψ accuracy (1σ)	0.2 (deg)

The 3D LCG map was obtained from a remote sensing mission, as detailed in [35]. The validation of the proposed navigation filtering scheme was carried out by exploiting a simulator developed with MATLAB/Simulink 2019b, running over an Intel Core i7-6500U with a CPU @2.50 GHz, a RAM of 12 GB, and a 1-TB hard-disk drive. The presence of several sensors required the development of a multi-rate simulator, where a characteristic sample rate was defined for each sensor according to the classical settings (see, e.g., [47–49]). In particular, we have: (i) 5 Hz for the GPS; (ii) 100 Hz for the IMU; and (iii) 20 Hz for the ultrasound sensors. The different output data rates made it necessary to implement an ad hoc strategy for the data synchronisation. In particular, we followed the approach proposed in [50], where the data provided by the slower sensors (i.e., GPS and ultrasonic) are fused with the data from the faster sensor (IMU) only when available. This means that the global sample time coincides with the fastest one, i.e., 0.01 s. The opposite approach, proposed in [51], where the faster measurement data is adapted to the slower one, was discarded because it might cause a lower accuracy of the final estimation. The aforementioned simulator is based on the linearised dynamics of the UAV, obtained by linearising the nonlinear dynamics model presented in Section 2.1 with respect to the hovering condition. For the results presented below, we set the UAV initial position at $[36, 376, 37.5]^T$ m with respect to the local frame, with an airspeed of $[10, 2.9, 0.3]^T$, null angular velocity, and an attitude of $[0, \frac{\pi}{90}, -\frac{\pi}{6}]^T$, placing the UAV at the beginning of the first inter-row, at non-zero speed, and pointing toward its end.

Figure 7 provides an overview of the simulation framework where it is possible to observe the LCG map and the evolution of the UAV location (black circles) and the corresponding 2D intersecting planes (yellow rectangles) and 2D slices (green polyhedrons). The motion of the UAV was perturbed by a uniform, zero-mean random noise affecting both the position and attitude with a standard deviation of 0.1 m and 10 degrees, respectively.

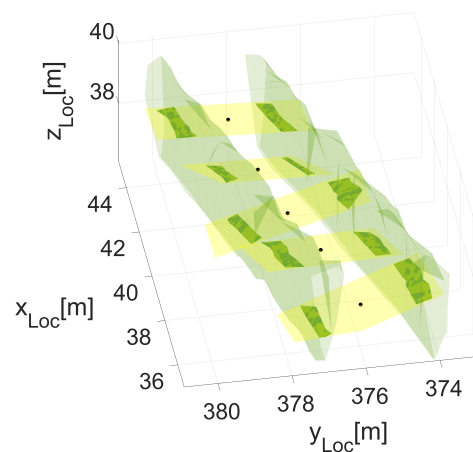


Figure 7. Evolution of the quadrotor center-of-mass position (black circles) within the LCG maps together with the approximating planes (yellow rectangles) and the generated 2D slices (dark green areas).

Before starting the validation of the navigation scheme, it was crucial to determine the optimal sensors configuration and to identify the minimum number of ultrasound sensors to be mounted onboard the UAV in order to obtain the desired estimation accuracy level. Figure 8 reports the results in terms of the lateral (E_{lat}^i) and longitudinal (E_{long}^i) estimation errors obtained by comparing the three different configurations: (i) with $i = 1$ distance sensor mounted at 45 degrees, with respect to the main axis, pointing towards the right vine row; (ii) with $i = 2$ sensors, one on each side of the UAV; and (iii) with $i = 4$ ultrasound sensors, each one mounted at the end of the UAV arms. In Figure 8, we also reported the performance obtained without the use of the LCG map (black lines) to highlight the benefits achievable with the proposed navigation scheme.

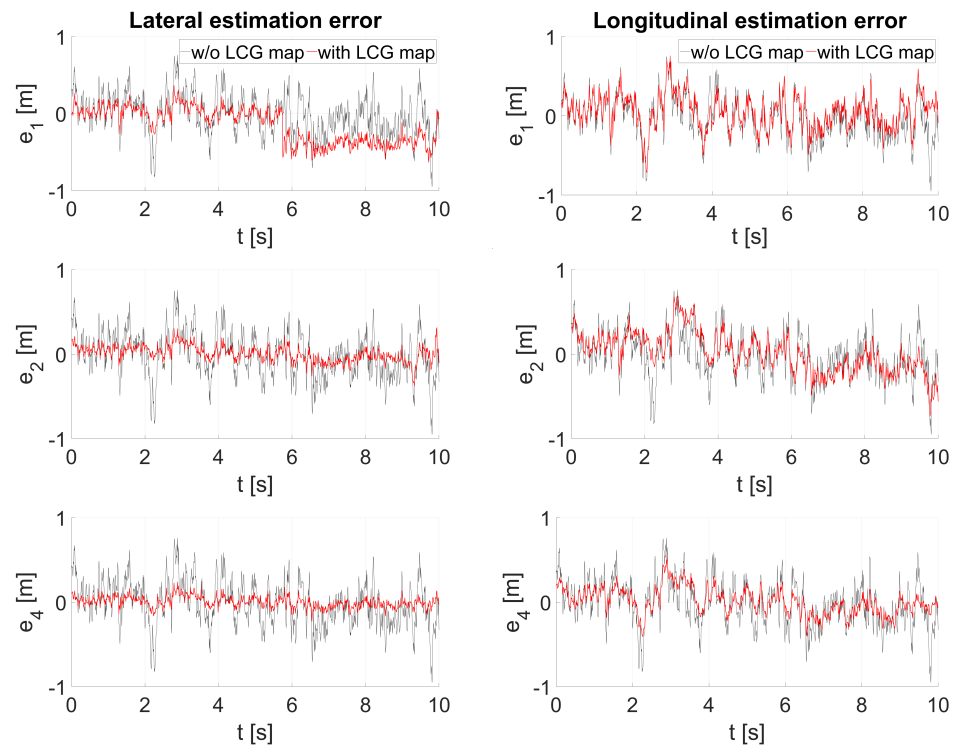


Figure 8. Time evolution of the lateral (**left column**) and longitudinal (**right column**) estimation errors for different ultrasound sensor configurations. The subscript in the vertical axis label, i.e., e_i , with $i = 1, 2, 4$, identifies the number of ultrasound sensors equipped onboard. In this figure, the estimation error obtained using only GPS and IMU (black lines) is compared with the one obtained including data from ultrasound sensors and LCG maps (red line).

Once the optimal sensor configuration has been assessed, we proceeded with the navigation filter validation in the aforementioned simulation environment, of which the results are depicted in Figure 9. In particular, we compared the evolution of the UAV's true position (yellow circles) and the corresponding confidence ellipses, i.e., representing all the possible positions where the vehicle could be in a given time instant according to the covariance matrix relative to the filter estimation, when the GPS and IMU data were fused (red ellipse) or not fused (black ellipse) with the information provided by the distance filter (which includes the LCG map in the estimation process).

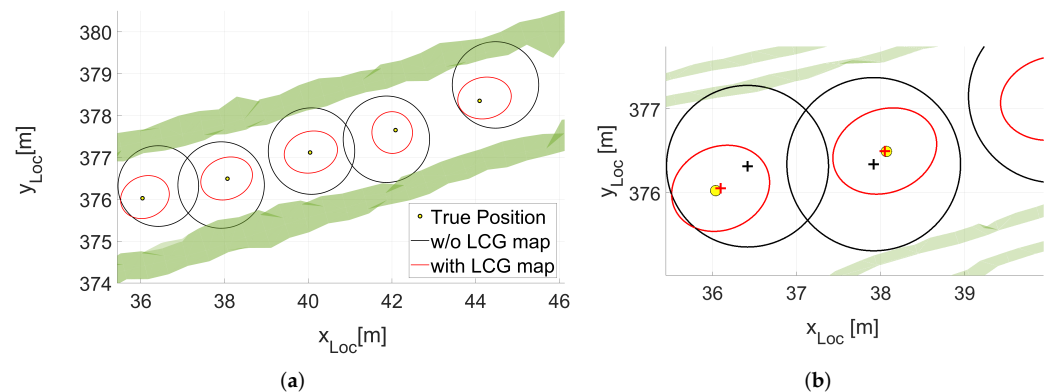


Figure 9. In (a), the UAV real position evolution (yellow circles) within vine rows and corresponding uncertainty ellipsoids when the LCG maps and ultrasounds are (red lines) or not (black lines) included into the navigation filter. In (b), it is possible to observe that the true position is always inside the uncertainty ellipsoids, but it is almost aligned with the red ellipses' centers.

A statistical analysis of the estimation error was performed. As shown in Figure 10, we retrieved the normal distribution of the lateral and longitudinal position errors achievable when the LCG maps are exploited (red bars) or not exploited (blue bars). The obtained standard deviations are reported in Table 3.

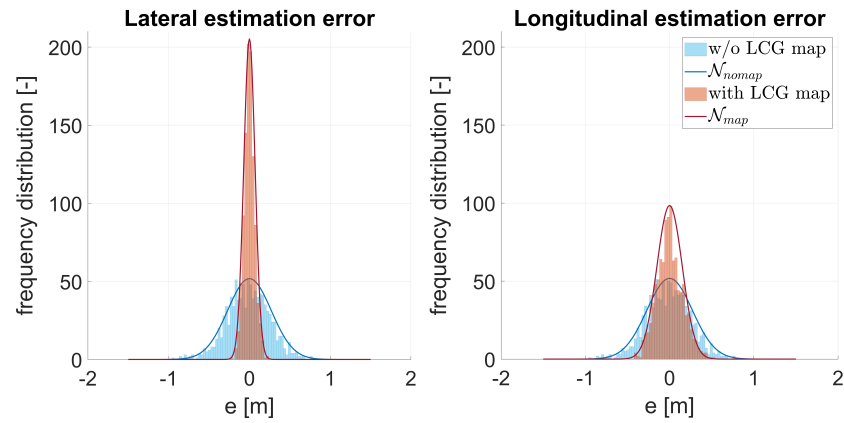


Figure 10. Frequency distribution of the lateral (left histogram) and longitudinal (right histogram) estimation error, comparing the results when the maps are combined with the onboard sensors (red bars) or when they are not exploited (blue bars).

Table 3. Comparison of the 1σ standard deviations for the lateral and longitudinal estimation errors when the LCG maps data are included or not included.

Configuration	1σ Lateral Value	1σ Longitudinal Value
with LCG map	0.0632 (m)	0.1529 (m)
w/o LCG map	0.2623 (m)	0.2616 (m)

Last, we checked the computational compatibility of the proposed navigation scheme, measuring the computational demand for each phase of the filter and for the entire navigation block, as represented in Figure 11.

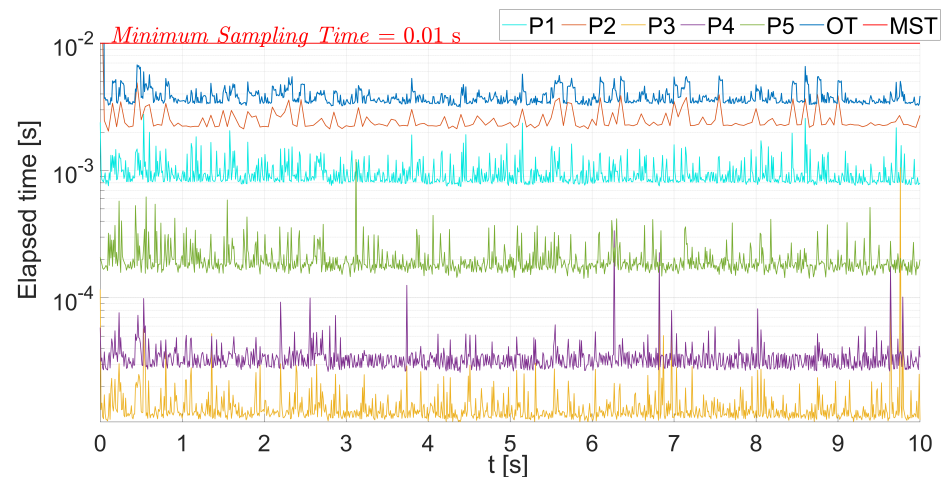


Figure 11. Elapsed time for each phase of the proposed filter, compared to the minimum sampling time (MST). In particular, we have: P1 = prediction phase, P2 = 3D/2D map conversion phase, P3 = row selection phase, P4 = distance filter phase, P5 = update phase, OT = overall Time.

5. Discussion

Regarding the study on the identification of the optimal number of sensors to be installed onboard the quadrotor, the obtained results confirmed that better performance can be achieved by increasing the number of equipped ultrasound sensors, up to a certain level. Above a given number of sensors, the achievable benefits do not balance the additional weight to be carried onboard. In detail, when only one distance sensor is employed, the accuracy of the new estimated position is generally improved with respect to the case when only the GPS and IMU data are used for position estimation. In particular, an average reduction close to 47% was obtained for the lateral estimation error, versus 35% for the longitudinal one. Adding a second ultrasound sensor to gather data from both vine rows allowed for the further improvement of those percentages up to 64% for the lateral estimation and 38% for the longitudinal one. Better results were obtained with the four-sensor configuration, with a reduction of the average estimation error of 72% for the lateral position and 49% for the longitudinal one. Further analyses were made by assuming double the number of ultrasonic sensors for each arm of the UAV. The results showed that the obtained improvements, i.e., an average reduction of 74% for the lateral estimation error and 55% for the longitudinal one, were not balancing the undesired additional computation effort and system complexity. In addition, as shown in Figure 9, the comparison between the prediction and the uncertainty propagation, obtained by including (red ellipses) or not including (black ellipses) the information provided by the ultrasonic sensors and the LCG map highlighted the benefits achievable with the proposed filtering approach. The algorithm's effectiveness was also supported by the obtained value of the standard deviation reduction when exploiting the proposed distance filter. Indeed, the standard deviation reduction achieved was close to 76% for the lateral error estimation and to 42% for the longitudinal one.

According to the presented results, it was possible to observe considerable improvements achievable by the exploitation of the proposed 3D distance filter, where the LCG maps were used to improve the quality of the location estimation, compared to a classical filtering scheme involving only the GPS and IMU sensors. In particular, it was possible to significantly reduce the average estimation error and standard deviation for the UAV lateral and longitudinal locations (see Figure 10). Specifically, it was possible to observe a higher accuracy for the lateral estimation, with respect to the longitudinal one, due to the more accurate information provided by the 3D distance filter and the maps in the lateral direction. These results are compatible with the need to avoid collision with the vine rows during the UAV autonomous navigation.

Last, the analysis of the computational costs required by each phase of the filtering scheme shows that the additional computational cost introduced by the 3D distance filter is compatible with the navigation time constraint. Indeed, to guarantee that all data provided by the sensors are properly processed and filtered in order to estimate, at each time step, the UAV location and attitude, the computation time required to solve the navigation task shall be lower than the navigation sample time, i.e., it has to be lower than 0.01 s. When this condition is not satisfied, some data are lost, compromising the navigation task. The obtained results, presented in Figure 11, showed that the most computationally demanding phase was related to the 3D/2D map conversion, whereas the distance filter update and the row selection phase proved to be the fastest, with computation times lower than 10^{-4} seconds. Overall, the total computation time measured during the validation, with an average value of about 0.005 s, demonstrated that the proposed filtering scheme is suitable for real-time scenarios.

6. Conclusions

In this paper, we presented an improved UAV localisation system to allow for the autonomous navigation of rotary-wing UAVs within vine rows, in order to perform scouting and crop-monitoring operations. The proposed method extends the ellipsoid method-based filtering approach to the 3D case. In particular, we proved the efficacy of the proposed 3D

sensor fusion approach in simulation. The improvements on the estimation of the UAV location when operated within the crops, despite the adoption of COTS sensors, are promising to the point that experimental validation campaigns have already been planned as the next research step. In particular, the first phase involves the realisation of the 3D LCG maps of the experimental testbed, i.e., the vineyard located inside the campus of the University of Turin (DiSAFA facilities). Then, once the testing UAV is customised and calibrated, mounting the selected sensors suite and uploading the proposed navigation algorithm, an extensive experimental campaign will be carried out to assess the autonomous navigation capabilities achievable in a real scenario. Last, additional flight tests will be carried out in a vineyard in Barolo (Piedmont, Italy) to compare the simulation and experimental results.

Author Contributions: Data curation, L.C. and A.B.; Funding acquisition, P.G. and F.D.; Methodology, C.D., M.M., L.C. and A.B.; Project Administration, M.M., L.C., P.G. and F.D.; Software, C.D.; Supervision, P.G. and F.D.; Writing—original draft, C.D., M.M., L.C. and A.B.; Writing—review & editing, P.G. and F.D. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Funding: This work was partially funded by the Italian IIT and MIUR within the project “New technical and operative solutions for the use of drones in Agriculture 4.0” (PRIN 2017, Prot. 2017S559BB).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Elkaim, G.H.; Lie, F.A.P.; Gebre-Egziabher, D. Principles of guidance, navigation, and control of UAVs. In *Handbook of Unmanned Aerial Vehicles*; Springer: Dordrecht, The Netherlands, 2015; pp. 347–380. [\[CrossRef\]](#)
2. Yao, H.; Qin, R.; Chen, X. Unmanned aerial vehicle for remote sensing applications—A review. *Remote Sens.* **2019**, *11*, 1443. [\[CrossRef\]](#)
3. Alatisse, M.B.; Hancke, G.P. A review on challenges of autonomous mobile robot and sensor fusion methods. *IEEE Access* **2020**, *8*, 39830–39846. [\[CrossRef\]](#)
4. Schirrmann, M.; Hamdorf, A.; Giebel, A.; Gleiniger, F.; Pflanz, M.; Dammer, K.H. Regression kriging for improving crop height models fusing ultra-sonic sensing with UAV imagery. *Remote Sens.* **2017**, *9*, 665. [\[CrossRef\]](#)
5. White, F.E. *Data Fusion Lexicon*; Technical Panel For C3; Joint Directors of Laboratories (JDL): Washington, DC, USA, 1991.
6. Castanedo, F. A review of data fusion techniques. *Sci. World J.* **2013**, *2013*, 704504. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45. [\[CrossRef\]](#)
8. De Marina, H.G.; Pereda, F.J.; Giron-Sierra, J.M.; Espinosa, F. UAV attitude estimation using unscented Kalman filter and TRIAD. *IEEE Trans. Ind. Electron.* **2011**, *59*, 4465–4474. [\[CrossRef\]](#)
9. Mao, G.; Drake, S.; Anderson, B.D. Design of an extended Kalman filter for UAV localization. In Proceedings of the 2007 IEEE Information, Decision and Control, Adelaide, Australia, 12–14 February 2007; pp. 224–229. [\[CrossRef\]](#)
10. Arellano-Cruz, L.A.; Galvan-Tejada, G.M.; Lozano-Leal, R. Performance comparison of positioning algorithms for UAV navigation purposes based on estimated distances. In Proceedings of the 2020 IEEE 17th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), Mexico City, Mexico, 11–13 November 2020; pp. 1–8. [\[CrossRef\]](#)
11. Steele, B.M. Maximum posterior probability estimators of map accuracy. *Remote Sens. Environ.* **2005**, *99*, 254–270. [\[CrossRef\]](#)
12. Zhang, J.; Liu, W.; Wu, Y. Novel technique for vision-based UAV navigation. *IEEE Trans. Aerosp. Electron. Syst.* **2011**, *47*, 2731–2741. [\[CrossRef\]](#)
13. Xie, W.; Wang, L.; Bai, B.; Peng, B.; Feng, Z. An improved algorithm based on particle filter for 3D UAV target tracking. In Proceedings of the IEEE International Conference on Communications (ICC 2019), Shanghai, China, 20–24 May 2019; pp. 1–6. [\[CrossRef\]](#)
14. Santos, N.P.; Lobo, V.; Bernardino, A. Unmanned aerial vehicle tracking using a particle filter based approach. In Proceedings of the 2019 IEEE Underwater Technology (UT), Kaohsiung, Taiwan, 16–19 April 2019; pp. 1–10. [\[CrossRef\]](#)
15. Rigatos, G. Distributed particle filtering over sensor networks for autonomous navigation of UAVs. In Proceedings of the 2010 IEEE 72nd Vehicular Technology Conference, Ottawa, ON, Canada, 6–9 September 2010.
16. Won, D.H.; Chun, S.; Sung, S.; Lee, Y.J.; Cho, J.; Joo, J.; Park, J. INS/vSLAM system using distributed particle filter. *Int. J. Control. Autom. Syst.* **2010**, *8*, 1232–1240. [\[CrossRef\]](#)
17. Uhlmann, J.K. Covariance consistency methods for fault-tolerant distributed data fusion. *Inf. Fusion* **2003**, *4*, 201–215. [\[CrossRef\]](#)

18. Gai, J.; Xiang, L.; Tang, L. Using a depth camera for crop row detection and mapping for under-canopy navigation of agricultural robotic vehicle. *Comput. Electron. Agric.* **2021**, *188*, 106301. [[CrossRef](#)]
19. Weiss, U.; Biber, P. Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. *Robot. Auton. Syst.* **2011**, *59*, 265–273. [[CrossRef](#)]
20. Mammarella, M.; Comba, L.; Biglia, A.; Dabbene, F.; Gay, P. Cooperation of unmanned systems for agricultural applications: A theoretical framework. *Biosyst. Eng.* **2021**, *in press*. [[CrossRef](#)]
21. Gené-Mola, J.; Gregorio, E.; Cheein, F.A.; Guevara, J.; Llorens, J.; Sanz-Cortiella, R.; Escolà, A.; Rosell-Polo, J.R. Fruit detection, yield prediction and canopy geometric characterization using LiDAR with forced air flow. *Comput. Electron. Agric.* **2020**, *168*, 105121. [[CrossRef](#)]
22. Chen, Y.; Zhang, B.; Zhou, J.; Wang, K. Real-time 3D unstructured environment reconstruction utilizing VR and Kinect-based immersive teleoperation for agricultural field robots. *Comput. Electron. Agric.* **2020**, *175*, 105579. [[CrossRef](#)]
23. Comba, L.; Biglia, A.; Ricauda Aimonino, D.; Barge, P.; Tortia, C.; Gay, P. 2D and 3D data fusion for crop monitoring in precision agriculture. In Proceedings of the 2019 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor), Portici, Italy, 24–26 October 2019; pp. 62–67. [[CrossRef](#)]
24. Rovira-Más, F.; Zhang, Q.; Reid, J.F. Stereo vision three-dimensional terrain maps for precision agriculture. *Comput. Electron. Agric.* **2008**, *60*, 133–143. [[CrossRef](#)]
25. Kim, W.S.; Lee, D.H.; Kim, Y.J.; Kim, T.; Lee, W.S.; Choi, C.H. Stereo-vision-based crop height estimation for agricultural robots. *Comput. Electron. Agric.* **2021**, *181*, 105937. [[CrossRef](#)]
26. Chen, Y.; Xiong, Y.; Zhang, B.; Zhou, J.; Zhang, Q. 3D point cloud semantic segmentation toward large-scale unstructured agricultural scene classification. *Comput. Electron. Agric.* **2021**, *190*, 106445. [[CrossRef](#)]
27. Comba, L.; Biglia, A.; Ricauda Aimonino, D.; Tortia, C.; Mania, E.; Guidoni, S.; Gay, P. Leaf Area Index evaluation in vineyards using 3D point clouds from UAV imagery. *Precis. Agric.* **2020**, *21*, 881–896. [[CrossRef](#)]
28. Barclay, M.; Galton, A. Comparison of region approximation techniques based on Delaunay triangulations and Voronoi diagrams. *Comput. Environ. Urban Syst.* **2008**, *32*, 261–267. [[CrossRef](#)]
29. Heinzer, T.J.; Williams, M.D.; Dogrul, E.C.; Kadir, T.N.; Brush, C.F.; Chung, F.I. Implementation of a feature-constraint mesh generation algorithm within a GIS. *Comput. Geosci.* **2012**, *49*, 46–52. [[CrossRef](#)]
30. Mebatsion, H.; Verboven, P.; Verlinden, B.; Ho, Q.T.; Nguyen, T.A.; Nicolaï, B. Microscale modelling of fruit tissue using Voronoi tessellations. *Comput. Electron. Agric.* **2006**, *52*, 36–48. [[CrossRef](#)]
31. Yang, Z.; Seo, Y.H.; Kim, T.w. Adaptive triangular-mesh reconstruction by mean-curvature-based refinement from point clouds using a moving parabolic approximation. *Comput.-Aided Des.* **2010**, *42*, 2–17. [[CrossRef](#)]
32. Kallinderis, Y.; Vijayan, P. Adaptive refinement-coarsening scheme for three-dimensional unstructured meshes. *AIAA J.* **1993**, *31*, 1440–1447. [[CrossRef](#)]
33. Almasi, O.N.; Rouhani, M. A geometric-based data reduction approach for large low dimensional datasets: Delaunay triangulation in SVM algorithms. *Mach. Learn. Appl.* **2021**, *4*, 100025. [[CrossRef](#)]
34. Chen, M.; Tang, Y.; Zou, X.; Huang, Z.; Zhou, H.; Chen, S. 3D global mapping of large-scale unstructured orchard integrating eye-in-hand stereo vision and SLAM. *Comput. Electron. Agric.* **2021**, *187*, 106237. [[CrossRef](#)]
35. Comba, L.; Zaman, S.; Biglia, A.; Ricauda Aimonino, D.; Dabbene, F.; Gay, P. Semantic interpretation and complexity reduction of 3D point clouds of vineyards. *Biosyst. Eng.* **2020**, *197*, 216–230. [[CrossRef](#)]
36. Mammarella, M.; Comba, L.; Biglia, A.; Dabbene, F.; Gay, P. Cooperation of unmanned systems for agricultural applications: A case study in a vineyard. *Biosyst. Eng.* **2021**, *in press*. [[CrossRef](#)]
37. Donati, C.; Mammarella, M.; Comba, L.; Biglia, A.; Dabbene, F.; Gay, P. Improving agricultural drone localization using georeferenced low-complexity maps. In Proceedings of the 2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor), Trento-Bolzano, Italy, 3–5 November 2021; doi: 10.1109/MetroAgriFor52389.2021.9628607. [[CrossRef](#)]
38. Capello, E.; Quagliotti, F.; Tempo, R. Randomized approaches and adaptive control for quadrotor UAVs. In Proceedings of the IEEE 2013 International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 28–31 May 2013; pp. 461–470. [[CrossRef](#)]
39. Powers, C.; Mellinger, D.; Kumar, V. Quadrotor kinematics and dynamics. In *Handbook of Unmanned Aerial Vehicles*; Springer: Dordrecht, The Netherlands, 2015.
40. Liao, A.; Todd, M.J. The ellipsoid algorithm using parallel cuts. *Comput. Optim. Appl.* **1993**, *2*, 299–316. doi: 10.1007/BF01299543. [[CrossRef](#)]
41. Kailath, T.; Sayed, A.H.; Hassibi, B. *Linear Estimation*; Prentice Hall: Upper Saddle River, NJ, USA, 2000.
42. Radoglou-Grammatikis, P.; Sarigiannidis, P.; Lagkas, T.; Moscholios, I. A compilation of UAV applications for precision agriculture. *Comput. Netw.* **2020**, *172*, 107148. [[CrossRef](#)]
43. Puri, V.; Nayyar, A.; Raja, L. Agriculture drones: A modern breakthrough in precision agriculture. *J. Stat. Manag. Syst.* **2017**, *20*, 507–518. [[CrossRef](#)]
44. DJI Matrice 100 Technical Sheet. Available online: <https://www.dji.com/it/matrice100/info#specs> (accessed on 31 January 2022).
45. VectorNav VN-200 GNSS/IMU Datasheet. Available online: <https://www.vectornav.com/products/detail/vn-200> (accessed on 31 January 2022).

46. Ultrasonic Ranging Module HC-SR04 Datasheet. Available online: <https://www.elecrow.com/hcsr04-ultrasonic-ranging-sensor-p-316.html> (accessed on 31 January 2022).
47. Christiansen, M.P.; Laursen, M.S.; Jørgensen, R.N.; Skovsen, S.; Gislum, R. Designing and testing a UAV mapping system for agricultural field surveying. *Sensors* **2017**, *17*, 2703. [[CrossRef](#)] [[PubMed](#)]
48. Xiang, H.; Tian, L. Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (UAV). *Biosyst. Eng.* **2011**, *108*, 174–190. [[CrossRef](#)]
49. Rambabu, R.; Bahiki, M.R.; Azrad, S. Multi-sensor fusion based UAV collision avoidance system. *J. Teknol.* **2015**, *76*, 89–93. [[CrossRef](#)]
50. Oh, K.H.; Ahn, H.S. Extended Kalman filter with multi-frequency reference data for quadrotor navigation. In Proceedings of the 2015 15th International Conference on Control, Automation and Systems (ICCAS), Busan, Korea, 13–16 October 2015; pp. 201–206. [[CrossRef](#)]
51. Wang, Y.; Nguyen, B.M.; Kotchapansompote, P.; Fujimoto, H.; Hori, Y. Vision-based vehicle body slip angle estimation with multi-rate Kalman filter considering time delay. In Proceedings of the 2012 IEEE International Symposium on Industrial Electronics, Hangzhou, China, 28–31 May 2012; pp. 1506–1511. [[CrossRef](#)]