engineering proceedings

Proceeding Paper

Indoor Position Estimation Using Ultrasonic Beacon Sensors and Extended Kalman Filter †

Tolga Bodrumlu [1,*] and Fikret Caliskan [2]

1 Mechatronics Engineering Department, Istanbul Technical University, Istanbul 34457, Turkey
2 Control and Automation Engineering Department, Istanbul Technical University, Istanbul 34457, Turkey
* Correspondence: bodrumlu@itu.edu.tr

† Presented at the 9th International Electronic Conference on Sensors and Applications, 1–15 November 2022; Available online: https://ecsa-9.sciforum.net/.

Abstract: With the invention of GPS and related technologies, outdoor positional systems have become very accurate. However, there is still a need for efficient, reliable, and less expensive technology for indoor navigation. There are lots of techniques used for indoor navigation, such as acoustic, Wi-Fi-based, proximity-based, infrared systems and SLAM algorithms. In this study, accurate position estimation was attempted by combining the acceleration and gyroscope data and the raw distance data with the help of the Extended Kalman Filter (EKF). Initially, a position estimation was obtained using the Recursive Least Square (RLS) method with a trilateration algorithm. This solution was used as a starting point for RLS. Here, the first solution point is updated as the initial solution for each distance, and the result calculated by the RLS method is updated as the next solution. This approach enables the distance measurement and position estimation to be executed simultaneously, avoids the unnecessary waiting time, and speeds up the positioning estimation. After that, this position estimation is fused with the acceleration and gyroscope data. In order to test the designed algorithm, synthetic data were used. As a result of these tests, it has been observed that this EKF structure created for indoor navigation gives accurate results.

Keywords: indoor navigation; Extended Kalman Filter; sensor fusion

check for updates

Citation: Bodrumlu, T.; Caliskan, F. Indoor Position Estimation Using Ultrasonic Beacon Sensors and Extended Kalman Filter. *Eng. Proc.* **2022**, *27*, 16. https://doi.org/10.3390/ecsa-9-13353

Academic Editor: Jean-marc Laheurte

Published: 1 November 2022

1. Introduction

In recent years, unmanned aerial vehicles (UAVs) have been widely used in military, industry, agriculture, and other areas, such as aerial photography and air reconnaissance [1–3]. All these specified areas are outdoors, so the UAVs usually receive the GPS signals properly. If there is no GPS signal or a weak GPS signal, however, positioning accuracy is directly affected. Nowadays, there is a great demand for UAV inspection based on indoor technology, and this demand is related to control optimization and path tracking.

There are a lot of techniques for indoor positioning, such as vision-based, lidar-based, Wi-Fi-based, Bluetooth-based, UWB-based and IMU-based techniques [4–6]. A motion-capturing system, which uses multiple high-speed cameras to obtain the relative position of the object, has the disadvantage of complex layout and difficult calibration. VICON and OptiTrack are examples of this system. Since positioning accuracy can now reach millimeter level, the disadvantages of the system have prompted the emergence of some simultaneous localization and mapping schemes, such as Oriented FAST and Rotated BRIEF SLAM (ORB-SLAM) [7], semi-direct visual odometry (SVO) [8] and direct sparse odometry (DSO) [9]; these schemes use a single monocular camera or a binocular camera placed on the UAV body to obtain the relative position of the UAV in the environment. In addition, Gmapping [10], Hector [11] and Cartographer [12] are examples of lidar-based positioning techniques. Due to the weight of the lidar, which is generally a single line applied to the UAV, only a two-dimensional position can be obtained. Using Wi-Fi for indoor positioning is well-established, and its accuracy can reach a few meters [13]. However, its requirements—in terms of the number of Wi-Fi access points associated with the costs and power consumption—make this solution impossible without consistent retrofitting. Bluetooth Low Energy (BLE) and Wi-Fi use the same frequency, but BLE is designed as a short-range, energy-efficient communication protocol, which allows devices to communicate through short messages [14]. BLE-based localization is typically performed by installing a set of proximity beacons at known locations. Receivers transmit the RSSI (distance from the sender) from the nearest beacons and calculate their own position using these values [15]. There are two categories for the BLE-based localization algorithm—distance-based and fingerprinting-based [16]. Distance-based algorithms directly translate RSSI values into the position coordinates. These methods require at least three RSSI measurements to estimate the position [17]. On the other hand, fingerprinting-based algorithms exploit a vector of RSSI measurements at known fingerprint positions to create a so-called reference fingerprint map (RFM). A machine learning regressor is then fed with the RFM data to create a relationship rule between new RSSI measurements and their corresponding position estimates [18]. UWB positioning is light in weight, simple in layout and stable in positioning, and the accuracy can reach to centimeter level [19,20]. Using only UWB cannot meet the requirements of an indoor high-precision operation. IMU is a common sensor for orientation estimation. IMU, however, estimates its position by integration, which accumulates errors due to drift.

In this paper, an accurate position estimation is calculated by combining the IMU and the raw distance data with the help of the Extended Kalman Filter (EKF). Initially, a position estimation is obtained using the Recursive Least Square (RLS) method with a trilateration algorithm. This solution is used as a starting point for RLS. This position estimation is then fused with acceleration and gyroscope data. These algorithm simulations are performed in a MATLAB environment. The average results show that the proposed algorithm gives accurate results within less than ten cm precision.

2. Position Estimation Algorithm

2.1. Geometric Approach

A geometric approach has been put forward in the basis of the study. As shown in the Figure 1 below, three reference points are given—$B_1(x_1, y_1, z_1)$, $B_2(x_2, y_2, z_2)$ and $B_3(x_3, y_3, z_3)$—and d_1, d_2 and d_3 interval measurements up to point A are given. The determination of the coordinates of point A is carried out by solving the system of quadratic equations.

$$\begin{array}{c} (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 = d_1{}^2 \\ (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = d_2{}^2 \\ (x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 = d_3{}^2 \end{array} \quad (1)$$

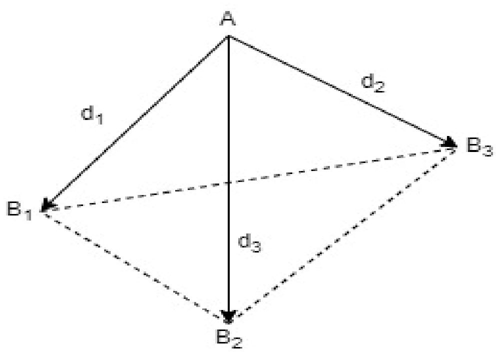


Figure 1. Reference Points and Interval Measurements.

The system of equations given here can expressed as follows:

$$\begin{aligned}(x^2+y^2+z^2)-2x_1x-2y_1y-2z_1z=d_1{}^2-x_1^2-y_1^2-z_1^2\\(x^2+y^2+z^2)-2x_2x-2y_2y-2z_2z=d_2{}^2-x_2^2-y_2^2-z_2^2\\(x^2+y^2+z^2)-2x_3x-2y_3y-2z_3z=d_3{}^2-x_3^2-y_3^2-z_3^2\end{aligned} \tag{2}$$

In addition to that, this expression can be shown in matrix form as below.

$$\begin{bmatrix}1 & -2x_1 & -2y_1 & -2z_1\\1 & -2x_2 & -2y_2 & -2z_2\\1 & -2x_3 & -2y_3 & -2z_3\end{bmatrix}\begin{bmatrix}x^2+y^2+z^2\\x\\y\\z\end{bmatrix}=\begin{bmatrix}s_1^2-x_1^2-y_1^2-z_1^2\\s_2^2-x_2^2-y_2^2-z_2^2\\s_3^2-x_3^2-y_3^2-z_3^2\end{bmatrix} \tag{3}$$

This matrix form generally can be expressed as follows.

$$\begin{gathered}A_0.x=b_0 \quad \mathrm{x}\in E\\E=\left\{\ (x_0,x_1,x_2,x_3)^T\epsilon \quad x_0=x_1^2+x_2^2+x_3^2\right\}\end{gathered} \tag{4}$$

While looking at the solution set of this system, it can be seen that there are two different approaches. The first approach is that points B_1, B_2 and B_3 are not on the same straight line, and the second approach is that the points are on the same straight line.

Case 1. B_1, B_2 and B_3 are not in a straight line

In this case, the following propositions are true. Rank(A_0) = 3, and dim(Kern(A_0)) = 1. Then the general solution of (4) can be shown as:

$$x=x_p\ +t.x_h \tag{5}$$

where t is a real coefficient, it is seen that x_p is the special solution of (4) and it is also the solution of the system $A_0.\ x=0$, which is a homogeneous system at x_h. The vectors x_p and x_h can be calculated using the Gaussian elimination method.

$$x_p=\left(x_{p0},x_{p1},x_{p2},x_{p3}\right)^T,\ x_h=(x_{h0},x_{h1},x_{h2},x_{h3})^T,x=(x_0,x_1,x_2,x_3)^T \tag{6}$$

x_p, x_h and x are expressed as above. If we substitute these expressions in (6), we can obtain the expressions given below:

$$x_0=x_{p0}+tx_{h0,}\quad x_1=x_{p1}+tx_{h1}\,,x_2=x_{p2}+tx_{h2}\ ,\ x_3=x_{p3}+tx_{h3} \tag{7}$$

By using the constraint $x\in E$,

$$x_{p0}+tx_{h0}=\left(x_{p1}+tx_{h1}\right)^2+\left(x_{p2}+tx_{h2}\right)^2+\left(x_{p3}+tx_{h3}\right)^2 \tag{8}$$

$$\begin{gathered}t^2\left(x_{h1}^2+x_{h2}^2+x_{h3}^2\right)+t\left(2x_{p1}x_{h1}+2x_{p2}x_h+2x_{p3}x_{h3}-x_{h0}\right)+\\x_{p1}^2+x_{p2}^2+x_{p3}^2-x_{p0}=0\end{gathered} \tag{9}$$

This is a quadratic equation in the form $at^2+bt+c=0$ with the solutions.

$$t_{1/2}=\frac{-b\pm\sqrt{b^2-4ac}}{2a} \tag{10}$$

The solutions of the equation system can be shown below.

$$x_1=x_p+t_1x_h\ \ ,\ \ x_2=x_p+t_2x_h \tag{11}$$

Case 2. B_1, B_2 and B_3 are in a straight line

Then the following propositions are true. Rank(A_0) = 2, and dim(Kern(A_0)) = 2. Then the general solution of (4) can be shown as:

$$x = x_p \ + t.x_{h1} \ + k.x_{h2} \tag{12}$$

With real parameters t and k; x_p is a particular solution of (4), and x_{h1} and x_{h2} are two solutions of the homogeneous system A_0. x = 0. They are linearly independent solutions and, therefore, form a basis of Kern(A_0). If there are more than three reference points, the general solution can be found using the least square method as follows.

$$\hat{x} = \left(A^T A\right)^{-1} A^T b \tag{13}$$

The projection of p on the column space of A is

$$p = A\left(A^T A\right)^{-1} A^T b \tag{14}$$

In this case, the coordinates of p in the Col (A) column space represent the $\hat{x}$ solution. Although, if the measurements are uncorrelated but have different uncertainties, Weighted Least Squares (WLS) is used. In this case, the solution of $\hat{x}$ is found with the help of the following expression:

$$\hat{x} = \left(A^T V^{-1} A\right)^{-1} A^T V^{-1} b \tag{15}$$

This solution is used as a starting point for the Recursive Least Square (RLS). Let x_0 be the initial solution, and by every incoming distance, x_0 is updated in x_1 by using the RLS. The approach enables distance measurement and positioning calculation to be executed simultaneously. Hence, a position assignment can be initiated although not all distances are known. This avoids the unnecessary waiting time and speeds up the positioning calculation. More detail for this approach is discussed in [21]. Distance data is used together with the RLS algorithm to help in calculating the position. In the next section, the details of a more accurate position estimation with the help of EKF will be explained. A sensor fusion algorithm is used with accelerometer, gyroscope and distance data, and the position is calculated.

2.2. Sensor Fusion Algorithm

There are lots of sensor fusion algorithms, such as Feature Aggregation, Temporal Fusion, Support Vector Machine, Kalman Filter, etc. Although, in this system, Kalman Filter was selected for use to perform a more accurate position estimation. Kalman Filter gives good results in linear systems, but since there are very few linear systems in the real world, the Extended Kalman Filter (EKF) is used, which gives better results in non-linear systems. The EKF solves this problem by calculating the Jacobian of F and H around the estimated states, which in turn, yields a trajectory of the model function around the stated. The details of the EKF that are utilized in this work are presented. The nonlinear process model and noise used in EKF are as given:

$$x(k+1) = f(x(k), u(k)) + w(k) \tag{16}$$

In this equation, $x(k)$ and $x(k+1)$ represents the states of the system at k and $k+1$, respectively. In addition, $u(k)$ and $w(k)$ represent the control signal and the process noise (in Gaussian distribution), respectively. The process is expressed by $f(.)$. The measurement model, which relates the state variables to the measurements, is expressed with the following equation:

$$z(k) = h(x(k)) + v(k) \tag{17}$$

In this equation, $h(.)$, $v(k)$ and $z(k)$ represent nonlinear measurement function, measurement noise (in Gaussian distribution) and measurements, respectively. In EKF, the filter gain is calculated in the same way as in the linear Kalman Filter. For this reason, the nonlinear process and measurement models are linearized around the current system states. This linearization is performed using the first terms of the Taylor series expansion of the function of interest.

$$x(k+1) \approx \tilde{x}(k+1) + F(x(k) - \tilde{x}(k)) + \Gamma w(k) \tag{18}$$

The mean value of the noise is zero. (w = 0)

$$\tilde{x}(k+1) \approx f(x(k), 0) \tag{19}$$

The F matrix is the Jacobian matrix of the process function (f), according to the states (x). The Γ matrix is the Jacobian matrix of the process function with respect to the noise (w).

$$F_{i,j} = \left.\frac{\partial f_i}{\partial x_j}\right|_{(\hat{x}(k+1),0)}, \ \Gamma_{i,j} = \left.\frac{\partial f_i}{\partial w_i}\right|_{(\hat{x}(k+1),0)} \tag{20}$$

Similarly, the nonlinear measurement function is linearized around the predicted states.

$$z(k+1) \approx \tilde{z}(k+1) + H(x(k+1) - \hat{x}(k+1)) + \Phi v(k+1) \tag{21}$$

The expected noise value is zero (v = 0):

$$\tilde{z}(k+1) = h(\tilde{x}(k+1), 0) \tag{22}$$

The H matrix is the Jacobian matrix (x), according to the system states of the measurement function (h). Likewise, the Φ matrix is the Jacobian matrix with respect to the measurement noise (v) of the measurement function. The general schematic of the EKF structure used in the system is given in Figure 2 as follows:

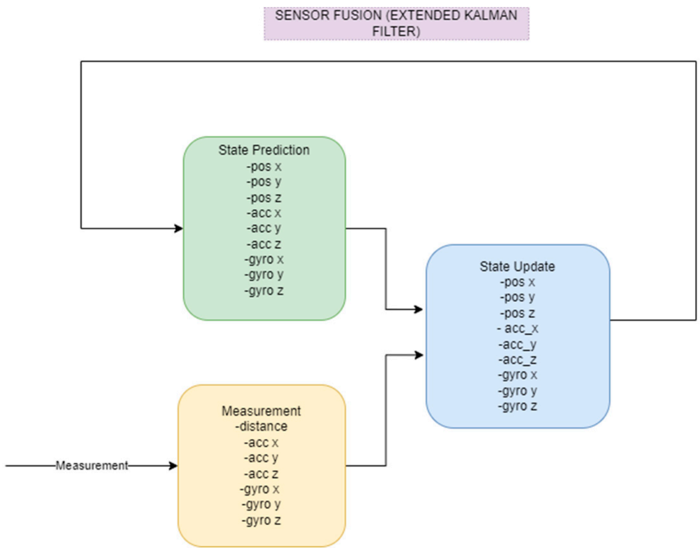


Figure 2. Sensor Fusion (EKF) Structure Schematic.

The state vector of the system is given as:

$$x = \begin{bmatrix} pos_x \\ pos_y \\ pos_z \\ acc_x \\ acc_y \\ acc_z \\ gyro_x \\ gyro_y \\ gyro_z \end{bmatrix} \tag{23}$$

The state transition of the system is given as:

$$A = \begin{bmatrix} 1 & 0 & 0 & dt & 0 & 0 & dt^2/2 & 0 & 0 \\ 0 & 1 & 0 & 0 & dt & 1 & 0 & dt^2/2 & 0 \\ 0 & 0 & 1 & 0 & 0 & dt & 0 & 0 & dt^2/2 \\ 0 & 0 & 0 & 1 & 0 & 0 & dt & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & dt & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & dt \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{24}$$

The measurement vector is given as:

$$z = \begin{bmatrix} distance \\ acc_x \\ acc_y \\ acc_z \\ gyro_x \\ gyro_y \\ gyro_z \end{bmatrix} \tag{25}$$

The measurement noise covariance matrix, R, was determined based on the average noise levels of measurements. Assuming that the measurements are not correlated with each other, the diagonal matrix below is chosen. The standard deviation values of the measurements are calculated, and the measurement noise covariance matrix is decided by using these values.

$$\mathrm{R} = \mathrm{diag}([\mathrm{r}_1\ \mathrm{r}_2\ \mathrm{r}_3\ 0\ 0\ 0\ \mathrm{r}_7\ \mathrm{r}_8\ \mathrm{r}_9]) \tag{26}$$

Here, $\mathrm{r}_1 \ldots \mathrm{r}_9$ values are the distances of the beacon sensors from each other. The resulting process noise covariance matrix is as follows:

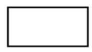
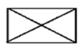
$$Q = \begin{bmatrix} 0.01 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0001 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.16 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.01 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.16 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.01 \end{bmatrix} \tag{27}$$

In the next section, the results obtained by applying the designed algorithm on the simulation will be explained.

3. Simulation System and Results

The MATLAB environment was used while creating the simulation system. The data were produced synthetically in MATLAB, and the algorithm was tested under the generated data. In addition, three different trajectories were used while generating synthetic data. These trajectories are shown in Table 1. In Figures 3–5, both the position calculations were calculated with the trilateration algorithm alone, and the position calculations obtained as a result of using the IMU data together with the trilateration and EKF algorithms are shown. The minimum, maximum and average error amounts of the calculated positions are shown in detail in Table 2. The RMS value was used while generating error amounts. According to the simplicity and complexity of the determined trajectories, the error amounts obtained by using only the trilateration algorithm differ. For example, the error amounts of position estimation and position estimation obtained by using only the trilateration algorithm between Trajectory 1 and Trajectory 3 differ considerably. However, with the inclusion of EKF in the calculation of position estimation, it is easily observed that the amount of error obtained decreases both in the relevant Figures 3–5 and in the values given in Table 2.

Table 1. Trajectories.

Trajectory Name	Beacon Number	Trajectories
Trajectory 1	5	
Trajectory 2	5	
Trajectory 3	5	

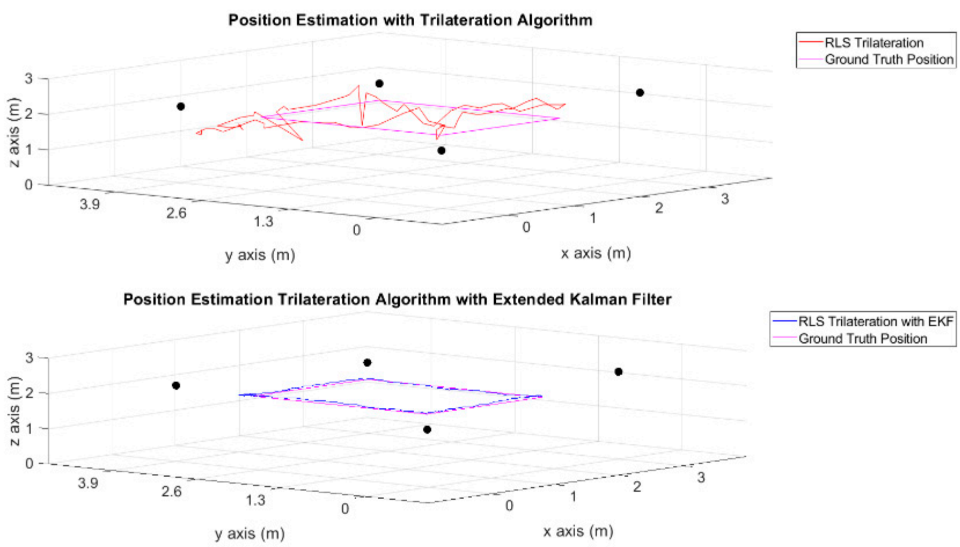


Figure 3. Position Estimation in Path 1 Trilateration Algorithm with and without EKF.

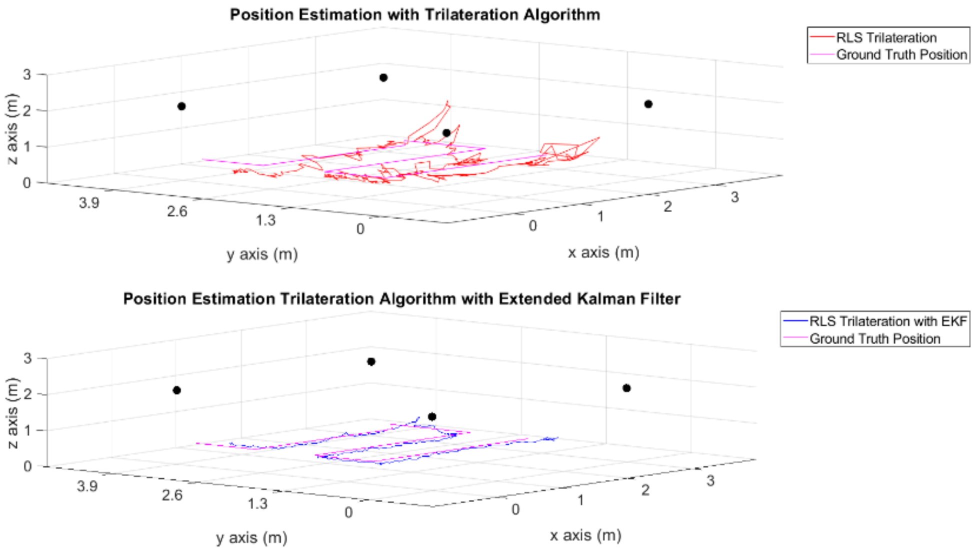


Figure 4. Position Estimation in Path 2 Trilateration Algorithm with and without EKF.

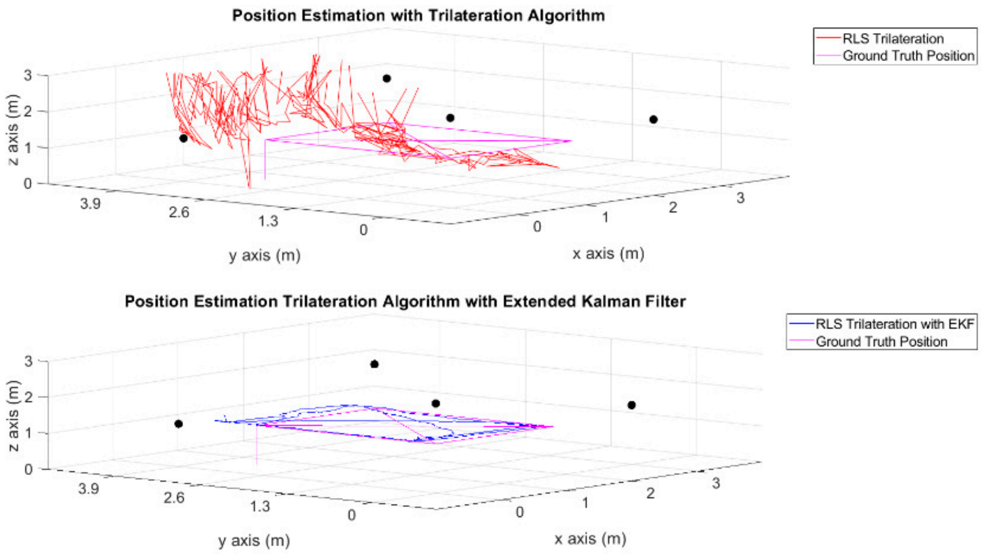


Figure 5. Position Estimation in Path 3 Trilateration Algorithm with and without EKF.

Table 2. Error Comparison Table.

Trajectory	Algorithm	Min Error (m)	Mean Error (m)	Max Error (m)
Trajectory 1	Trilateration Algorithm	0.073409	0.335606	0.994064
Trajectory 1	Trilateration Algorithm + EKF	0.019016	0.061539	0.143751
Trajectory 2	Trilateration Algorithm	0.077657	0.453345	1.768042
Trajectory 2	Trilateration Algorithm + EKF	0.016976	0.082958	0.168863
Trajectory 3	Trilateration Algorithm	0.353665	1.258376	5.003108
Trajectory 3	Trilateration Algorithm + EKF	0.001041	0.083402	0.164720

4. Conclusions

In this study, position estimation was made by combining IMU and raw distance data with the help of the Extended Kalman Filter (EKF). Simulation of the system is carried on in the MATLAB environment. The simulation result shows that the proposed method gives the correct position in centimeter precision levels. First, a geometric solution method was used in the algorithm; then this method was combined with the EKF algorithm. When the results are examined, it is observed that the amount of error is quite high when only the geometric approach is used. It has been seen that the position estimation has reached the desired level with the use of EKF as well as the geometric approach. In the future, this designed algorithm will be tested with the real sensor data. If the obtained results are at the desired level, the integration of the algorithm into the UAV will be started.

Author Contributions: Conceptualization, T.B. and F.C.; methodology, T.B.; software and data acquisition T.B.; validation, T.B.; writing—original draft preparation, T.B.; writing—review and editing, T.B. and F.C.; visualization, T.B.; supervision, F.C.; project administration, F.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was financially supported by Istanbul Technical University, grant number 42754.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Valavanis, K.P.; Vachtsevanos, G.J. *UAV Applications: Introduction*; Springer: Amsterdam, The Netherlands, 2015.
2. Samad, T.; Bay, J.S.; Godbole, D. Network-centric systems for military operations in urban Terrain: The role of UAVs. *Proc. IEEE* **2007**, *95*, 92–107. [CrossRef]
3. Li, Z.; Liu, Y.; Walker, R.; Hayward, R.; Zhang, J. Towards automatic power line detection for a UAV surveillance system using pulse coupled neural filter and an improved Hough transform. *Mach. Vis. Appl.* **2010**, *21*, 677–686. [CrossRef]
4. Nirjon, S.; Liu, J.; DeJean, G.; Priyantha, B.; Jin, Y.; Hart, T. COIN-GPS: Indoor localization from direct GPS receiving. In Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services—MobiSys 2014, Bretton Woods, NH, USA, 16–19 June 2014; pp. 301–314.
5. Vasisht, D.; Kumar, S.; Katabi, D. Decimeter-Level Localization with a Single WiFi Access Point. In Proceedings of the USENINX Symposium on Networked Systems Design and Implementation, Santa Clara, CA, USA, 16–18 March 2016; pp. 165–178.
6. Zafari, F.; Papapanagiotou, I.; Christidis, K. Microlocation for internet-of-things-equipped smart buildings. *IEEE Internet Things J.* **2016**, *3*, 96–112. [CrossRef]
7. Mur-Artal, R.; Tardos, J.D. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [CrossRef]
8. Forster, C.; Pizzoli, M.; Scaramuzza, D. SVO: Fast semi-direct monocular visual odometry. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 15–22.
9. Engel, J.; Koltun, V.; Cremers, D. Direct sparse odometry. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 611–625. [CrossRef] [PubMed]
10. Grisetti, G.; Stachniss, C.; Burgard, W. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Trans. Robot.* **2007**, *23*, 34–46. [CrossRef]
11. Kohlbrecher, S.; von Stryk, O.; Meyer, J.; Klingauf, U. A flexible and scalable SLAM system with full 3D motion estimation. In Proceedings of the 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics, Kyoto, Japan, 1–5 November 2011; pp. 155–160.
12. Ren, R.; Fu, H.; Wu, M. Large-scale outdoor SLAM based on 2D lidar. *Electronics* **2019**, *8*, 613. [CrossRef]
13. Chintalapudi, K.; Padmanabha Iyer, A.; Padmanabhan, V.N. Indoor localization without the pain. In Proceedings of the Annual International Conference on Mobile Computing and Networking, Mobicom, Chicago, IL, USA, 20–24 September 2010; pp. 173–184.
14. Gomez, C.; Oller, J.; Paradells, J. Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology. *Sensors* **2012**, *12*, 11734–11753. [CrossRef]
15. Wang, Y.; Ye, Q.; Cheng, J.; Wang, L. RSSI-Based Bluetooth Indoor Localization. In Proceedings of the 11th International Conference on Mobile Ad-Hoc and Sensor Networks, MSN 2015, Shenzhen, China, 16–18 December 2015; pp. 165–171.
16. Cabarkapa, D.; Grujic, I.; Pavlović, P. Comparative analysis of the Bluetooth Low-Energy indoor positioning systems. In Proceedings of the 2015 12th International Conference on Telecommunication in Modern Satellite, Cable and Broadcasting Services (TELSIKS), Nis, Serbia, 14–17 October 2015; pp. 76–79.
17. Rida, M.E.; Liu, F.; Jadi, Y.; Algawhari, A.A.; Askourih, A. Indoor location position based on bluetooth signal strength. In Proceedings of the 2015 2nd International Conference on Information Science and Control Engineering, ICISCE 2015, Shanghai, China, 24–26 April 2015; pp. 769–773.
18. Kriz, P.; Maly, F.; Kozel, T. Improving Indoor Localization Using Bluetooth Low Energy Beacons. *Mob. Inf. Syst.* **2016**, *2016*, 2083094. [CrossRef]

19. Aiello, G.R.; Rogerson, G.D. Ultra-wideband wireless systems. *IEEE Microw. Mag.* **2003**, *4*, 36–47. [CrossRef]
20. Chehri, A.; Fortier, P.; Tardif, P.M. UWB-based sensor networks for localization in mining environments. *Ad Hoc Netw.* **2009**, *7*, 987–1000. [CrossRef]
21. Norrdine, A. An Algebraic Solution to the Multilateration Problem. In Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation, Sydney, Australia, 13–15 November 2012; pp. 1–4.