# Non-Line-Of-Sight Error Mitigation For Ultrasonic Positioning System Using LiDAR

A. Dudlik, C. Mikkelsen, N. Sørensen, H. Schiøler

***Abstract*—Localisation methods often fuse measurements from both proprioceptive and exteroceptive sensors. Such exteroceptive sensors can be used in multilateration by measuring the distance to pre-installed beacons; however, these position estimates can be biased due to Non-Line-of-Sight (NLoS) errors, caused by an occlusion between receiver and beacon, resulting in the measured Time of Arrival increasing, elongating distance measurement.**
**This paper proposes a method to NLoS identification and mitigation, where a robot's position estimate is obtained using an Extended Kalman Filter (EKF), which fuses IMU data with distance measurements from an Ultrasonic Beacon System. The NLoS detection is done using LiDAR measurements, to compare the position of the detected surroundings with the position of the currently measured beacon; if the object is detected to be on a direct path between the LiDAR and current beacon, the needed occlusion height is calculated and compared to a Workspace Height Model (WHM). Subsequently, if NLoS is detected, the R-value in the EKF decreases, so the distance measurement from the occluded beacon is weighted less. The system is tested in a manufacturing laboratory, where the results for the NLoS scenario with a LiDAR-augmented EKF is compared to a baseline EKF position estimator. The results show no significant difference between the two mentioned methods.***

***Keywords: Indoors Localization, LiDAR, Ultra-Sonic Beacons, Kalman Filter, Non-Line-Of-Sight mitigation*.**

## I. Introduction

With the on-going fourth industrial revolution, innovation is regularly done in industrial environments and production facilities; this includes flexible production lines and mobile robots for transportation, both systems requiring knowledge on their location [1]. There are localisation methods relying on proprioceptive sensors, such as odometry and inertial navigation, that localise based on previous local measurements (i.e. dead-reckoning). On the other hand, there are methods relying on exteroceptive sensors, focusing on absolute position measurements (i.e. global reference-based systems), that localise relative to global features, e.g. active beacons, global positioning systems (GPS), magnetometers, landmark localisation using computer vision or map matching. [2]

In GPS-denied environments, a proprioceptive sensors based localization stack on the robot is often combined with active beacon architecture. The proprioceptive sensors can provide high precision estimates over short ranges, but they can suffer from drift, which can be alleviated by fusing it with infrastructural localization, while the local reference system results provide increased precision to the infrastructural system [3] [4]. Nevertheless, if something occludes the path between the robot's receiver and the beacon transmitter, the Time-of-Arrival (TOA) measurement that the active beacon localisation system uses would be skewed, which leads to an error in distance measurement. This is called a Non-Line-of-Sight (NLoS) error [5].

There are different solutions for detecting and mitigating the NLoS error. It was shown that NLoS can be detected with the difference in Euclidean distance [6], or by using the Mahalanobis distance [7]. When it comes to NLoS mitigation, geometry-based methods have shown promise for radio-based localization [8]; however, these methods cannot be applied to sound-based localisation as they use assumptions inherent to the reflection properties of electromagnetic waves. It was also shown that it can be set up as a minimization problem [9], although that solution is computation heavy. Lastly, it was shown that applying a dual-filter with a Kalman Filter and a Friedland filter [10] can also mitigate the NLoS error.

Sensor fusion is beneficial for robot localisation, as it is used to augment the weak-points of one sensor, with the benefits of another, thereby resulting in a stronger and more reliable overall system. As such, the robot position is estimated, based on multiple sensor readings. There are different methods for sensor fusion based on probabilistic models. A widely used sensor fusion method is the Kalman Filter (KF), derived from Bayes Filter, which assumes linearity of the system and Gaussian distribution of a measurement noise. However, real-world robot systems are non-linear and that is where Extended Kalman Filter can potentially be applied (EKF), with a local linearization of the observed state. [11] [12] [13]

LiDARs are regularly used for obstacle avoidance and mapping. Therefore, it is plausible to assume many mobile robots are equipped with such a sensor. It is worth investigating if a distance measurement obtained by a LiDAR can be fused with an EKF localization system, to alleviate NLoS errors.

This paper seeks to mitigate the NLoS-errors of an Ultrasonic Beacon System (GOTposition [14]) fused with localization stack using an accelerometer and gyroscope, with a low-end RPLiDAR A1 [15]. The LiDAR is used to detect obstacles between the detected beacon and the differential drive robot, thereby introducing a way to quantify the level of NLoS of a beacon distance measurement in the EKF, which is explained in the following section II along with test methodology. The test results are shown in section III and discussed in section IV.

## II. Methods and Materials

This paper seeks to conclude if detecting possible NLoS with LiDAR can improve the localisation estimate of a robot operating in an industrial environment.

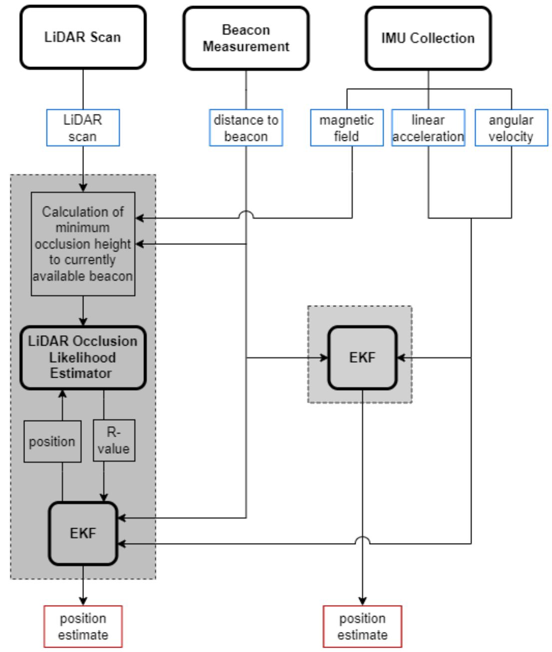


**Fig. 1:** A flow-diagram illustrating how the system works, where system inputs are marked in blue boxes, whilst the system outputs are marked in red boxed, which are position estimates. The position estimate is obtained simultaneously with two EKFs, where one is augmented with LiDAR and based on the LiDAR scan that is fed to a Height Model subsystem; if NLoS is detected, it changes the measurement covariance estimate $R$ in the EKF, so the beacon distance measurement is weighted less.

The implementation of the system is done by obtaining distance measurements provided by active beacons mounted on the ceiling in the tested area, fused with positioning data obtained with an on-board IMU. When the LiDAR detects an obstacle in the path of the current beacon measurement, a probability of that obstacle creating NLoS is calculated with the Workspace Height Model (WHM) and fed to the EKF.

The beacons used for testing are provided by Aalborg University based on the GOT hardware [14]. It was observed that the system received beacon data at a rate of approximately $5\ Hz$.

### *A. Extended Kalman Filter*

The implementation of the IMU-driven EKF [16] uses the on-board accelerometer and gyroscope data as inputs and fused with the distance measurement to an active beacon. The idea behind Kalman filters is to estimate the posteriori process state $\hat{x_t}$ in terms of a priori process state $\hat{x}_{t-1}$, with a weighted difference (so-called *residual*) between the actual measurement $z_t$ and predicted (priori) measurement $H(\hat{x_t})$ [12]. The state $x_t$ at time $t$ is defined to be:

$$x_t = [p^T, v, h^T]^T \tag{1}$$

where $p$ is a 2d position vector, $v$ is a 1d forward velocity and $h$ is a 2d heading unit vector. The dynamics for each component are the following:

$$\frac{d}{dt}p = v \cdot h \tag{2}$$

$$\frac{d}{dt}v = a_x \tag{3}$$

$$\frac{d}{dt}||h|| = Rot_{90} \cdot h \cdot \omega_z \tag{4}$$

where $a_x$ is linear acceleration along the robot's x-axis, $\omega_z$ is angular velocity around the robot's z-axis and $Rot_{90}$ is a 90 degree rotation matrix. The measurement residual is calculated as a squared difference between the actual measurement $z_t$ and the estimated measurement $\hat{z_t}$, which are the following:

$$z_t = d^2 \tag{5}$$

$$\hat{z_t} = (p - b_i)^2 \tag{6}$$

where $d$ is a ToA-calculated distance between the robot and the currently active beacon, $p$ is a calculated 2d position vector described by equation 2 and $b_i$ is a set of known 2d beacon positions, where $i$ denotes the currently active beacon. The mentioned $d$ distance measurement is projected down to the robot's operational plane, making it a 2d component. Now, the state $x_t$ can be updated, based on the obtained measurement residual. The weight of a measurement is based on the measurement noise covariance $R$, which residual noise covariance $S$ depends on; $R$ is a constant of 0.01. The process noise covariance $Q$, which predicts the covariance estimate $P$, is chosen to be an identity matrix multiplied by 100.

The state transition Jacobian $F$, used for the predicted covariance estimate $P$, is a $5x5$ matrix, resulting in:

$$F = \begin{bmatrix} 1 & 0 & h_x \cdot \Delta t & v \cdot \Delta t & 0 \\ 0 & 1 & h_y \cdot \Delta t & 0 & v \cdot \Delta t \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -\omega_z \cdot \Delta t \\ 0 & 0 & 0 & \omega_z \cdot \Delta t & 1 \end{bmatrix}$$

based on Euler discretization of state models described in equations 2, 3 and 4. The observation Jacobian $H$, used for the residual covariance, is a $1x5$ matrix, resulting in:

$$H = \begin{bmatrix} -2(b_{ix} - p_x) & -2(b_{iy} - p_y) & 0 & 0 & 0 \end{bmatrix}$$

which maps the true state space into the observed state space, using a residual described as a difference between the actual measurement $z$ (see equation 5) and the estimated measurement $\hat{z}$ (see equation 6).

### *B. LiDAR Occlusion Likelihood Estimator*

The LiDAR Occlusion Likelihood Estimation (LOLE) system was developed to detect NLoS during the position estimation process. As the EKF is operating in 2d space, the position of the currently active beacon is initially projected down to the same 2d plane that the LiDAR is operating in. With LiDAR measurements, the system continuously checks if there is any obstacle between the LiDAR and the projected beacon position; if there is, the system calculates how tall the obstacle needs to be to create NLoS. This calculated minimum obstacle height $O_h$ is then fed to the WHM, explained in method II-C. The LOLE system also accounts for cases of non-LiDAR detected obstacles - if the LiDAR does not detect an obstacle, but the difference between the estimated $\hat{D}$ and measured $D$ distance between the robot and the beacon differs more than 5%, then the system assumes the detection of

NLoS and the R-value in the EKF is updated to be the maximum of 100, meaning the beacon reading will now be weighted less. If the occlusion is not detected through either methods, then the R-value stays the same as in the original EKF, meaning 0.01. This is summarized in algorithm 1. As such, the R-value of the system will either be R = 0.01, if the measurement is trusted; R = 100, if the system deems the measurement to be noisy or R = the probability of the object occluding, gotten from the height model II-C, if the LiDAR detects an obstacle in the signal path.

**Algorithm 1** LiDAR Occlusion Likelihood Estimator

**if** $|D - \hat{D}| > 5\%$ of $\hat{D}$ **then**
    set $bell = 100$;
**end if**
**for** the LiDAR detects an obstacle (at LiDAR angle [i]) between itself and the beacon ($\pm 2$ degrees) **do**
    **if** the distance to the LiDAR-detected-object is $<$ the projected distance to the beacon **then**
        save measurement in array
    **end if**
    **if** last run of the for Loop **then**
        find the smallest measurement in array
        find the angle between the floor and the beacon from the robot's point of view
        find the *minimum obstacle height*($O_h$) needed to introduce occlusion at the measured distance
        find the percentage of objects in the workspace with a height above $O_h$ using the Workspace-height-model
        bell = percentage of taller objects, multiplied with 10
    **end if**
    **else**
    bell = 0.01
    **end else**
**end for**
R = bell

### C. Workspace Height Model

As illustrated in figure 3, the obstacle is in the path between the LiDAR and currently active beacon, but it is not tall enough to create NLoS. The WHM was created for this reason, in order to limit the number of beacons filtered out by LOLE (explained in section II-B).

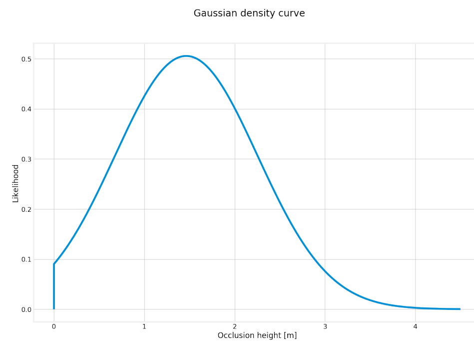


**Fig. 2:** The Gaussian density curve, truncated to positive values (as object height cannot be negative); the x-axis represents occlusion height, whilst the y-axis represents likelihood. This function is used to find the probability of an object being over a certain height; the area under the curve from $O_h$ to infinity represents the percentage of objects exceeding the height of the minimum occlusion. The found percentage is then used in method II-B. This plot is a rough approximation of the workspace obstacle heights measured in the workspace, based on the observed distribution of object heights measured.

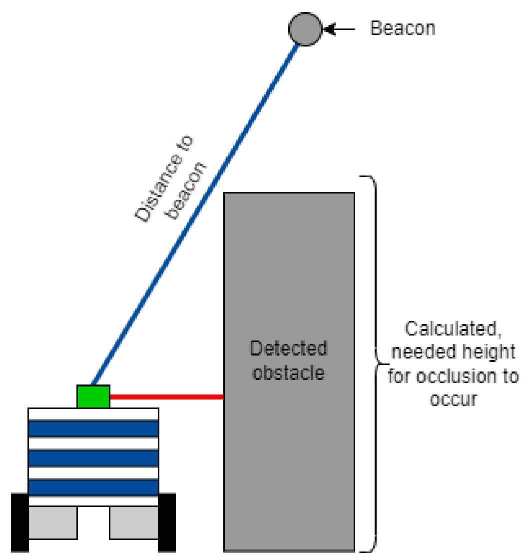


**Fig. 3:** This figure illustrates the reasoning for modelling the WHM, explained in method 2; the LiDAR detects an obstacle between itself and the beacon that is currently transmitting the data, where the needed height for occlusion is extrapolated.

Here, each calculated object's height is compared to the WHM and the likelihood of the object occluding the beacon is used to weigh the beacon measurement (without this solution, each time an obstacle was detected between the beacon and the robot, the distance measurement would result in a fixed R-value).
The height model was created by measuring the heights of a majority of objects in the testing facility. The model only represents the object's height and does not account for the rest of their volume. The mean and variance for this data-set was calculated and used to create a Gaussian density curve, truncated to positive values, as seen in figure 2.

### D. Testing Description

The tests were performed at Aalborg University in an industrial manufacturing laboratory of $12x45\ m$. The testing area within this laboratory was limited to the area where initial static tests showed a good beacon-based position estimate, with 5 beacons in range and minimal foot-traffic during the test periods (see the testing area in figure 4). The NLoS was introduced by constructing a 2 m high tower, using $0.5\ m$ cubes. This tower obscured the Line-of-Sight (LoS) to a single beacon within a limited area of the full scope of the $2.78\ m$ testing path. The approximate ground truth was measured in relation to a specific beacon, where the approximated starting and ending position are at (22653,1039) mm and (22673,3822) mm, respectively, relative to the world frame origo. The test setup is sketched in figure 5.

To ensure repeatability, the tests were performed by moving the robot along a pre-determined path, which was encompassed with metallic rails. The path was measured relative to a beacon, to enable a "ground-truth" path for result comparisons. The data provided by the EKF and LOLE-augmented EKF was recorded simultaneously for each of 40 test (20 for LoS scenario and 20 for NLoS scenario).

The test results are analysed using the t-test methodology [17], to see if the LiDAR addition to the EKF makes any significant difference. This method was used for both the LoS and NLoS scenarios. The critical t-value used in the test is an $\alpha$ of 95 percent.


**Fig. 4:** This map shows the testing facility, where functioning beacons are marked in red and faulty beacons are marked in blue. The blue not-infilled dot represents a beacon that is disregarded from the system due to discrepancies between the "known" and actual beacon position. The testing area marked in this figure represents the area that was initially found to have good coverage, with 5 beacons in range (sketched in figure 5).

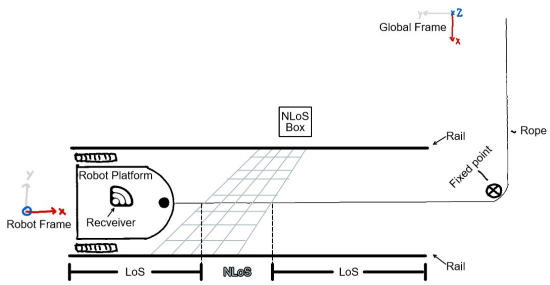


**Fig. 5:** A sketch of the test setup, where the robot is pulled with a rope along rails. The setup was placed in the "Testing area" shown in figure 4.

## III. RESULTS

The tests were conducted according to the method explained in section II-D. The tests were performed in the testing area, shown in figure 4, repeated 20 times for the LoS scenario and another 20 times for the NLoS scenario. This section shows the results of the mentioned tests, where the visualisation of the obtained position estimate is presented in figure 6. For the calculated position estimate, an absolute mean is calculated, along with standard deviation, which is summarized in table I. These results are only representing the position estimation error perpendicular to the path travelled (marked in red in figure 6) and not the position estimation error parallel to this path.

Table I shows the test results; here, the column *"Absolute Mean[mm]"* is the mean position estimate error and the *"Standard Deviation[mm]"* is the standard deviation from this mean. The results in the table are not fully representative, as the error along the $x$- and $y$-coordinates is assumed to be correlated, however the error along the path of travel($x$) is not analysed.

| **NLoS** | Absolute Mean error[mm] | STD of error[mm] |
|---|---|---|
| Kalman | 205.35 | 169.84 |
| LiDAR | 192.41 | 141.44 |
| **LoS** | Absolute Mean error[mm] | STD of error[mm] |
| Kalman | 263.25 | 321.92 |
| LiDAR | 265.16 | 328.88 |

**TABLE I:** This table show the average results from the tests with and without NLoS. The data is projected down to a line perpendicular to the "true path". Therefore, the STD is the relative deviation from the mean error.

A t-test was performed to analyse if the LOLE augmentation of the EKF significantly differed from the baseline EKF. In LoS conditions, the two setups resulted in a t-score of -0.0002 with a 95% confidence and under NLoS conditions, the t-score was -0.0023.

## IV. DISCUSSION

In this paper, two systems for indoor positioning were tested, namely the baseline EKF was compared to its LOLE-augmented version, where the LOLE evaluates NLoS occurrence on the beacon measurements by weighing obstacle NLoS, based on the likelihood of the obstacle heights occluding. Both systems were tested in a controlled LoS and NLoS environment. The purpose was to investigate whether the inclusion of such a LiDAR NLoS estimation system would be an improvement over the baseline EKF position estimator. The results were analysed with a t-test, where the results showed a non-significant difference between the two solutions (seen in section III). It is concluded that the used system implementation does not, in fact, improve the position estimation to a significant degree, perpendicular to the path travelled.

It is theorized that potential improvements could be made by better managing some observed error sources, such as the solution assuming a levelled plane of operation for the robot and receiver, as it was observed that the used testing facility had some minor variance along the floor plane. The tests were performed by manually pulling the robot along a track, wherein the velocity was not constant, which limited the data analysis options. If the tests were repeated with a constant velocity and/or better position tracking along the path over time, then

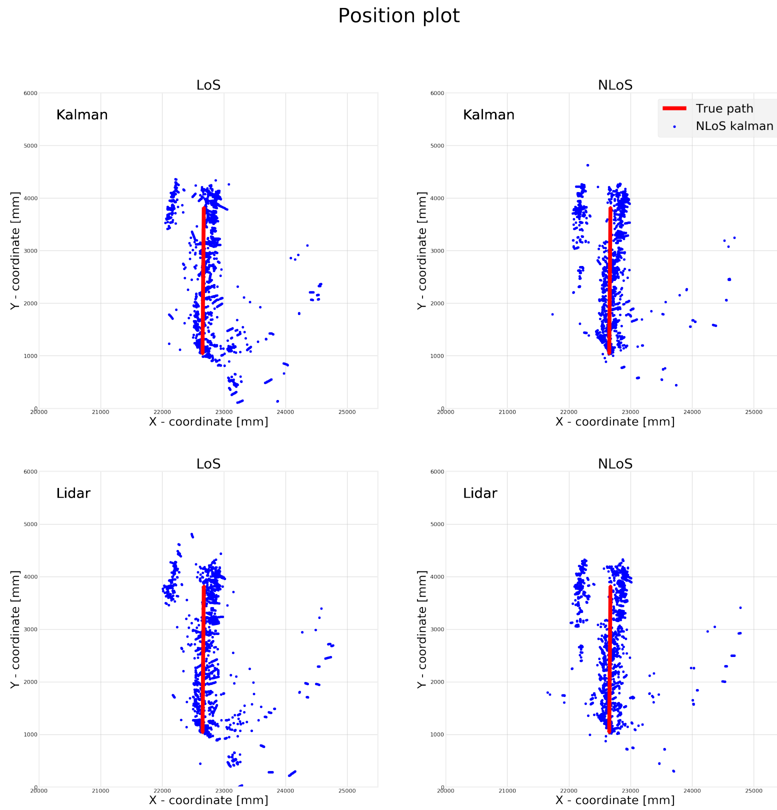


**Fig. 6:** This figure shows the obtained test results for the position estimate; the red line is the true path travelled by the robot, the blue dots are the recorded position estimates throughout 20 repetitions of tests. This is presented for both NLoS and LoS conditions.

the results could be comparable in both x- and y-dimensions, as the NLoS introduces error along both of them.

It was assumed that both systems should have no significant differences in the LoS scenario, since the LOLE-augmented EKF should not change anything; this was successfully observed in the t-test, giving a -0.0002 t-score, with a confidence of 95%. The tests performed under NLoS conditions was assumed to have an actual impact on the method, thereby having a significant difference between the two methods; however, the t-score was calculated to be -0.0023 and shows a non-significant difference, with a confidence of 95%. While the difference is deemed insignificant, it can be observed that there is a factor 10 between the results, as such there is an impact and further system tuning would be recommended.

As it was observed that the position estimate error increased over time, it is believed that the EKF system models requires further work as well.

Overall, this implementation of augmenting the position estimation EKF with the LOLE system has shown no significant improvements in alleviating the Non-Line-of-Sight error. It is still believed that the concept has promise, however, the implementation used in this project requires further refinement and tuning.

## Acknowledgement

Big thanks to *Jens Frederik Dalsgaard Nielsen* for his assistance throughout the project.

## Bibliography

[1] A. Cretu-Sîrcu, H. Schiøler, J. Cederholm, I. Sîrcu, A. Schjørring, I. Rodriguez Larrad, G. Berardinelli, and O. Madsen. Evaluation and comparison of ultrasonic and uwb technology for indoor localization in an industrial environment, 2022.

[2] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe. Mobile robot positioning: Sensors and techniques. *Journal of robotic systems*, 14(4):231–249, 1997.

[3] Hiroaki Seki, Yasuo Tanaka, Masaharu Takano, and Ken Sasaki. Positioning system for indoor mobile robot using active ultrasonic beacons. *IFAC Proceedings Volumes*, 31(3):195–200, 1998. 3rd IFAC Symposium on Intelligent Autonomous Vehicles 1998 (IAV'98), Madrid, Spain, 25-27 March.

[4] Jashandeep Bhuller, Paolo Dela Peña, Vladimir Christian Ocampo II, Julio Simeon, and Lawrence Materum. Ultra Wide Band-based Control of Emulated Autonomous Vehicles for Collision Avoidance in a Four-Way Intersection. *Advances in Science, Technology and Engineering Systems Journal*, 5(6):724–730, 2020.

[5] J. J. Guerrero R. Casas, A. Marco and J. Falcó. Robust estimator for non-line-of-sight error mitigation in indoor localization, 2005.

[6] Yang Yan, Zhengjian Li, and Long Cheng. A mobile localization method in smart indoor environment using polynomial fitting for wireless sensor network, 2020. https://doi.org/10.1155/2020/6787252.

[7] Xingzhen Bai, Liting Dong, Leijiao Ge, Hongxiang Xu, Jinchang Zhang, and Jun Yan. Robust localization of mobile robot in industrial environments with non-line-of-sight situation, 2020. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8959160.

[8] Jingyu Hua, Yejia Yin, Anding Wang, Yu Zhang, and Weidang Lu. Geometry-based non-line-of-sight error mitigation and localization in wireless communications, 2019. https://doi.org/10.1007/s11432-019-9909-5.

[9] S. Venkatraman, J. Caffery, and H.-R. You. Location using los range estimation in nlos environments. In *Vehicular Technology Conference. IEEE 55th Vehicular Technology Conference. VTC Spring 2002 (Cat. No.02CH37367)*, volume 2, pages 856–860 vol.2, 2002.

[10] Wuk Kim, Gyu-In Jee, and JangGyu Lee. Wireless location with nlos error mitigation in korean cdma system. In *Second International Conference on 3G Mobile Communication Technologies*, pages 134–138, 2001.

[11] Man Lok Fung, Michael Z. Q. Chen, and Yong Hua Chen. Sensor fusion: A review of methods and applications, 2017.

[12] Gary Bishop Greg Welch. An introduction to the kalman filter, 2006.

[13] K. M. Lynch H. Choset. Principles of robot motion, 2005.

[14] Gotposition ps (ping-from-sky), Last accessed: 2022. https://gotposition.com/index.php/technology_overview/.

[15] Rplidar a1 spec., Last accessed: 2022. https://www.slamtec.com/en/Lidar/A1Spec.

[16] Adam Bry, Abraham Bachrach, and Nicholas Roy. State estimation for aggressive flight in gps-denied environments using onboard sensing. In *2012 IEEE International Conference on Robotics and Automation*, pages 1–8, 2012.

[17] Nathaniel E. Helwig. Two sample t test, 2020. http://users.stat.umn.edu/~helwig/notes/TwoSampleTest.pdf.