

DOCTORAL THESIS

Enhancing UWB and Multi-Sensor Positioning with ML-based Uncertainty Estimation

Mihkel Tommingas

TALLINN UNIVERSITY OF TECHNOLOGY
DOCTORAL THESIS
35/2025

Enhancing UWB and Multi-Sensor Positioning with ML-based Uncertainty Estimation

MIHKEL TOMMINGAS



TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Thomas Johann Seebeck Department of Electronics

The dissertation was accepted for the defence of the degree of Doctor of Philosophy (Information and Communication Technology) on 7 May 2025

Supervisor: Professor Muhammad Mahtab Alam,
Thomas Johann Seebeck Department of Electronics,
School of Information Technologies,
Tallinn University of Technology,
Tallinn, Estonia

Co-supervisor: Dr. Ivo Mürsepp, PhD,
Thomas Johann Seebeck Department of Electronics,
School of Information Technologies,
Tallinn University of Technology,
Tallinn, Estonia

Co-supervisor: Dr. Taavi Laadung, PhD,
Eliko Tehnoloogia Arenduskeskus OÜ,
Tallinn, Estonia

Opponents: Professor Susana Sargento,
University of Aveiro,
Aveiro, Portugal

Professor Luca Reggiani,
Politecnico di Milano,
Milano, Italy

Defence of the thesis: 13 June 2025, Tallinn

Declaration:

Hereby I declare that this doctoral thesis, my original investigation and achievement, submitted for the doctoral degree at Tallinn University of Technology, has not been submitted for any academic degree elsewhere.

Mihkel Tommingas

signature

Copyright: Mihkel Tommingas, 2025
ISSN 2585-6898 (publication)
ISBN 978-9916-80-307-3 (publication)
ISSN 2585-6901 (PDF)
ISBN 978-9916-80-308-0 (PDF)
DOI <https://doi.org/10.23658/taltech.35/2025>
Printed by Koopia Nlini & Rauam

Tommingas, M. (2025). *Enhancing UWB and Multi-Sensor Positioning with ML-based Uncertainty Estimation* [TalTech Press]. <https://doi.org/10.23658/taltech.35/2025>

TALLINNA TEHNIKAÜLIKOOL
DOKTORITÖÖ
35/2025

Ülilairiba ja mitme sensoriga positsioneerimissüsteemide täpsuse parandamine masinõppe meetodil

MIHKEL TOMMINGAS

Contents

List of Publications	7
Author's Contributions to the Publications	8
Abbreviations.....	9
1 Introduction	11
1.1 Ultra-Wideband sensor background	11
1.2 GNSS sensor background	12
1.3 Problem statement and research questions	13
1.4 Thesis organization.....	14
2 Theoretical background of thesis contributions.....	15
2.1 Geometric positioning	15
2.2 Position estimation.....	17
2.3 Ranging residuals.....	20
2.4 Unified frame of reference for multi-sensor systems.....	21
2.4.1 Discussion.....	23
2.5 ML methods used in the contributions	24
2.5.1 Regression Tree.....	25
2.5.2 Random Forest	26
2.5.3 Extreme Gradient Boosting	28
2.5.4 Discussion.....	30
2.6 Features used in the contributions	31
2.6.1 Features for UWB positioning uncertainty estimation	31
2.6.2 Features for GNSS positioning uncertainty estimation	33
2.7 GNSS positioning correction methods	34
2.7.1 Principles of DGNSS	34
2.7.2 Principles of GNSS RTK	34
3 Related work	36
3.1 Integrity estimation for UWB localization.....	36
3.1.1 Application of ranging residuals	36
3.1.2 Application of channel impulse response	37
3.1.3 Application of signal parameters	38
3.2 Integrity estimation for GNSS and multi-sensor localization	39
3.2.1 Geofencing.....	41
3.2.2 Positioning data distribution	42
3.2.3 Dilution of precision.....	42
3.3 ML application in related works	43
3.3.1 Machine learning in GNSS positioning	43
3.3.2 Machine learning in UWB positioning.....	45
3.4 Research gaps	47
4 UWB positioning accuracy classification using ML	48
4.1 Background and motivation	48
4.2 ML model training.....	49
4.3 ML model testing.....	51
4.4 Discussion	51

5	UWB end coordinate correction using ML	53
5.1	Background and motivation	53
5.2	ML model training	53
5.2.1	Regression tree model training	54
5.2.2	Random Forest model training	55
5.2.3	XGBoost model training	56
5.3	Coordinate filtering	57
5.3.1	KF and AKF filtering	58
5.3.2	EKF filtering	58
5.4	ML model testing and application for filtering	60
5.5	Discussion	63
6	GNSS and UWB sensor fusion with ML-based uncertainty estimation	64
6.1	Background and motivation	64
6.2	Data collection	65
6.3	ML model training	67
6.4	ML model testing	68
6.5	Sensor fusion and filtering	68
6.6	Practical experiments and results	70
6.6.1	Repeatability tests	73
6.7	Discussion	74
7	Conclusion	77
7.1	Summary	77
7.2	Research questions	78
7.3	Future work	79
	List of Figures	83
	List of Tables	84
	References	85
	Acknowledgements	96
	Abstract	97
	Kokkuvõte	99
	Appendix 1	101
	Appendix 2	109
	Appendix 3	127
	Curriculum Vitae	142
	Elulookirjeldus	144

List of Publications

This PhD thesis is based on the following publications that are referred to in the text by Roman numbers.

- I M. Tommingas, S. Ulp, M. M. Alam, I. Mürsepp, and T. Laadung, "Estimating UWB Positioning Integrity Based on Ranging Residuals," in *2023 24th International Conference on Applied Electromagnetics and Communications (ICECOM)*, pp. 1–5, IEEE, 2023
- II M. Tommingas, M. M. Alam, I. Mürsepp, and S. Ulp, "UWB Positioning Integrity Estimation Using Ranging Residuals and ML Augmented Filtering," *IEEE Journal of Indoor and Seamless Positioning and Navigation*, vol. 2, pp. 205–218, 2024
- III M. Tommingas, T. Laadung, S. Varbla, I. Mürsepp, and M. Mahtab Alam, "UWB and GNSS Sensor Fusion Using ML-Based Positioning Uncertainty Estimation," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 2177–2189, 2025

Author's Contributions to the Publications

- I **Publication I:** In this article, I was the lead author. I investigated the feasibility of using alternative methods for Ultra-Wideband (UWB) positioning uncertainty estimation. I wrote the code for UWB data processing: calculations of features and positioning estimations. I used the training data to develop models based on proposed features. I used the Extreme Gradient Boosting (XGBoost) Machine Learning (ML) library to train and test the model in RStudio environment. I wrote the manuscript and prepared the figures and tables. I revised the manuscript according to comments from my supervisors and reviewers.
- II **Publication II:** In this article, I was the lead author. I conducted UWB measurement campaigns for additional data collection. I wrote the code for UWB data processing: calculations of features and positioning estimates. I used the training data to develop datasets based on the proposed features. I used ML libraries to train and test the Regression Tree, Random Forest, and XGBoost models. I wrote the Adaptive Kalman Filtering (AKF) code and the ML implementation scheme. I conducted positioning performance tests for different filtering approaches. I wrote the manuscript and prepared the figures and tables. I revised the manuscript according to comments from my supervisors and reviewers.
- III **Publication III:** In this article, I was the lead author. I investigated the feasibility of alternative methods for estimating Global Navigation Satellite System (GNSS) positioning uncertainty. I wrote the code for GNSS data extraction and processing. I used the training data to develop datasets based on the proposed features. I used the XGBoost ML library to train and test the GNSS ML model. I conducted the GNSS measurement campaign and sensor fusion tests. I wrote the manuscript and prepared the figures and tables. I revised the manuscript according to comments from my supervisors and reviewers.

Abbreviations

2D	Two-Dimensional
3D	Three-Dimensional
AF	Adjustment Factor
AKF	Adaptive Kalman Filter
AML	Approximate Maximum Likelihood
AoA	Angle of Arrival
CART	Classification and Regression Tree
CEP	Circular Error Probable
CIR	Channel Impulse Response
CNN	Convolutional Neural Networks
CNR	Carrier to Noise Ratio
COTS	Commercial Off-The-Shelf
CS	Channel Statistics
DoP	Dilution of Precision
DR	Dead Reckoning
DRMS	Distance Root Mean Square
DT	Decision Tree
ECEF	Earth-Centered, Earth-Fixed
EKF	Extended Kalman Filter
ENU	East-North-Up
GNSS	Global Navigation Satellite System
HDoP	Horizontal Dilution of Precision
IMR	Iterative Minimum Residual
IMU	Inertial Measuring Unit
IPS	Indoor Positioning System
INS	Inertial Navigation System
KF	Kalman Filter
LCC-Rwgh	Lower-Cost Residual Weighting
LoS	Line-of-Sight
LS	Least Squares
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
ML	Machine Learning
MLE	Mean Location Error
MPC	Multi-Path Components
MSE	Mean Squared Error
NLoS	Non-Line-of-Sight
NLS	Non-Linear Least Squares
NMEA	National Marine Electronics Association
NN	Neural Networks
PPP	Precise Point Positioning
RECO	Robust Estimation for Correlated Observations
RF	Radio Frequency
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
RQ	Research Question
RSS	Received Signal Strength

RT	Regression Tree
RTK	Real-Time Kinematics
RTLS	Real-Time Location System
Rwgh	Residual Weighting
SRwgh	Select Residual Weighting
SSQ	Sum of Squares
ST	Subtopic
SVM	Support Vector Machine
TDoA	Time Difference of Arrival
ToA	Time of Arrival
UTM	Universal Transverse Mercator
UWB	Ultra-Wideband
WiFi	Wireless Fidelity
WLS	Weighted Least Squares
XGBoost	Extreme Gradient Boosting

1 Introduction

Positioning systems are technologies that can be used to determine the location of an object in space [4]. These systems provide automatic object location detection, which is then applied in various other depending technologies and applications [5]. The last decades have been marked by continuous innovation and refinement in various indoor and outdoor positioning systems with ongoing research and development paving the way for more accurate, reliable, and flexible localization solutions. Although the Global Navigation Satellite System (GNSS) has become the industry standard for accurate outdoor positioning, its unfeasible application in indoor areas has seen the development of alternative solutions based on various Radio Frequency (RF) technologies [6]. In particular, Ultra-Wideband (UWB) technology has been widely implemented in positioning solutions and is considered by many state-of-the-art surveys to be arguably the most promising RF-based solution for indoor positioning [4], [7], [8]. Moreover, combining different positioning technologies (e.g., indoor and outdoor) into a multi-sensor solution, while preserving high accuracy and stability, still remains one of key challenges. This entails extensive research and innovation, especially during recent years with advancements in artificial intelligence and the application of Machine Learning (ML) techniques for improved performance of positioning technologies. The main advantage of ML-based approaches is the ability to make decisions effectively using observed data without explicit mathematical formulation. Compared to traditional statistical methods, ML techniques enable the identification of complex dependencies in data that may not be apparent through exploratory data analysis [9]. Similarly, this thesis aims to develop distinct ML models with the goal to enhance object positioning algorithms and therefore improve overall localization performance with indoor and multi-sensor positioning systems.

On the practical level, this PhD thesis presents industry application-driven research, which sees the use of wireless indoor and multi-sensor positioning systems and enhancing their performance. In collaboration with Eliko Tehnoloogia Arenduskeskus OÜ, the PhD topic is based on real-life industry challenges of real-time object tracking. With regards to wireless positioning, industrial areas are typically cluttered with obstructions between UWB sensors, that can significantly affect the positioning performance. Furthermore, production areas and warehouses usually extend also to outdoor areas necessitating the need for fusing the existing Eliko UWB Real-Time Location System (RTLs), with add-on solutions (e.g., GNSS). By using the measurement data gathered from real-life environments, this thesis aims to enhance the positioning performance for both UWB positioning and multi-sensor networks.

1.1 Ultra-Wideband sensor background

UWB is arguably one of the most precise RF-based technologies for indoor localization and low-power personal area networks [10]. Compared to other RF-based Indoor Positioning Systems (IPS), UWB is also one of the most accurate non-hybrid positioning technologies, capable of achieving decimeter-level accuracy [11]. In contrast, hybrid positioning systems often combine data from multiple sources to improve accuracy and coverage.

The UWB RF signal employs a wide bandwidth, resulting from short, sub-nanosecond-duration pulses in the time domain. This enables superior temporal resolution, leading to a more accurate time and therefore more accurate distance measurements [8]. Short pulses also facilitate Time of Arrival (ToA) determination for burst transmission between the transmitter and receiver - a significant advantage over other indoor positioning techniques [12]. Additionally, brief transmission of pulses allows for easier differentiation be-

tween direct and reflected signals [4]. A wide bandwidth makes the RF signal more robust with several benefits: it is less sensitive to multi-path effects, offers resistance against signal interference, and allows for high-rate data transmission [13]. Furthermore, the signal can pass through walls, equipment, and other obstacles except being severely degraded by metallic and conductive liquid materials [14], [15].

Due to its low power consumption and short transmission pulses, UWB is well-suited for body-centric and wearable networks, minimizing potential health concerns [16]. Furthermore, the low-power impulses limit interference with other radio systems [14].

The main limitations of UWB technology are its relatively short range, complex infrastructure and installation, Non-Line-of-Sight (NLoS) challenges, and high cost [16], [17]. Providing an accurate position in a large and cluttered industrial indoor area would require a dense network of sensors, which in turn results in high installation costs. This approach contrasts to positioning systems that apply pre-installed wireless infrastructure (e.g., Bluetooth or WiFi networks). Additionally, systems that use Time Difference of Arrival (TDoA) with very short RF signal pulses, may experience synchronization overhead, which can necessitate significant synchronization between the anchors [18].

A distinctive constraint is also related to the application UWB channel statistics data, which is commonly used for NLoS detection and mitigation schemes. For example, gathering and transferring Channel Impulse Response (CIR) information entails in a significant latency of approximately one second, making it unfeasible to be used in a high update rate positioning system [19]. This limitation has seen the development of alternative solutions, which investigate possibilities of employing raw ranging information for improving UWB-based positioning. A more in-depth discussion regarding UWB-based features, different methods for positioning uncertainty estimation and research gaps is presented in the related works in Sections 2.6.1, 3.1 and 3.4.

Despite these limitations, UWB remains a promising technology for indoor positioning and tracking, with ongoing research aimed at addressing its challenges and expanding its capabilities.

1.2 GNSS sensor background

GNSS has become the industry standard for accurate outdoor positioning, recognized by system integrators as a mature technology. GNSS provides accurate, continuous, world-wide, three-dimensional position and velocity information to users with appropriate receivers [20]. This field evolves continuously, with ongoing research to enhance accuracy, reliability, and service continuity to meet emerging demands in safety-critical applications and location-based services [21].

To improve accuracy and precision in operational environments, GNSS incorporates various techniques such as Differential GNSS (DGNSS), Precise Point Positioning (PPP), GNSS Real-Time Kinematic (RTK), and Space- or Ground-Based Augmentation Systems (SBAS/GBAS). The choice of technique depends on factors like required accuracy, infrastructure availability, and cost. Table 1 shows how these techniques compare in terms of typical positioning accuracy.

However, GNSS positioning is not without limitations, particularly in challenging conditions. GNSS radio signals are significantly influenced by the propagation environment [21]. As GNSS relies on Line-of-Sight (LoS) transmission between the receiver and satellites, any obstructions or impediments can degrade the overall accuracy and precision. Furthermore, ionospheric and tropospheric delays, interference, number of servicing satellites, multipath effects, and signal obstructions can further compromise positioning, especially in dense urban and indoor areas, often leading to inaccurate estimates or signal loss [24].

Table 1: Comparison of techniques for improving GNSS positioning performance.

Technique	Accuracy
DGNSS	Centimeter level and up to 10 m [21]
RTK	Sub-centimeter/Centimeter level [22]
PPP	Centimeter/Decimeter level [22]
SBAS/GBAS	Sub-meter/Meter level [23]

It should be noted that, in this thesis, sensor operation environments are defined as indoor, outdoor, and transition areas. The latter marks the zone between indoor and outdoor areas (e.g., near the edge of a building).

Degraded GNSS performance indoors or in transition areas is one of the main motivations for research in multi-sensor solutions. In this thesis, GNSS is used together with UWB positioning technology to provide seamless positioning awareness regardless of operational area (e.g., indoor or outdoor scenario). Multi-sensor positioning in transition zones is considered particularly challenging as the positioning performance of both indoor and outdoor sensors is usually degraded. Furthermore, accurate assessment of the sensors' positioning integrity in different areas is difficult with dynamic changes in positioning conditions. Various techniques have been used to estimate GNSS positioning performance in sensor fusion solutions. Dilution of Precision (DoP), for instance, quantifies the impact of satellite geometry on position error, essentially representing the geometric uncertainty of the estimated position. However, GNSS positioning performance also depends on several other key factors, as mentioned in the previous paragraph. Therefore, one integrity parameter (e.g., DoP) may not be sufficient to comprehensively describe positioning performance in different operational environments. This thesis investigates different methods and their limitations in assessing GNSS positioning uncertainty and proposes an alternative approach in the form of an ML model.

A more in-depth discussion regarding GNSS features, different methods for positioning uncertainty estimation and research gaps is presented in the related works in Sections 2.6.2, 3.2 and 3.4.

1.3 Problem statement and research questions

As previously described, the methods used for improving the performance of positioning are not without limitations, this motivates the need for further research. For UWB systems, the main problem lies with developing alternative solutions in estimating UWB positioning integrity without the knowledge of channel statistics. Whereas GNSS positioning performance is significantly affected by different internal and external factors. Therefore, GNSS positioning performance may not be adequately reflected by one distinct feature. Considering the background and motivation of this thesis, the following hypothesis can be formulated:

"A set of sensor-related features, that indirectly reflect positioning quality, can be used to develop distinct ML models that could give a more comprehensive and accurate representation of uncertainty for the current coordinate. This estimate could be applied in adaptive filtering to improve the accuracy and precision of a positioning sensor."

The goal of this thesis is to provide solutions that could improve the positioning performance of a UWB and a multi-sensor positioning system. Therefore, estimating positioning uncertainty can be considered as one of the key challenges. Furthermore, implementing alternative ML-based methods would lead to identifying relevant subtopics (ST):

1. **ST1** Positioning uncertainty estimation;
2. **ST2** Machine learning applications in positioning systems;
3. **ST3** Coordinate correction with filtering;
4. **ST4** Multi-sensor positioning algorithms;
5. **ST5** Seamless indoor-outdoor positioning.

Based on the hypothesis and relevant subtopics, the following research questions (RQ) are formulated:

1. **RQ1** How to assess UWB positioning uncertainty without the knowledge of channel statistics and is the alternative approach in estimating the uncertainty feasible to be used in a high update rate positioning system? (ST1)
2. **RQ2** How to improve UWB coordinate accuracy and precision with supervised learning (e.g., Regression Trees, Neural Networks, Support-Vector Machines)? (ST2, ST3)
3. **RQ3** How to achieve a more comprehensive GNSS positioning uncertainty estimate to improve the positioning performance? (ST1, ST4)
4. **RQ4** How to augment multi-sensor fusion with ML for improved positioning performance in indoor-outdoor and obstructed environments? (ST4, ST5)

1.4 Thesis organization

The thesis is organized into seven distinct chapters, including this Chapter 1. Chapter 2 is intended to provide the reader with the main theoretical background behind the research and the contributions. Chapter 3 describes related work in the state of the art. Chapter 4 provides the method for ML-based coordinate accuracy classification in UWB positioning. Chapter 5 describes UWB end coordinate correction using ML-based uncertainty estimation with coordinate filtering. Chapter 6 discusses the main methods regarding GNSS and UWB ML-augmented sensor fusion. Lastly, Chapter 7 concludes the thesis with a short summary, answers to research questions, and potential perspectives for future research.

2 Theoretical background of thesis contributions

This thesis applies theory from different research areas including geometric and geodetic position estimation, machine learning, and localization using UWB and GNSS technologies. The following sections describe the background of most prevalent positioning methods and their mathematical models. In addition, the ML methods and their input features used in the contributions (Chapters 4, 5, 6) are explained in more detail.

2.1 Geometric positioning

Geometric positioning can be categorized into three main techniques: lateration, angulation, and signal strength-based methods [21]. The first two methods assume LoS propagation between an object and reference stations [25]. Lateration, also known as range-based positioning, estimates an object's position by applying distances or distance differences from multiple reference points and it can be either circular or hyperbolic, respectively. For the latter, distance differences are used based on time delays between signals, while circular lateration considers signals Time of Arrival (ToA). Distances d are derived by multiplying signals propagation velocity c by the measured propagation time t :

$$c \cdot t = d. \quad (1)$$

Circular lateration depicts ranges as circles (in 2D positioning) or spheres (in 3D positioning), centered at fixed coordinates of the reference points, as can be seen in Fig. 1. By measuring ToA between each previously known reference point and the object, the resulting distances d_i (derived with (1)) and reference point coordinates (x_i, y_i) can be used to estimate the position at (x, y) . It should be noted that usually, for a single solution in 2D space, at least three-, and in 3D space four reference points are required [26]. Furthermore, (x_i, y_i) should not be collinear as this may result in a flip ambiguity with possible solutions on either side of the line [27].

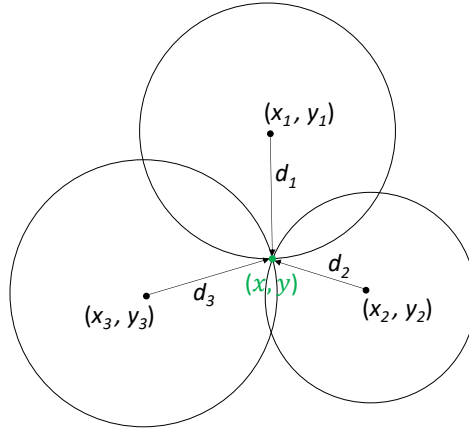


Figure 1: A 2D trilateration (a variant of multilateration) scheme showing three reference points at (x_i, y_i) and distances d_i to the object at coordinates (x, y) . As the ranges intersect at only one point, the object's coordinates can be calculated. This figure illustrates an ideal scenario, without ranging error. Figure from Publication I.

Circular multilateration can be described as a system of non-linear equations, which are solved for coordinates (x, y) :

$$(x_i - x)^2 + (y_i - y)^2 = d_i^2, \quad i = 1, \dots, N, \quad (2)$$

where N is the number of reference points used in 2D position estimation. This equation can be augmented with z coordinates for 3D positioning. However, this usually requires at least four reference points to obtain a unique solution. The main limitation in a ToA-based UWB positioning system is its power inefficiency as the tag has to negotiate ranging information separately with each individual anchor in range [28].

In hyperbolic lateration, the focus is on using TDoA to determine the relative position of an object. Unlike absolute ToA measurements, TDoA incorporates time difference between a signal's arrival at multiple reference points. For each TDoA measurement, the object's position lies on hyperbolas with a constant range difference between the measuring units [5]. Precisely measured time differences are converted into distances which can be visualized as intersecting hyperbolas as shown in Fig. 2. Additionally, in hyperbolic lateration, the object's position may not always be uniquely determined depending on the geometrical conditioning of the reference points [29]. Multiple hyperbolas can intersect at more than one definite point, leading to multiple solutions that could produce the same set of time differences.

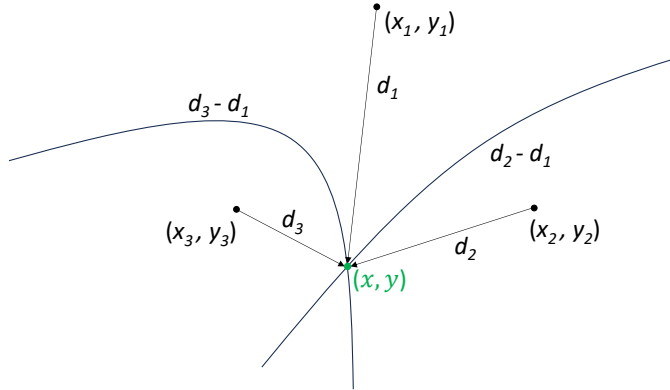


Figure 2: An illustration of a 2D TDoA lateration scheme without ranging noise. Time-synchronized reference points measure the signal propagation time differences from an object. These differences, converted to distance differences (hyperbolas), are used with the reference point coordinates to determine the object's location at (x, y) .

Mathematical representation of finding the TDoA position at (x, y) is following [5]:

$$\sqrt{(x_i - x)^2 + (y_i - y)^2} - \sqrt{(x_1 - x)^2 + (y_1 - y)^2} = d_i - d_1, \quad i = 2, \dots, N, \quad (3)$$

where N is the number of reference points.

In TDoA lateration, the reference nodes have to be precisely synchronized with each other as an offset of 1 ns translates to an error of 30 cm [11]. TDoA offers a more efficient approach to positioning compared to ToA-based systems. Although the latter requires multiple transmissions between the anchors and the tag, TDoA uses a single tag broadcast, enabling positioning with significantly lower power consumption [28].

In the angulation approach, an object's position is determined at the intersection of angle lines of bearing i.e., Angle of Arrival (AoA). Each line is formed by the circular radius from a reference point to the object [5]. This method utilizes directional antenna arrays to measure the direction of incoming signals [11]. As illustrated in Fig. 3, only two reference points with their respective angle measurements are required to estimate the position in 2D. For 3D positioning, these two reference points need to provide also the elevation angles.

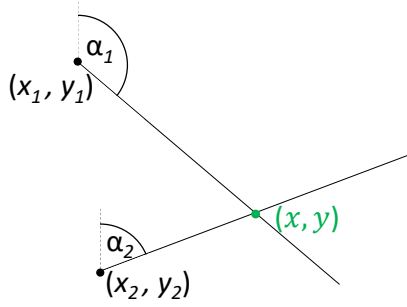


Figure 3: A 2D AoA scheme without ranging noise. The directions α_i of the signal from (x, y) are measured by directional antenna arrays at (x_i, y_i) , which are then used to estimate the position.

Mathematical representation of finding the AoA position at (x, y) is following [11]:

$$\frac{y - y_i}{x - x_i} = \tan(\alpha_i), \quad i = 1, \dots, N, \quad (4)$$

However, AoA-based positioning also has some limitations as directional measurements require costly antenna arrays [21]. Additionally, with a practical number of antennas, the achievable estimated position accuracy with ToA is superior to AoA methods [30].

In UWB-based localization, Received Signal Strength (RSS) can also be integrated for various positioning schemes. It can be used for fingerprinting techniques with signal strength maps with location estimation based on signal characteristics [29]. For instance, a database can store RSS patterns from specific subareas, and the device's estimated position is determined by matching the current signal pattern to the closest database entry [31]. Additionally, signal strength can be used as a distance estimator in lateration algorithms [21], [32]. A pathloss model can be employed to relate RSS to distance, which is then used in a ranging formula for a lateration scheme [31]. However, RSS is not very suitable for UWB localization as it does not fully leverage UWB's significant bandwidth [21]. Additionally, implementing pathloss models or storing RSS patterns is cumbersome and area specific as signal propagation depends on the physical environment. More specifically, changes in the area with existing obstructions may cause the fingerprinting database to become outdated. Finally, RSS-based positioning is not meant for accurate positioning but rather as a means of awareness for location-based services [31].

Considering the superior accuracy of ToA over AoA and better robustness over TDoA, this thesis applied ToA-based multilateration to estimate the position of the UWB tag.

2.2 Position estimation

While geometric techniques offer an intuitive approach for position estimation in ideal, noise-free conditions, they do not represent a systematic approach using noisy measure-

ments [29]. In practice, ranging measurements are subject to noise, which results in significant positioning uncertainty. This scenario is illustrated in Fig. 4, where the object can be located in the vicinity of overlapping ranging circles. Such multilateration problem can be solved with different methods such as: analytical, Least Squares (LS), Taylor series, two-stage maximum likelihood, genetic algorithm, linear lines of position and approximate maximum likelihood (AML) among others [33], [34], [35]. In the performance comparison by Shen, Zetik and Thoma [35] it was found that for an arbitrary sensor node topology, Taylor series method and AML offered accurate location estimation at intermediate noise levels and provided the best tradeoff between average error and failure rate compared to other methods. On the other hand, Taylor series method has been criticized for its computational ineffectiveness (recursive algorithm) and convergence towards a local minimum if the initial guess is too far from the true position [35], [36]. Nevertheless, in this thesis Taylor series method was implemented to approximate and linearize the non-linear equations.

In all the contributions of this thesis, the coordinate calculation of the tag is considered as a two-step process. Firstly, estimating the initial position of the tag with LS and then optimizing the solution with a Non-Linear Least Squares (NLS) approach. Both involve solving the multilateration problem using ToA ranging measurements discussed in the previous section. Additionally, this thesis considers UWB positioning in 3D space.

In Step 1 a set of equations is used to find the initial estimate of the tag's position $(\hat{x}, \hat{y}, \hat{z})$:

$$(x_i - \hat{x})^2 + (y_i - \hat{y})^2 + (z_i - \hat{z})^2 = d_i^2, \quad i = 1, 2, \dots, N, \quad (5)$$

where (x_i, y_i, z_i) is the known coordinates of the i -th UWB anchor and d_i is the measured distance between the UWB tag and the i -th anchor. The initial guess of the tag's position $(\hat{x}, \hat{y}, \hat{z})$ can be found by performing linearization on (5) and applying the LS method. Firstly, an anchor (x_r, y_r, z_r) with the shortest measured distance to the tag d_r is taken as a reference point [37]. Next, the non-linear expressions in all available equations N are expanded as:

$$x_i^2 - 2x_i\hat{x} + \hat{x}^2 + y_i^2 - 2y_i\hat{y} + \hat{y}^2 + z_i^2 - 2z_i\hat{z} + \hat{z}^2 = d_i^2, \quad i = 1, 2, \dots, N \quad (6)$$

and the reference anchor (x_r, y_r, z_r) equation:

$$x_r^2 - 2x_r\hat{x} + \hat{x}^2 + y_r^2 - 2y_r\hat{y} + \hat{y}^2 + z_r^2 - 2z_r\hat{z} + \hat{z}^2 = d_r^2 \quad (7)$$

is subtracted from the rest of the expressions.

The goal is to rearrange the terms with regard to unknowns \hat{x} , \hat{y} and \hat{z} in a way that satisfies the following linear model (8) [38]:

$$\mathbf{A}\boldsymbol{\theta} = \mathbf{b}, \quad (8)$$

where

$$\mathbf{A} = -2 \begin{bmatrix} x_1 - x_r & y_1 - y_r & z_1 - z_r \\ x_2 - x_r & y_2 - y_r & z_2 - z_r \\ \vdots & \vdots & \vdots \\ x_{N-1} - x_r & y_{N-1} - y_r & z_{N-1} - z_r \end{bmatrix}, \quad (9)$$

$$\boldsymbol{\theta} = \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} \quad (10)$$

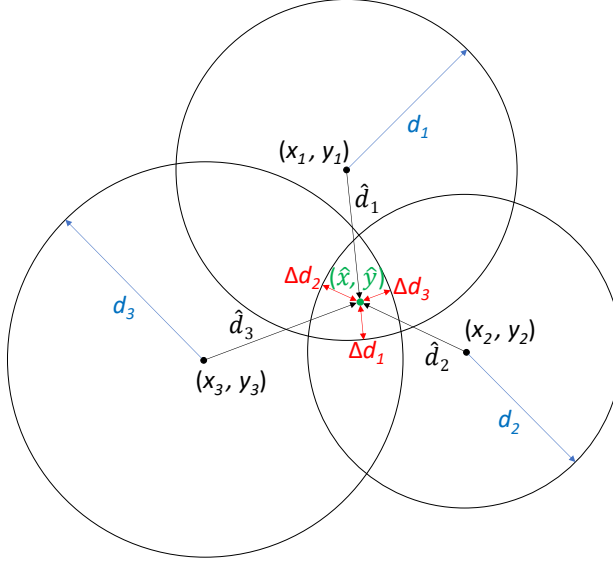


Figure 4: A 2D trilateration scheme showing fixed coordinates at (x_i, y_i) and inaccurate pseudoranges d_i . The tag's location can be estimated in the vicinity of three overlapping circles at (\hat{x}, \hat{y}) . Figure from Publication I.

and

$$\mathbf{b} = \begin{bmatrix} d_1^2 - d_r^2 - x_1^2 + x_r^2 - y_1^2 + y_r^2 - z_1^2 + z_r^2 \\ d_2^2 - d_r^2 - x_2^2 + x_r^2 - y_2^2 + y_r^2 - z_2^2 + z_r^2 \\ \vdots \\ d_{N-1}^2 - d_r^2 - x_{N-1}^2 + x_r^2 - y_{N-1}^2 + y_r^2 - z_{N-1}^2 + z_r^2 \end{bmatrix}. \quad (11)$$

The initial guess of the tag's approximate position θ has the following LS solution:

$$\theta = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}. \quad (12)$$

However, this position estimate can be further optimized with a non-linear least squares model to provide a more accurate solution. Considering that UWB positioning is done in 3D space, then the tag's estimated position at $(\hat{x}, \hat{y}, \hat{z})$ can be found by minimizing the objective function:

$$\hat{x}, \hat{y}, \hat{z} = \underset{x, y, z}{\operatorname{argmin}} \sum_{i=1}^N ((x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2 - d_i^2)^2 \quad (13)$$

where x , y and z denote the coordinates that provide the smallest error. In essence, the non-linear equations are first approximated and linearized using Taylor series. At each iteration, the gradient of the linearized error function is calculated with Gauss-Newton method.

Renaming the initial guess from the LS solution (12) as $T_G = (x_G, y_G, z_G)$, anchor coordinates as $C_i = (x_i, y_i, z_i)$, and optimal target coordinates as $\hat{T} = (\hat{x}, \hat{y}, \hat{z})$, the measured

distances d_i are approximated through first-order Taylor series expansion [26]:

$$\begin{aligned}
d_i(\hat{T}) &\approx d_i(T_G) + \frac{\partial d_i(\hat{T})}{\partial \hat{x}} \Big|_{T_G} \Delta x + \frac{\partial d_i(\hat{T})}{\partial \hat{y}} \Big|_{T_G} \Delta y + \frac{\partial d_i(\hat{T})}{\partial \hat{z}} \Big|_{T_G} \Delta z \\
&\approx d_i(T_G) + \frac{\hat{x} - x_i}{\|C_i - \hat{T}\|} \Big|_{T_G} \Delta x + \frac{\hat{y} - y_i}{\|C_i - \hat{T}\|} \Big|_{T_G} \Delta y + \frac{\hat{z} - z_i}{\|C_i - \hat{T}\|} \Big|_{T_G} \Delta z \\
&\approx d_i(T_G) + \frac{\hat{x} - x_i}{d_i(\hat{T})} \Big|_{T_G} \Delta x + \frac{\hat{y} - y_i}{d_i(\hat{T})} \Big|_{T_G} \Delta y + \frac{\hat{z} - z_i}{d_i(\hat{T})} \Big|_{T_G} \Delta z \\
&\approx d_i(T_G) + \frac{x_G - x_i}{d_i(T_G)} \Delta x + \frac{y_G - y_i}{d_i(T_G)} \Delta y + \frac{z_G - z_i}{d_i(T_G)} \Delta z,
\end{aligned} \tag{14}$$

where $\|\cdot\|$ denotes the Euclidean norm and Δx , Δy and Δz are equal to $\hat{x} - x_G$, $\hat{y} - y_G$ and $\hat{z} - z_G$, respectively. Considering that Δx , Δy and Δz are multiplied to first-order derivatives when:

$$J_i = \begin{bmatrix} \frac{x_G - x_i}{d_i(x_G, y_G, z_G)} & \frac{y_G - y_i}{d_i(x_G, y_G, z_G)} & \frac{z_G - z_i}{d_i(x_G, y_G, z_G)} \end{bmatrix}, \tag{15}$$

then (14) can be rearranged into matrix form:

$$\Delta d_{NLS} = J \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}, \tag{16}$$

with Δd_{NLS} representing the difference between measured and estimated distances:

$$\Delta d_{NLS} = d_i(\hat{x}, \hat{y}, \hat{z}) - d_i(x_G, y_G, z_G). \tag{17}$$

The error corrections Δx , Δy and Δz can be found by solving the Normal Equation as shown in (12) and substituting values accordingly:

$$\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = (J^T J)^{-1} J^T \Delta d_{NLS}. \tag{18}$$

Using the error correction vector, the initial guess coordinates x_G , y_G , and z_G are updated with Gauss-Newton iteration until a convergence criterion has been reached. This is usually set as a maximum iteration count or if the values of the error correction vector are sufficiently small [26]. After reaching a pre-determined threshold, the final position estimate results as:

$$\begin{bmatrix} x_G + \Delta x \\ y_G + \Delta y \\ z_G + \Delta z \end{bmatrix} = \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix}. \tag{19}$$

2.3 Ranging residuals

As shown in Fig. 5, a range d_i can have an offset to the estimated position. Furthermore, it can be seen that the difference between an individual measured distance d_i , and distance \hat{d}_i calculated from the estimated coordinate (\hat{x}, \hat{y}) , results in a residual Δd_i as:

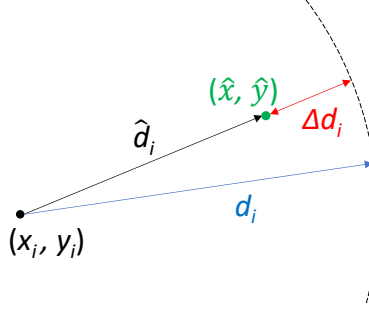


Figure 5: An inaccurately estimated position at (\hat{x}, \hat{y}) can result in an offset Δd_i between a measured range d_i and distance \hat{d}_i . The latter is calculated between the reference point (x_i, y_i) and the estimated position. The estimated position can also be located beyond a ranging measurement, resulting in a negative residual.

$$\Delta d_i = d_i - \hat{d}_i. \quad (20)$$

This error can be used to indirectly describe the quality of UWB observations (therefore the quality of estimated position) [39]. Residual information can also be used in positioning error mitigation schemes. For example, a specific distance can be discarded or a set of ranges can be dismissed if their residual magnitude exceeds a certain threshold. This approach has been used in the state of the art by many authors as will be shown in Section 3.1.1. However, discarding inaccurate ranges may lead to positioning delays or even dropouts as inaccurate position estimates could be discarded. This is usually caused by severe NLoS conditions in challenging positioning environments.

This thesis proposes an alternative use of ranging residuals. Different statistical features of residuals are used in conjunction with other parameters from state of the art, to describe UWB positioning uncertainty. Moreover, the ensemble of these features is used in an ML model which predicts the current coordinate offset. This information is then used in a coordinate filtering scheme to improve positioning precision and accuracy.

2.4 Unified frame of reference for multi-sensor systems

Locating an object in a frame of reference requires the use of coordinates, which can be assigned to a specific object of interest. In a multi-sensor RTLS, incorporating both UWB and GNSS positioning solutions, the coordinate systems are mismatched. The UWB system, primarily designed for indoor environments, operates in a local frame of reference, that has been previously established during the setup of the system. On the other hand, the GNSS system, intended for outdoor operation, provides geodetic coordinates in the format of latitude, longitude and height, which are based on a conventional terrestrial reference system [22]. To seamlessly track an object both indoors and outdoors, a unified reference frame is essential, providing accurate coordinates regardless of the positioning system. The usual approach would be to project geodetic coordinates in an area of operation, which also contains the local frame of reference. With both reference frames in the metric scale, the unified frame can be then established with coordinate rotation and shifting operations.

In the literature, various methods have been used to establish a unified frame of reference for a sensor fusion solution. For instance, Zhang *et al.* utilized Gauss-Krüger pro-

jection shown in Fig. 6, which is a variant of the transverse Mercator projection, for GNSS coordinate transformation [40]. This method is explained in more detail in [41] and [42]. The Universal Transverse Mercator (UTM), illustrated in Fig. 7, was the preferred method of coordinate projection for Di Pietra, Lv, and Rykala in [43], [44], [45], and [46].

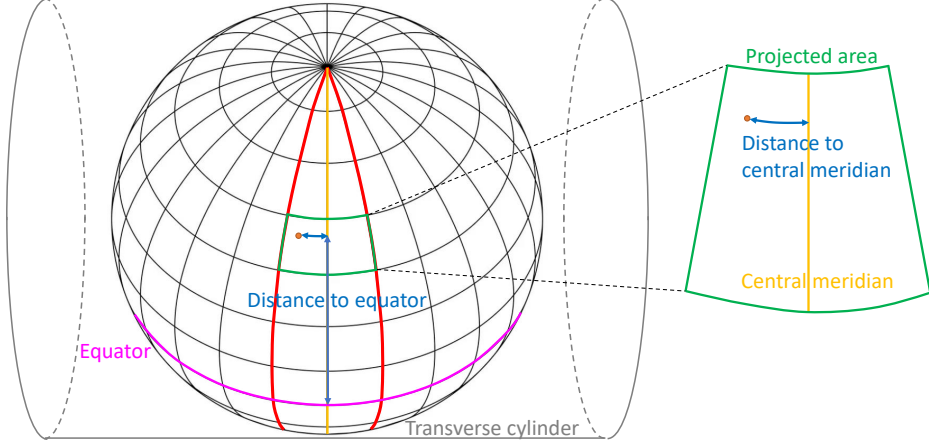


Figure 6: In the Gauss-Krüger projection, a tangent transverse cylinder, which touches along a meridian (yellow line), is used to project the ellipsoidal model of the Earth. The scale distortions grow rapidly with increasing distance from the central meridian, so the projection width is limited to 3 to 6 degrees [47].

Alternatively, spherical coordinates can be converted into geocentric Cartesian or Earth-centered, Earth-fixed coordinates, which accurately represent their spherical counterparts [48]. However, these coordinates can be further shifted and rotated onto an arbitrary local tangent plane, typically situated within the region of interest [49] as shown in Fig. 8.

This approach, referred to as East-North-Up (ENU) coordinates or local geodetic coordinate system, was used in works such as [50], [51], [52], [53], and [54]. The ENU method satisfies the need to map the coordinates in a small area on the Earth as it conforms so nearly to a plane that geometric distortion on such a system is negligible [22].

The ENU transformation is twofold. Firstly, the geodetic latitude ϕ , longitude λ , and height h values (e.g., extracted from a GNSS device) are converted into Earth-centered, Earth-fixed (ECEF) coordinate system [55]:

$$N(\phi) = \frac{a^2}{\sqrt{a^2 \cos^2 \phi + b^2 \sin^2 \phi}}, \quad (21)$$

$$X = (N(\phi) + h) \cos \phi \cos \lambda, \quad (22)$$

$$Y = (N(\phi) + h) \cos \phi \sin \lambda, \quad (23)$$

$$Z = \left(\frac{b^2}{a^2} N(\phi) + h \right) \sin \phi, \quad (24)$$

where a and b are Earth's equatorial and polar radii respectively and X, Y, Z represent ECEF coordinates. This transformation would be done using geodetic coordinates of both the point of interest (e.g., GNSS rover) and point of origin (e.g., GNSS base station), denoted by X_p, Y_p, Z_p and X_o, Y_o, Z_o , respectively. Next, these values are shifted with regard

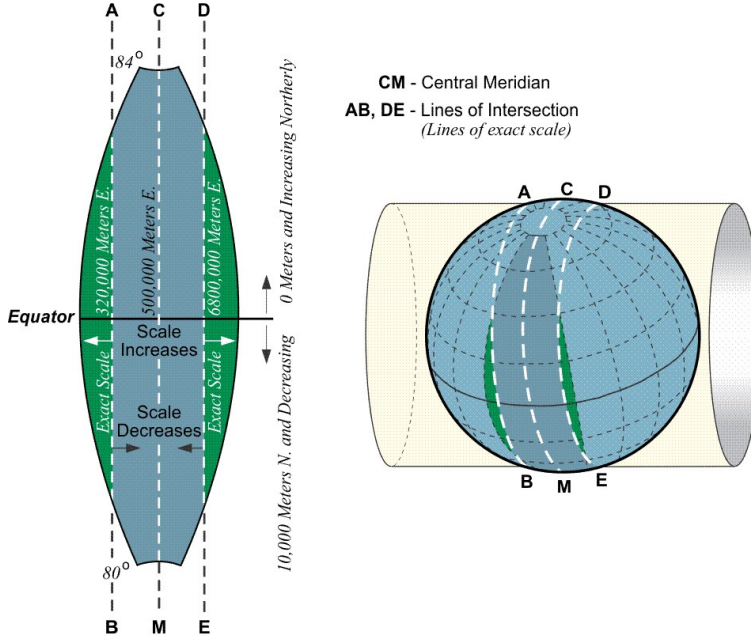


Figure 7: In the UTM projection, the cylinder touches along two standard parallels on the Earth's surface, reducing distortion near the central meridian. This figure is from [22].

to the point of origin. The vector values (E , N and U) from the origin to the point of interest are calculated as:

$$\mathbf{R}_r = \begin{bmatrix} -\sin \lambda_o & \cos \lambda_o & 0 \\ -\sin \phi_o \cos \lambda_o & \sin \phi_o \sin \lambda_o & \cos \phi_o \\ \cos \phi_o \cos \lambda_o & \cos \phi_o \sin \lambda_o & \sin \phi_o \end{bmatrix}, \quad (25)$$

$$\mathbf{R}_s = \begin{bmatrix} X_p - X_o \\ Y_p - Y_o \\ Z_p - Z_o \end{bmatrix}, \quad (26)$$

$$\mathbf{R}_r \times \mathbf{R}_s = \begin{bmatrix} E \\ N \\ U \end{bmatrix}, \quad (27)$$

where \mathbf{R}_r and \mathbf{R}_s mark the rotation and shifting matrices, respectively. ENU coordinates for the point of origin are (0, 0, 0), and the length between the origin and point of interest can be calculated with the Euclidean distance formula.

2.4.1 Discussion

As shown in the previous section, a unified frame of reference can be established with different methods. However, due to the Earth's spherical geometry, projecting it onto a flat plane inevitably introduces distortions. Moreover, different projection methods produce varying levels of accuracy depending on the size of the projected area. Thus, if the geodetic coordinates of a GNSS receiver are transformed into the local frame of reference of an indoor system, then inaccuracies will also be present for the outdoor coordinates. Furthermore, these inaccuracies tend to increase with distance from the origin of the

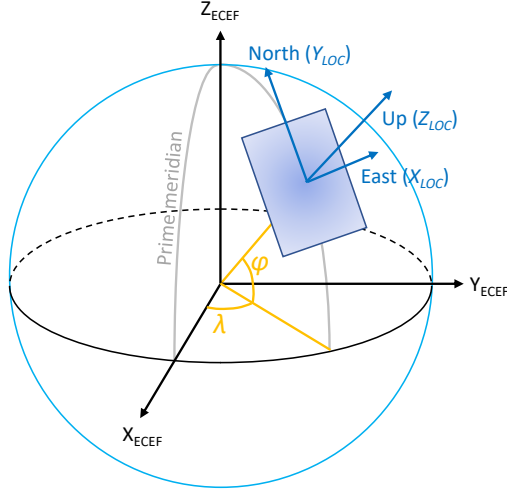


Figure 8: The ENU method maps spherical coordinates onto a flat tangent plane located at the project site. It is assumed that the flat plane represents a part of the surface on the sphere in a sufficiently small area.

projection [47]. An appropriate projection method should consider the application. For example, ENU method is often applied in robotics, when coordinate mapping is done in a small area [56]. On the other hand, UTM and Gauss-Krüger methods are suitable for mapping and surveying large regions of an area [41]. Additionally, projection algorithms often involve computationally intensive operations, including arithmetic, trigonometric, and exponentiation operations, which can contribute to coordinate calculation delays in a real-time sensor fusion system. Lastly, a well-established unified frame of reference can provide convenient and intuitive coordinates for system operators. For example, if the ENU frame is established at the positioning site, the distances and coordinates of objects can be easily interpreted from a map. In contrast, if the positioning system uses native coordinates from a projection (e.g., UTM), the multi-digit numbers may be less user-friendly for everyday use.

Considering computational effectiveness, conversion accuracy, and accessible coordinate representation, the ENU method was used in this thesis for GNSS geodetic coordinate conversions. Furthermore, considering geodetic projections in a small area, then ENU is the most suitable method for this. The theory represented in Section 2.4 is applied in Publication III for GNSS and UWB sensor fusion.

2.5 ML methods used in the contributions

In recent years, Machine Learning (ML) algorithms have been successfully applied in different localization solutions. The main advantage of such approaches is the ability to make decisions effectively using observed data without accurate mathematical formulation [9]. Compared to traditional statistical methods, ML techniques enable the identification of complex dependencies in data that may not be apparent through exploratory data analysis. While the goal is not to derive an explicit mathematical formula for the data distribution, it can effectively be used to train algorithms to learn the relation between input features and their response variables [57]. This thesis describes three distinct ML techniques

that have been used to improve the positioning performance of UWB and GNSS: Regression Tree (RT), Random Forest, and Extreme Gradient Boosting (XGBoost). In essence, ML algorithms are used to perform supervised learning to train a model using a set of previously established features and their response variables, where the latter is based on pre-measured true coordinate offset. The estimated or predicted coordinate error can then be used as a measure of uncertainty in a coordinate filtering scheme to assign appropriate weights for estimated coordinates.

2.5.1 Regression Tree

Regression trees are ML methods for constructing prediction models from data. The models are obtained by recursively partitioning the data space and fitting a simple prediction model within each partition. As a result, the partitioning can be represented graphically as a tree-like structure [58]. The algorithm assigns each sample from a dataset into a prediction based on the feature attributes of each sample [59]. The decision at a particular node of the tree, referred to as the split criterion, is typically a condition on one or more feature variables in the training data. The split criterion divides the training data into two or more parts with the goal of reducing the level of mixing of variables as much as possible. Each node in the tree predictor represents a subset of data-space defined by the combination of split criteria in the nodes above it, and the final prediction for a new sample is determined by traversing the tree based on its feature values. Therefore, the tree is constructed as a hierarchical partitioning using supervised instances during the training [60].

A simplified example of a Regression Tree is shown in Fig. 9. Let the training set be defined as $T = (x_n, y_n)$, with n observations, m features, and response variables y_n . Function $F_n = (m, t_n)$, splits the data based on threshold t_n and feature m into left and right branches. Usually, the Mean Squared Error (MSE) is used for quantitative evaluation at each split. Minimizing over function F_n , the optimal cut is achieved. Since in regression problems, the tree structure minimizes the MSE of the predictions, then the tree growing process is repeated until no improvement in the loss function (28) is attained [61]:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (28)$$

Regression Tree advantages:

- The algorithm is based on ordering and splitting the values within each feature, therefore scaling and normalization is not required;
- Visual splits of the data and ordered feature importances are easy to understand and interpret;
- Robust in terms of missing data or outliers;
- Model is built only using observed data without assumptions to data distribution;
- Creating a single Regression Tree and making predictions is computationally faster when compared to tree ensembles.

Regression Tree limitations:

- Regression Trees usually have high variance, which means that small changes to data (e.g., adding samples) can lead to large structural changes in the tree;

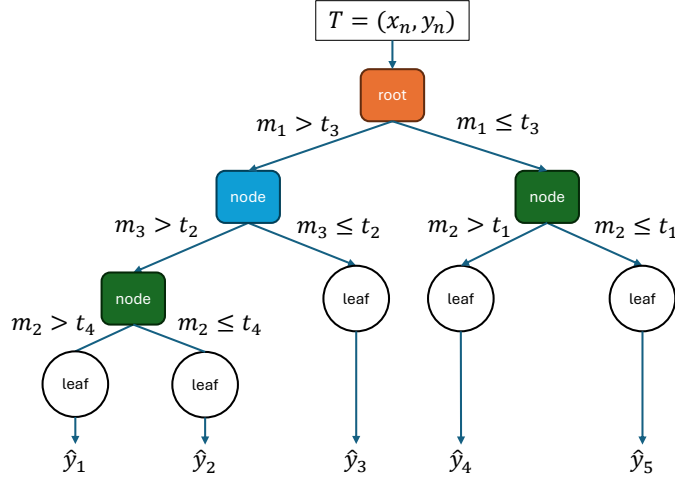


Figure 9: In the Regression Tree, the input value is traversed through a series of nodes, where data features m are compared to preset thresholds t_n at each node resulting in prediction \hat{y}_n .

- Without regularization techniques (e.g., pruning, early stopping) there is a high risk of overfitting;
- Reaching a global optimum for the model is not guaranteed;
- Regression Trees can be biased towards the majority class in imbalanced datasets.

2.5.2 Random Forest

Random Forest algorithm, developed by Leo Breiman, combines multiple tree predictors, each built using a random vector sampled independently and identically distributed across all trees [62]. It is an ML method that relies on ensemble learning to address some of the limitations of an individual Regression Tree model. Instead of relying on a single regression tree, Random Forest constructs a multitude of regression trees using bootstrapped aggregation (i.e., bagging). This involves creating multiple subsets of the training data through random sampling, meaning some samples may appear in multiple instances in a subset, while others are omitted. This resampling process helps to reduce variance, a key drawback of single regression trees. Each subset is then used to train an individual regression tree.

In order to make predictions on new data, each tree provides its own prediction. For classification, the final prediction is determined by a majority vote among the trees. For regression, the final prediction is the average of all individual tree predictions. Random Forest leverages the collective informational gain of multiple trees, leading to improved accuracy and reduced overfitting compared to a single regression tree [63], [64].

A simplified example of a Random Forest is shown in Fig. 10. Let the training set be defined as $T = (x_n, y_n)$, $x \in \mathbb{R}^m$, $y \in \mathbb{R}$, with n observations, m features, and response variables y_n . The input data for each tree is provided through bootstrap sampling, where n observations are randomly selected with replacements from T . These independent and identically distributed vectors are marked with θ_K . During the training, the algorithm splits the data at each node, so that the parameters of split functions become optimized to fit with dataset T . During this step, the regression tree has to make the best split among all

variables. The splitting process starts at the root and each node applies its own split function to the new input x . Splitting is repeated until a terminal node is reached or when a node contains less than a predefined number of observations. The ensemble of regression trees produces K outputs for prediction output \hat{y}_K . The aggregation is done by averaging the outputs of all trees [65].

Random Forest advantages:

- Usually has high accuracy and low bias;
- Low variance when compared to a single regression tree;
- Appropriate for large datasets with high dimensionality;
- Robust regarding missing data;
- Bootstrap sampling is useful when data is limited.

Random Forest limitations:

- Categorical data classification may include bias towards features with more levels;
- Can overfit data if regularization (e.g., pruning) is not used;
- Tree models are more complex and difficult to interpret than single regression tree;
- Large datasets usually produce large number of trees, which can be computationally expensive to train;
- Prediction time can be slow as it requires querying each tree in the forest.

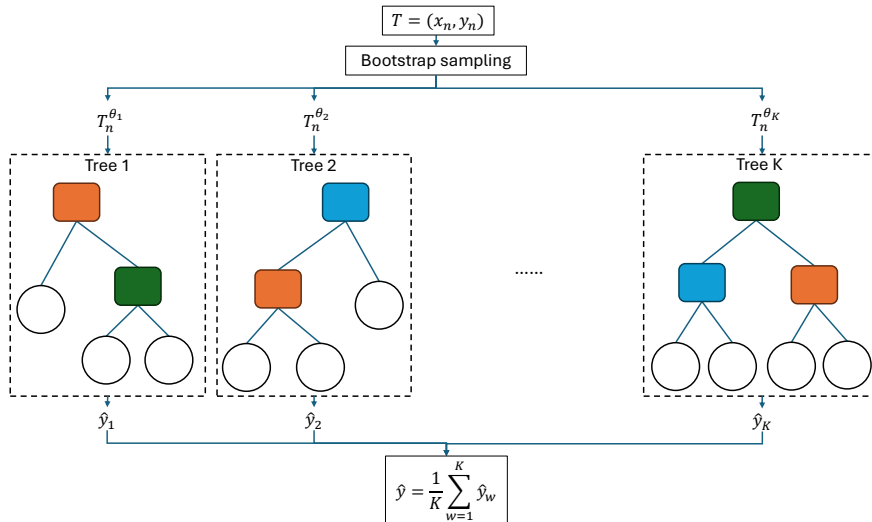


Figure 10: A simplified example of the Random Forest structure. Instead of using a single regression tree, Random Forest relies on multiple trees generated using bootstrapped training data. Each tree produces an individual prediction, which is then used to average the result.

2.5.3 Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost), is a scalable and highly effective end-to-end decision tree-based gradient boosting system, which is widely used by data scientists to achieve state-of-the-art results on many ML challenges [66]. Based on the research by Chen and Guestrin, XGBoost has been established as an open-source software library that provides a regularizing gradient boosting framework for *R*, *Java*, *Scala*, *C++*, *Python*, *Perl*, and *Julia* [67]. XGBoost represents a variant of Gradient Boosting Machine (GBM) as a technique used to tackle both regression and classification prediction problems [68].

In gradient boosting algorithms, each tree predictor is trained on data, taking into account the success of previous trees in a sequential manner. At each instant, the results of the model are weighted according to the outcome of the preceding instant. After each training iteration, the weights are redistributed so that correctly predicted outcomes are given a lower weight, and output that is misclassified increases their weights to highlight more difficult cases. In this way, subsequent learners will focus on the more weighted instances during their training [68]. XGBoost is an ensemble method where new models are created to correct the residuals of errors of prior models as shown in Fig. 11. These models are then combined to produce the final prediction. For a given data set $D = (x_i, y_i)$ with $i = 1, \dots, n$, $x_i \in \mathbb{R}^m$ and $y_i \in \mathbb{R}$, n observations and m features, a tree ensemble model uses K additive functions to predict the output \hat{y}_i .

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i), f_k \in F, \quad (29)$$

where F is the set of all possible Classification and Regression Trees (CART). At each iteration of gradient boosting, the residual will be applied to correct the previous prediction value by the k^{th} tree to the i^{th} observation. The tree ensemble model consists of a set of CART-s. XGBoost represents an optimized and efficient implementation of the gradient

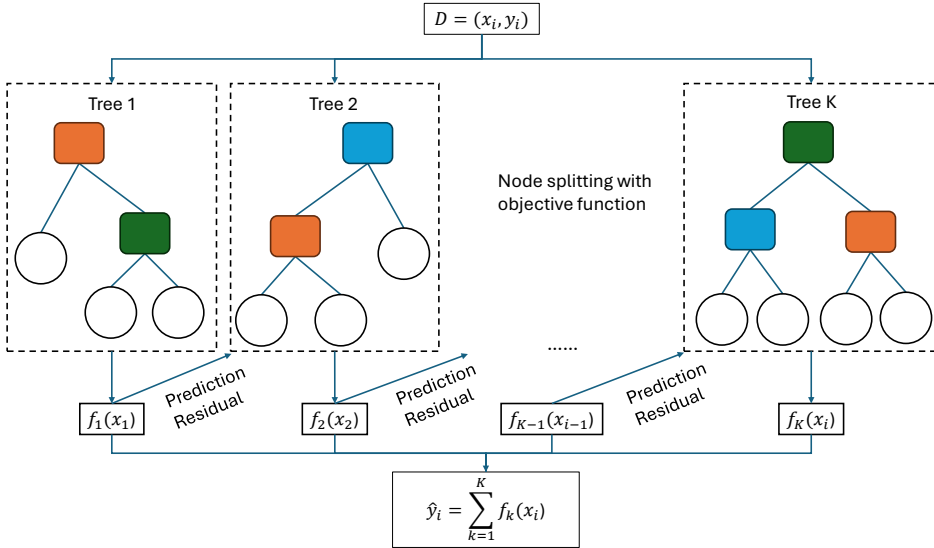


Figure 11: A general architecture of gradient boosting used in the XGBoost algorithm. The colored boxes are different features, which make up a decision/regression tree. Each subsequent tree improves the previous tree according to errors reflected by the prediction residuals. The summation of all trees results in a prediction estimate.

boosting algorithm. Compared to the latter, XGBoost incorporates regularization techniques to prevent model overfitting and improve generalization performance. It also utilizes first- and second-order gradient information to guide the tree construction process. The algorithm also makes use of parallel tree learning for faster model generation [66].

The set of functions used in the model can be learned by minimizing the regularized objective function:

$$L = \sum_{i=1}^n l(\hat{y}_i, y_i) + \sum_{k=1}^K \Omega(f_k), \quad (30)$$

where

$$\Omega(f_k) = \gamma T + \frac{1}{2} \lambda w^2. \quad (31)$$

The first term l represents the loss function and it measures the difference between the predicted value \hat{y}_i and response variable y_i . The second term Ω represents the regularization term, which is a factor for measuring the complexity of tree f_k with γ and λ being the regularization degrees, which also helps to avoid over-fitting. T and w are the number of leaves and the vector of score values to each leaf, respectively.

In order to minimize the objective function (30), function $f_t(x_i)$ that most improves the model, is added. If \hat{y}_i^t is the prediction of i -th instance at t -th iteration, then by additive manner of gradient boosting procedures the i -th prediction can also be presented as $\hat{y}_i^t = \hat{y}_i^{t-1} + f_t(x_i)$. And the objective function (30) as:

$$L^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-i)} + f_t(x_i)) + \Omega(f_t), \quad (32)$$

For optimization, a second-order approximation is applied:

$$L^{(t)} \approx \sum_{i=1}^n (l(y_i, \hat{y}_i^{(t-i)} + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)) + \Omega(f_t), \quad (33)$$

where $g_i = \partial_{\hat{y}_i^{(t-i)}} l(y_i, \hat{y}_i^{(t-i)})$ and $h_i = \partial_{\hat{y}_i^{(t-i)}}^2 l(y_i, \hat{y}_i^{(t-i)})$ are first and second order gradient statistics of the loss function, respectively. After removing constant terms, the objective function at iteration t becomes:

$$L^{(t)} = \sum_{i=1}^n (g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)) + \Omega(f_t). \quad (34)$$

which is the optimization goal for the new tree [66], [69]. By defining $I_j = \{i | q(x_i) = j\}$ as the instance set of leaf j and expanding the regularization term Ω the objective function becomes:

$$\begin{aligned} L^{(t)} &= \sum_{i=1}^n (g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)) + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \\ &= \sum_{j=1}^T ((\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2) + \gamma T. \end{aligned} \quad (35)$$

For a fixed tree structure $q(x)$, the optimal weight w_j^* for leaf j is found by:

$$w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}, \quad (36)$$

And the corresponding optimal value as:

$$L^{(t)}(q) = -\frac{1}{2} \sum_{j=1}^T \frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T. \quad (37)$$

Equation (37) can be used as a scoring function to measure the quality of the tree q similar to impurity scores for decision/regression trees. However, this function is derived for a wider range of objective functions [66].

While in practice it is impossible to enumerate all possible tree structures, a greedy algorithm, that starts from a single leaf and iteratively adds branches to the tree is used instead. The loss function after the splits for the existing tree is defined as:

$$L_S = \frac{1}{2} \left[\frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} + \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma, \quad (38)$$

where I_L and I_R are the instance sets of left and right nodes after the split. Additional techniques to prevent overfitting, such as weight scaling (shrinkage) and column subsampling are described in the original XGBoost article by Chen and Guestrin [66].

XGBoost advantages:

- Robust to imbalanced datasets;
- Gradient boosting can be optimized on many objective functions, so it can be extended to many different problem spaces;
- Optimized for computational speed;
- Has built-in regularization techniques which help to avoid overfitting and improve generalization.

XGBoost limitations:

- Requires rigorous hyperparameter tuning compared to individual Regression Trees or Random Forest;
- Sensitive to overfitting if data is noisy;
- Sensitive to outliers as each subsequent tree is trained to correct errors from previous trees;
- Complex tree models are difficult to interpret than a single Regression Tree.

2.5.4 Discussion

While the aforementioned ML methods are only a small part of existing ML techniques, these three methods were used in the contributions presented in this thesis. The reason for selecting these distinct methods were the following. Firstly, the goal was to compare different ML methods in terms of their coordinate offset prediction accuracy. This is also related to improvement of end coordinate, as the prediction was used as a measure of uncertainty in coordinate filtering. Essentially, a more accurate prediction would yield a more accurate and precise end coordinate. Secondly, the comparison was also done with

regard to computational speed. In a high position update rate positioning system it is crucial to consider measurement uncertainty at each measurement update. For example, in a UWB system with a 10 Hz update rate, the ML prediction must be done at least every 100 ms. Using a large ML model may take a longer time to process and produce a timely prediction [2]. Therefore, the goal was to compare different derivations of decision tree-based methods to produce the predictions with respect to accuracy and computational time. Considering the aforementioned advantages and limitations for each method, a simple Regression Tree would be compared to more complex methods such as Random Forest and gradient boosted trees.

2.6 Features used in the contributions

The following section describes features that were used to train the ML models and to estimate the positioning uncertainty for UWB and GNSS systems.

2.6.1 Features for UWB positioning uncertainty estimation

In order to estimate the uncertainty of a position estimate, the ML model relies on distinct established features, that have been used during the training. Usually it is done through incorporating features of low-level data such as channel state information, channel statistics, or ranging parameters to find patterns and dependencies between potential features and use this information for error detection and mitigation schemes in UWB positioning. In this thesis, the focus is on UWB features primarily related to ranging residuals, which was explained in Section 2.3.

As was shown in Fig. 5, depending on the location of the estimated position, \hat{d}_i may be longer or shorter compared to the individual measured distance d_i resulting in a positive or negative residual. A significant change in the magnitude of a residual may indicate that the propagation path of a UWB signal is affected by an obstruction [38], [39]. Therefore, residual features were calculated for three different sets: positive, negative, and overall residuals. Additionally, statistical equations contain averaging to remove the dependence on the quantity of available residuals.

Residual statistics:

average Sum of Squares (SSQ)

$$SSQ = \frac{\sum_{i=1}^n \Delta d_i^2}{n}, \quad (39)$$

Root Mean Square (RMS)

$$RMS = \sqrt{\frac{SSQ}{n}}, \quad (40)$$

mean

$$\bar{x} = \frac{\sum_{i=1}^n \Delta d_i}{n}, \quad (41)$$

Mean Absolute Deviation (MAD)

$$MAD = \frac{\sum_{i=1}^n |\Delta d_i - \bar{x}|}{n}, \quad (42)$$

standard deviation

$$s = \sqrt{\frac{\sum_{i=1}^n (\Delta d_i - \bar{x})^2}{n}}, \quad (43)$$

and variance

$$v = s^2, \quad (44)$$

where n represents the number of residuals in a corresponding positive, negative, or overall set (also used as a feature).

Number of residuals in an interval:

Small residuals indicate proximity to the NLS solution, whereas large residuals imply erroneous measurements. By counting the number of residuals in a preset range, it can be assumed whether the NLS algorithm uses accurate measurements as its input. Following ranges were chosen based on overall accuracy of UWB positioning [70], [71]: $0 \dots 0.1$ m, $0.1 \dots 0.2$ m, $0.2 \dots 0.4$ m, $0.4 \dots 0.8$ m, $0.8 \dots 1.6$ m, $1.6 \dots 3.2$ m, $3.2 \dots 6.4$ m, $6.4 \dots 12.8$ m, $12.8 \dots 25.6$ m, $25.6 \dots \infty$ m. The idea was to divide residual magnitudes into separate range categories. The quantity of residuals in each class could be then used for coordinate offset classification (used in Publication I).

LS and NLS metrics:

These values are associated with position calculations as discussed in Section 2.2. The chosen parameters include Euclidean distance ΔD between LS (x_G, y_G, z_G) and NLS $(\hat{x}, \hat{y}, \hat{z})$ solutions and the number of Gauss-Newton iterations to convergence n_{GN} . For the latter, there is no implicit equation as the iteration counter is initialized at each coordinate optimization process

$$\Delta D = \sqrt{(x_G - \hat{x})^2 + (y_G - \hat{y})^2 + (z_G - \hat{z})^2}. \quad (45)$$

Geometrical integrity of positioning:

In a positioning system, Dilution of Precision (DoP) indicates geometric uncertainty of an estimated position relative to servicing nodes (e.g., UWB anchors or GNSS satellites). It contains the knowledge of positioning accuracy under specific base station network and scene characteristics [72]. In this thesis, Position DoP (PDoP) and Horizontal DoP (HDoP) metrics were used to reflect the integrity of an estimated position. To calculate DoP, a set of ranging equations (5) can be implemented with pre-calculated end coordinates from (19). By finding partial derivatives with respect to each coordinate similarly as was shown in (14), the result is formulated in matrix form as:

$$\mathbf{W} = \begin{bmatrix} \frac{x_1 - \hat{x}}{d_1} & \frac{y_1 - \hat{y}}{d_1} & \frac{z_1 - \hat{z}}{d_1} & 1 \\ \frac{x_2 - \hat{x}}{d_2} & \frac{y_2 - \hat{y}}{d_2} & \frac{z_2 - \hat{z}}{d_2} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{x_N - \hat{x}}{d_N} & \frac{y_N - \hat{y}}{d_N} & \frac{z_N - \hat{z}}{d_N} & 1 \end{bmatrix}. \quad (46)$$

Next, the covariance matrix \mathbf{O} is calculated from the LS normal matrix:

$$\mathbf{O} = (\mathbf{W}^T \mathbf{W})^{-1} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_y^2 & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_z^2 \end{bmatrix}. \quad (47)$$

Lastly, HDoP and PDoP are calculated from the trace of matrix \mathbf{O} as:

$$HDoP = \sqrt{\sigma_x^2 + \sigma_y^2}, \quad (48)$$

$$PDoP = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}. \quad (49)$$

2.6.2 Features for GNSS positioning uncertainty estimation

In order to improve the performance of GNSS positioning in less than optimal signal environments, ML-based GNSS models have been investigated as early as the 1990s. GNSS sensors provide abundant data suitable for various ML models, depending on the specific application.

The Fieldbee L2 GNSS RTK receiver used in this research outputs data with the following National Marine Electronics Association (NMEA) 0183 message headers: \$GPGGA, \$GPGST, \$GPZDA, \$GPRMC and \$GPVTG. Table 2 presents the fields considered by the author as most relevant in describing positioning quality and therefore applied in GNSS ML model training.

Table 2: Features from NMEA messages [73].

Message	Field	Description	Symbol	Example
\$GPGGA	7	GPS quality indicator	x	4
\$GPGGA	8	Number of satellites in use	xx	11
\$GPGGA	9	Horizontal dilution of precision	x.x	1.1
\$GPGGA	14	Age of correction data (in seconds)	xx	8
\$GPGST	3	RMS value of the standard deviation of the pseudorange measurements	x.x	2.7
\$GPGST	7	Standard deviation of latitude error (m)	x.x	1.2
\$GPGST	8	Standard deviation of longitude error (m)	x.x	3.2
\$GPGST	9	Standard deviation of altitude error (m)	x.x	4.5

The GNSS device used in this research operates with three correction states, which indicator can be extracted from \$GPGGA field 7. These corrections include Differential Global Positioning System (DGPS) (value 2), floating-point RTK (value 5), and fixed RTK (value 4). **It should be noted that the message output of the receiver was limited by the manufacturer to GP headers only. Therefore, only DGPS could be used during the tests.** The correction modes are explained in more detail in Section 2.7. The receiver also outputs the precalculated solution for HDOP (\$GPGGA field 9), number of satellites in use (\$GPGGA field 8) and age of correction data (\$GPGGA field 14).

The \$GPGST log contains pseudorange measurement noise statistics that are translated into the position domain to give statistical measures of the quality of the position solution. This log reflects the accuracy of the solution type used in the \$GPGGA message, except for the RMS field, which does not represent carrier-phase-based positions but the accuracy of the pseudorange position [73].

\$GPZDA log provides Coordinated Universal Time (UTC) and date information. \$GPRMC contains time, date, position, track and speed data, while \$GPVTG log carries track-made-good and relative ground speed values. As these logs do not provide information regarding positioning quality, then they are not considered as potential features.

2.7 GNSS positioning correction methods

This section covers the main GNSS coordinate correction methods used in Publication III. These include DGPS, RTK floating-point, and RTK fix modes. In Publication III, a GNSS RTK device was used in real-life measurement campaigns and sensor fusion tests. The aforementioned correction methods all involve incorporating a GNSS base station on a coordinated control position (unless only relative coordinates are desired), which is the source of correction signals for the GNSS rover as shown in Fig. 12. The observations of both GNSS devices must be simultaneous as they must observe the same satellites and both DGPS and RTK methods rely on real-time communication between the devices [22]. The subsequent sections describe the main principles behind the correction methods used by the device.

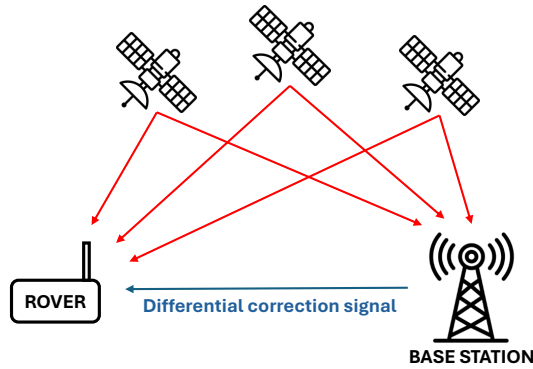


Figure 12: Differential correction-based GNSS positioning. While both DGNSS and RTK utilize corrections transmitted from a stationary GNSS base station, they differ in their method. DGNSS relies on pseudorange differences for coordinate corrections, whereas RTK incorporates carrier-phase observations.

2.7.1 Principles of DGNSS

DGNSS represents a method of relative positioning, which relies on coded GNSS pseudorange measurements. It is an extension of GNSS technology, based on satellite and terrestrial communication, and requires at least two receivers. One is established as a control station at a known location and the other device acts as a rover at an unknown location. The goal of DGNSS is to determine the errors related to pseudorange observables and is calculated by comparing the value from the GNSS receiver and the value computed using the coordinates of the satellites and the reference station [74]. Since both devices simultaneously track the same codes from the same satellites, many of the errors in the observations are common to both receivers. Therefore, the errors are correlated and tend to cancel each other to some degree [22]. Over the years, DGNSS has been remarkably improved with the possibility of meter- or even submeter accuracy. However, positioning performance is still inferior when compared to corrections with carrier-phase measurements. In dense urban areas, the DGNSS method may provide a positioning accuracy lower than 10 m [21].

2.7.2 Principles of GNSS RTK

GNSS RTK is a correction method of determining the relative location between known and unknown positions using GNSS carrier-phase measurements. It is capable of providing cm-level positioning accuracy. The goal is to employ simultaneous carrier-phase

measurements from both the rover and the base station and correct them in real-time. Differencing techniques are implemented to eliminate signal phase biases, clock offsets and atmospheric errors. Carrier-phase measurement observations depend on fixing the integer cycle ambiguity, which refers to the unknown whole number of cycles of the carrier wavelengths that have passed between the satellite and the receiver [22]. This is the key challenge in RTK-based positioning. Before the ambiguities are resolved, the GNSS receiver uses RTK floating-point estimates as a rough position estimate. Once a valid integer solution is computed, the receiver initiates the RTK fixed ambiguity solution, resulting in a significantly improved positioning accuracy [75], [76]. While the idea of using carrier-phase measurements for improving positioning performance was already developed in the 1980s, only after improvement of hardware and data processing algorithms, RTK became more widely used. Real-time positioning systems usually incorporate "on-the-fly" techniques to quickly resolve the carrier-phase ambiguities. This requires a dual-frequency receiver, capable of processing both carrier-phase and pseudorange measurements and it is not required for the receiver to remain stationary [22].

Publication III employed the Fieldbee L2 GNSS RTK receiver and base station, which are dual-frequency multi-constellation devices capable of DGPS, RTK floating-point, and RTK fixed modes [77].

The established theoretical framework surrounding different positioning algorithms and ML methods are used extensively in the existing literature. The subsequent chapter gives an overview of various related contributions, reflecting potential areas of improvement, and ultimately revealing the specific gaps that motivate the current research.

3 Related work

The subsequent sections delve into state of the art approaches for estimating positioning integrity. The sections have been categorized based on sensor application and the specific context of the thesis. Firstly, techniques for UWB-based positioning are presented (Section 3.1), followed by methods for GNSS and multi-sensor systems (Section 3.2), ML application in related works (Section 3.3), and a concluding section with research gaps is presented in Section 3.4. A discussion of each approach is presented at the end of each section.

3.1 Integrity estimation for UWB localization

This section provides an overview of the state of the art in integrity estimation regarding UWB localization. As the research presented in the thesis was conducted using ToA-based UWB positioning, the respective literature is presented. It is considered that UWB-based positioning integrity is closely tied to erroneous estimates caused by NLoS between UWB anchors and the tag [78]. Therefore, the following sections explore various state-of-the-art approaches for mitigating and detecting NLoS conditions.

3.1.1 Application of ranging residuals

Distance residual has been extensively used in NLoS identification methods as a measure of inconsistency in localization. In essence, a large ranging residual would indicate NLoS conditions, while a small residual suggests a lower ranging noise, therefore a more accurate position estimate.

Chen was among the pioneers in applying ranging residuals to detect NLoS in ToA-based localization using distinct groups of UWB anchors. It was observed that localization with LoS noise was consistent whereas the presence of NLoS resulted in larger estimated residuals. Furthermore, Chen proposed a Residual weighting algorithm (Rwgh) for NLoS mitigation, which applies a normalized residual as a weight to a position estimate [79]. Subsequent research aimed to reduce the computational complexity of this algorithm [80], [81], [82].

Jiao *et al.* investigated several residual-based algorithms for ToA positioning in terms of their performance in computational complexity and positioning accuracy. The evaluated methods included Rwgh, Iterative Minimum Residual (IMR), Select Residual weighting (SRwgh), and Lower-Computational-Cost Residual weighting (LCC-Rwgh). It was concluded that LCC-Rwgh and IMR algorithms required significantly fewer intermediate least-squares calculations leading to a smaller computational complexity for given number of range estimates. However, LCC-Rwgh performance in indoor and outdoor environments comes at a cost of robustness as it is susceptible to NLoS errors [83].

Chan *et al.* developed a residual test to simultaneously determine the number of LoS anchors, identify them, and utilize only these nodes for localization. Residual test is based on the principle that normalized residuals of LoS measurements follow a central Chi-Square distribution. While tested in simulation studies with specific anchor geometry, NLoS conditions, and ranging errors, the proposed test determined correct number of LoS anchors over 90% of the time [84]. Discarding an invalid position estimate was implemented also by Li and Wang in their research concerning a factor graph-based UWB positioning algorithm with an improved Tukey robust kernel. They suggested that the sum of residuals over a certain threshold would indicate an invalid positioning result. By utilizing only estimated positions with small residuals, a graph optimization algorithm was implemented to retain only accurate observations at different moments. The authors im-

plemented their scheme on two distinct trajectories achieving a mean positioning error of 0.66 m and 0.19 m respectively [39].

Liu *et al.* investigated NLoS detection and mitigation using the Sum of Squares (SSQ) of distance residuals in an analysis of wireless localization in NLoS conditions. A high SSQ value, compared to a certain threshold, would indicate inconsistency in localization [85]. Silva and Hanke expanded on this concept by considering additional statistical features of ranging residuals including mean, maximum, and standard deviation of sets of residuals. However, due to high correlation between these features, only SSQ was applied in a naive Bayes classifier to identify NLoS with significant accuracy. The authors carried out simulations with different sets of UWB anchor placements achieving an over 90% NLoS classification accuracy [86].

As demonstrated in the study by Jiao *et al.*, utilizing residuals as a measure for positioning inconsistency, significantly reduces positioning errors. However, this improvement comes at a cost of increased computational complexity, which grows almost exponentially with the number of input distances [83]. Some approaches that use residuals to discriminate LoS or NLoS conditions, can also present limitations. Prematurely discarding anchors with NLoS or avoiding position estimates altogether can lead to latency issues especially when considering dynamic positioning. For example, as stated in Section 2.1, usually for a single solution in 2D space, at least three, and in 3D space four reference points are required. Discarding one anchor could cause solution ambiguity. Furthermore, preset thresholds for residual magnitudes imply a non-generalized solution. For instance, a preset threshold may vary in different environments and dynamic NLoS conditions. It is worth considering that anchors with NLoS may still provide valuable ranging information for filtering purposes and should not be completely disregarded.

3.1.2 Application of channel impulse response

Detecting NLoS conditions between the UWB anchor and tag, and mitigating its effects on positioning integrity has been a central topic for various authors. By analyzing the RF signal characteristics in the UWB propagation channel, it is possible to identify LoS and NLoS scenarios. Channel Impulse Response (CIR), which describes the propagation path of a signal, can be used to assess the amplitude and phase of a particular multipath component [87]. Commonly, researchers combine NLoS identification techniques with mitigation strategies to enhance UWB positioning performance.

Guvenc *et al.* investigated NLoS identification and mitigation through the use of amplitude and delay statistics of the Multi-Path Components (MPC) in the UWB channel. It was stated that log-normal random variables can be used as a model to describe these statistics. A proposed joint likelihood ratio test was applied in a simulation to estimate if a given signal has LoS or NLoS with over 90% success rate. Subsequently, LoS likelihood values were incorporated into a Weighted Least Squares (WLS) algorithm to mitigate the NLoS effects and significantly improve average location error and measurement variance [38].

Venkatesh and Buehrer proposed a statistical NLoS identification technique based on the hypothesis-testing of received signal parameters in the UWB channel. By exploiting the statistics of ToA, RSS, and the Root Mean Squared Delay Spread (RDS) they accurately distinguished between LoS/NLoS signals. Identified NLoS range estimates were then used to define a feasible region for potential solutions effectively mitigating range bias. Their proposed solution, employing a LS estimator with both LoS and NLoS ranges, achieved a localization accuracy of 0.5 m, outperforming other approaches that relied solely on LoS ranges or combined and biased LoS/NLoS ranges [88].

García *et al.* presented a robust UWB positioning solution designed for complex indoor environments with prevalent NLoS conditions. The system applied an NLoS detection and mitigation algorithm based on the skewness of the estimated CIR and further refining the NLoS errors with an Extended Kalman Filter (EKF). Upon NLoS detection, a fixed value was added to the measurement noise covariance value, effectively increasing the confidence towards LoS measurements. Real-life tests demonstrated that their solution improved positioning RMSE by approximately 67% compared to a solution without NLoS mitigation [89].

Machine Learning techniques have also been widely applied for LoS/NLoS identification and mitigation schemes, as it is a powerful tool to model dependencies between CIR parameters and respective response variables. These methods in terms of UWB positioning are described in more detail in Section 3.3.2.

CIR offers a direct and accurate representation of actual UWB channel conditions, making it a valuable tool for identifying NLoS scenarios. Additionally, CIR can be extracted for each tag-anchor pair during individual ranging sequences, eliminating the need for previous CIR data [11]. Furthermore, ML algorithms are well-suited to create models based on various statistical values derived from large real-world datasets.

However, it can be argued that CIR-based approaches have also certain limitations. Firstly, as pointed out by Barral *et al.*, gathering CIR samples entails a significant amount of data, causing a latency of approximately one second just for measurement transfer [19]. Considering a UWB positioning system with high positioning update rate and a significant network of anchors, using CIR becomes impractical and would be more applicable in small-scale and low update rate setups. Using an ML model in conjunction with real-time data extracted from CIR, increases data delays and computational complexity even further [39]. Moreover, ML models can be difficult to generalize as they depend on site-specific training data. For example, a model trained on a dataset collected from a residential area might not be suitable for a cluttered industrial environment as NLoS conditions appear dynamically with constantly changing obstructions between the anchors and the tag [86]. Finally, creating an ML model is a cumbersome process as it requires significant amounts of measurement data gathered from various environments with additional effort for training, validation, and testing.

3.1.3 Application of signal parameters

Various authors have proposed NLoS detection methods that do not rely on CIR information. The motivation for exploring alternative approaches stems from the same challenges outlined in the previous section, primarily the significant time delay associated with processing CIR information.

Barral *et al.* investigated the use of UWB Received Signal Strength (RSS) and raw ranging data to identify NLoS scenarios. Different statistical sets of RSS and ranging features were incorporated in a Support Vector Machine (SVM) classifier model, which was then experimentally evaluated in real-life LoS and NLoS conditions. The authors achieved an over 90% LoS/NLoS classification rate only using mean values of both RSS and range measurements [19]. In another work they proposed application and comparison of different ML models trained on RSS and ranging features, along with different filtering methods for final position estimates. It was reported that an iterative EKF with k-Nearest Neighbors (k-NN) based NLoS detection scheme, achieved a Mean Absolute Error (MAE) of approximately 0.084 m. However, it was concluded that the proposed NLoS identification and mitigation method may suffer from generalizing issues, as it is strongly dependent on the specific environment in which the models were trained [90].

Guo *et al.* combined Pedestrian Dead Reckoning (PDR) with gait detection and UWB positioning within a Kalman Filter (KF) algorithm. The authors acknowledged the importance of identifying NLoS conditions and proposed estimating their probability using the RSS difference between the first path signal and reflected signals. The proposed filtering scheme achieved an average error less than 0.16 m [91]. Similarly, Kim *et al.* investigated UWB NLoS identification by comparing the Received Signal Power Level (RSL) with First-Path Signal Power Level (FSL). An Advanced Channel Diagnostics Algorithm (ACDA) was proposed to differentiate between different power levels and determine NLoS or LoS scenarios. The ACDA achieved an over 99% success rate in NLoS identification for four different NLoS paths [92].

Wu *et al.* analyzed the principle and characteristics of NLoS error and proposed a NLoS mitigation method derived from the signal propagation path loss model. The authors provided calculation expressions for the NLoS error estimation while considering the effects of antenna direction. The method was implemented in low-complexity Direct-Path (DP) and Maximum Confidence Path (MCP) detection algorithms, suggesting potential application and practical design. The processed ranging results demonstrated that the proposed method can significantly reduce range estimation errors by several decimeters, which in turn enhances the positioning accuracy of UWB-based sensor networks [93].

While UWB signal parameters can be effectively applied in NLoS identification and mitigation schemes, there are also some drawbacks to be considered. Firstly, models based on RSS data can be difficult to generalize, as training data is site-specific. A model that performs well in one location may not work in another area [90]. Additionally, the received signal may be affected by dynamically changing obstructions which may cause signal attenuation or interference. It would be difficult to model respective path loss or signal strength changes for each obstruction. Moreover, considering filtering schemes, signal interference may cause an accurately estimated position to be weighted incorrectly, thus affecting the integrity of the filtered end coordinate.

3.2 Integrity estimation for GNSS and multi-sensor localization

This section discusses methods used in the state of the art for estimating positioning integrity in a GNSS-UWB multi-sensor fusion algorithm. Depending on the integration strategy, the joint positioning system can be loosely-coupled, tightly-coupled or ultra-tightly coupled. Since the latter involves baseband signal processing, then this approach is typically inaccessible for most Commercial Off-The-Shelf (COTS) products [94]. In the loose-coupling scenario, GNSS and UWB sensors operate independently and estimated positions are fused at a higher level, usually with a Kalman Filter scheme. Loose coupling can also be considered due to limitations in the GNSS receiver or design of the multi-sensor system. Some COTS products restrict access to raw data, with serial data as the sole source of positioning information. In contrast, a tightly-coupled solution integrates raw measurements (e.g., GNSS/UWB pseudoranges, GNSS carrier phase measurements, Doppler observables) resulting in a more efficient data usage and a higher control of noise terms [52], [94], [95].

A loosely coupled GNSS can provide information in the form of National Marine Electronics Association (NMEA) messages, usually transmitted over serial communication port (RS-232) [52], [96]. These messages contain geographical fix information, heading, date and time, velocity and track over ground, pseudorange noise statistics, constellation geometry information etc. Bao *et al.* utilized Multi-Task learning (MT-e&R) for deep feature extraction from the NMEA protocol to predict and calibrate the GNSS positioning error and estimate the GNSS measurement noise covariance. Furthermore, it was stated that

NMEA features can reasonably characterize complex positioning environments. On the other hand, Dyukov *et al.* relied on NMEA data to investigate the accuracy of GNSS velocity measurements in challenging conditions. The aim was to develop a quality indicator to filter out potentially unreliable GNSS speed records [97].

Tightly-coupled sensor fusion, leveraging raw measurement data and high customizability, offers high flexibility. By fusing complementary sensor measurements (e.g., UWB and GNSS positioning information) it can provide enhanced positioning performance.

Jiang *et al.* proposed a tightly-coupled integration of UWB, GNSS and Inertial Navigation System (INS) sensors for seamless indoor and outdoor positioning. They incorporated two distinct positioning workflows: one for indoor environments and another for outdoor-indoor transition areas. In the indoor mode, an INS complements an integrated UWB to mitigate positioning errors. In the second mode, GNSS is added as an additional positioning source alongside UWB. Both workflows utilize an Extended Kalman Filter (EKF) for measurement prediction and correction achieving seamless and accurate outdoor-indoor positioning with Distance Root Mean Square (DRMS) and RMSE of 5.25 cm and 10.18 cm, respectively. Recognizing the crucial role of measurement noise in determining positioning uncertainty, two different noise covariance matrices are used. The UWB noise covariance matrix is modeled as a constant diagonal matrix, considering the error of the original UWB measurement values (e.g., multipath variation and ranging noise). Conversely, GNSS noise covariance is based on the preset variance of the GNSS carrier phase measurements which is also considered to be constant [48].

Similarly, Song *et al.* proposed a tightly-coupled fusion of UWB, GNSS and Inertial Measurement Unit (IMU) data for indoor and outdoor positioning. To enhance positioning continuity and reliability, they introduced novel adaptive weighting factors. The system was tested in real-life environments consisting of complex indoor and outdoor areas. The authors used an EKF for positioning prediction and update, dynamically adjusting weights for UWB and GNSS sensors. These adaptive weights automatically adjust the covariance matrices of the measurement noise based on fluctuations in GNSS and UWB signal levels [98]. Following a similar approach, Li *et al.* assigned weights to UWB measurements based on their RSS indicator levels and to GNSS measurements based on the standard deviations of their pseudorange and phase measurements. These weighted measurements were then integrated in a tightly coupled PPP/INS/UWB framework for their low-cost unmanned ground vehicle. The authors reported seamless outdoor-indoor positioning with over 90% sensor availability, and approximately 20 cm 3D MAE and 30 cm 3D RMSE [51].

An alternative approach to estimating measurement uncertainty was used by Wang *et al.* in their tightly-coupled integration of multi-GNSS RTK, INS, UWB and map data. They constructed an Adjustment Factor (AF) based on the Robust Estimation for Correlated Observations (RECO) scheme [99]. The elements in the observation noise matrix (utilized in the EKF update step), are derived from the product of the AF and observation noise variance. Essentially, when the accuracy of observations is stable, the AF is 1. Conversely, when the observation values are severely abnormal, the AF is infinite and observation does not affect state estimates. In the intermediate case, the impact of observation anomalies is mitigated. Their proposed solution was tested in a real-life indoor-outdoor scenario, achieving an RMSE of approximately 0.24 m, a Mean Location Error (MLE) of 0.16 m and a maximum error of 1.14 m [54].

Having an accurate estimate of positioning uncertainty in multi-sensor fusion schemes is crucial for robust and seamless positioning in indoor, outdoor, and transition areas. As the accuracy and precision of the estimated position dynamically changes, using constant measurement noise values in a filtering scheme may significantly impact positioning per-

formance. For instance, in an indoor area, a severely inaccurate GNSS coordinate with an assigned constant weight, can adversely affect the final estimated position, even in the presence of an accurate UWB measurement. On the other hand, adaptive measurement uncertainty is more suitable for dynamic positioning as it considers preset parameters to assign weights to the estimated position of each sensor at each measurement update. Using weights based on received signal levels as measurement uncertainty estimate, provides a convenient way to bias towards a sensor with stronger signal [98]. However, as signals are susceptible to interference (e.g., constructive or destructive), signal strength can lead to inaccurate assessment of position accuracy. For example, an accurately estimated coordinate may have a weak signal level and vice versa. Alternatively, assigning sensors their respective uncertainty weights can also be done by using a preset robust estimator (e.g., RECO) [54]. It is a statistical technique to mitigate outliers and apply reduction factors to the observation weight matrix. For example, the weight of an outlying observation is reduced and vice versa. However, RECO suffers from optimal parameter tuning as the reduction factor of the weight elements should be determined beforehand [99]. Therefore, a poorly chosen parameter can lead to generalization and overfitting issues.

3.2.1 Geofencing

Geofencing is a technique, where an estimated position, provided by a sensor, is compared to a previously established map and its predefined boundaries. It quantifies, informs, or influences resource movements or positions based on real-time coordinates [100]. This method is particularly useful in areas with significant positioning errors, as it can effectively discard estimated positions that fall outside the specified area or track. Additionally, this method is computationally simple since the algorithm only compares sets of coordinates with a decision boundary. The geofence perimeter can be defined in either a local or global frame of reference, depending on the specific multi-sensor solution.

Considering a scenario, where an object traverses indoor and outdoor areas, respective positioning data can be used to provide coordinates based on the object's location. If the object is estimated to be indoors, only UWB is used while GNSS is considered only for outdoor areas. A similar approach was used by Di Pietra *et al.* in his work about pedestrian navigation using a loosely coupled integration of GNSS, UWB and INS [43]. Their data fusion algorithm leverages geofencing as a trigger to switch between indoor and outdoor environments. Another application of geofencing would be to determine NLoS conditions for UWB systems. As an object moves indoors, its estimated position can be continuously monitored relative to the servicing UWB anchors. If the object is estimated to be behind a geofenced boundary (e.g., a wall), the ranging information from that anchor may be excluded from subsequent position calculations. The proposed solution provides an overall 2D and 3D accuracy of 30 cm and 45 cm, respectively. Wang *et al.* developed a similar approach in their tightly coupled GNSS, UWB, and INS solution for autonomous vehicles [54]. Their system consistently monitors UWB signal occlusions by comparing the estimated position with a predefined map and adjusts the weight of an observation or discards the measurement entirely.

While geofencing offers several advantages in multi-sensor positioning systems, it also presents certain limitations. Firstly, geofencing relies on area-specific maps and requires tailored implementation. Establishing coordinate-based boundaries requires detailed knowledge of the area and potential movement paths, making implementation cumbersome. Secondly, since geofencing does not account for positioning uncertainty, erroneous estimates may still persist within the geofenced area. Additionally, defining a strict decision

boundary between operational areas of respective positioning sensors can potentially degrade the performance of a multi-sensor solution. For example, an object transitioning from an indoor UWB network to an outdoor area with GNSS reception may have accurate UWB position estimates discarded prematurely leading to a suboptimal solution. Furthermore, an accurate UWB solution may be replaced with a poor GNSS position estimate. In contrast to geofencing, the research presented in this thesis is more focused on a more generalized and comprehensive solution, one that can be applied without the need for prior knowledge of the positioning environment.

3.2.2 Positioning data distribution

An alternative method for estimating UWB positioning uncertainty was utilized by Zhang *et al.* in [40]. The author applied Circular Error Probable (CEP) as an uncertainty metric for UWB and also combined GPS, UWB and the Magnetic, Angular Rate, and Gravity (MARG) sensor to improve indoor/outdoor positioning and mitigate sensor dropouts. It was reported that the MARG sensor improved the overall positioning accuracy from 8.9 m to 3.2 m (approximately 64%).

It can be argued that CEP is a metric primarily used to evaluate stationary positions, as it is defined as the radius of a circle centered at the true position, containing 50% of the actual measurements [101]. Therefore, it would be difficult to use this measure for moving objects. Additionally, as mentioned by Lv *et al.*, using CEP requires prior knowledge of error distribution within the deployment area. Moreover, as the positioning sensors are switched based on GNSS DoP and UWB CEP threshold values, this approach may not guarantee a stable trajectory in the transition area [45].

3.2.3 Dilution of precision

In the literature it can be seen that dilution of precision is often used as an indicator for positioning uncertainty. It quantifies how much position error, that results from measurement errors, depends on the receiver/satellite (tag/anchor in case of UWB) relative geometry [21], [72]. For example, a sensor fusion solution exploiting two positioning sensors would consider a position estimate with a larger DoP as more imprecise compared to the other sensor [102]. Yao *et al.* incorporated UWB HDoP and signal Carrier-to-Noise Ratio (CNR) of GNSS to a tightly coupled INS positioning solution with federal filtering, achieving a decimeter-level 2D positioning error [103]. Lv *et al.* relied on HDoP to assess the changing accuracy of GNSS RTK and UWB systems in his work concerning seamless indoor and outdoor positioning of vehicles. Consequently, a unified positioning accuracy index was developed based on HDoP from both systems and implemented in the sensor fusion strategy, achieving an approximately 8 cm accuracy [45]. Zhu *et al.* used HDoP-based weights to determine the uncertainty of GNSS and UWB positioning in an integrated strategy with Dead Reckoning (DR), and Visual Map Matching (VMM) resulting in a sub-meter horizontal positioning accuracy [104]. Sun *et al.* utilized DoP as a measure of positioning accuracy in a semi-tightly coupled robust model for GNSS, UWB and INS sensors recognizing that a lower DoP factor indicates a lower amplification of ranging errors and a higher system fault tolerance. The authors reported positioning accuracy in east, north, up directions as 0.42 m, 0.55 m, and 3.2 m, respectively [105].

As shown by various authors, geometrical uncertainty can effectively be incorporated into multi-sensor positioning solutions. However, considering an object's position relative to the servicing nodes, it may not fully capture the complexities of positioning errors. For instance, UWB positioning is affected by the number of servicing anchors, their proximity to the tag, signal degradation caused by NLoS and various materials [7], [14]. Fur-

thermore, UWB positioning can be particularly challenging in narrow or confined spaces, where electromagnetic wave propagation will have various reflections, refractions, and keyhole effects, resulting in channel deterioration and an increase in ranging errors [72]. On the other hand, the GNSS positioning is subject to additional error sources. Firstly, there are errors caused by signal propagation paths with potential delays caused by the Earth's atmosphere and multipath effects. Secondly, in the ground segment there may be receiver instrument errors, which can influence positioning precision. Lastly, there may be errors related to the satellite segment: the satellite ephemeris, clock drift, and geometry of constellations [21], [106]. As can be seen, the errors of the estimated position are not limited to geometrical uncertainty and should ideally be considered in multi-sensor schemes.

3.3 ML application in related works

Machine learning is a powerful tool that has been extensively used in different positioning schemes to improve their robustness and performance. Compared to traditional statistical methods, ML techniques enables identification of complex dependencies in data that may not be apparent through exploratory data analysis. While the goal is not to derive an explicit mathematical formula for the data distribution, it can effectively be used to train algorithms to learn the relation between input features and their response variables [57]. Considering the context of this thesis, this section gives an overview of main ML approaches to UWB, GNSS and seamless positioning.

3.3.1 Machine learning in GNSS positioning

In order to improve the performance of GNSS positioning in less than optimal signal environments, ML-based GNSS models have been investigated as early as 1990s. GNSS sensors can provide nearly limitless quantity of data that can be used in a variety of ML models depending on context of application. Siemuri *et al.* has done an extensive systematic literature review on the topic of ML techniques for GNSS use cases [57]. These scenarios include: GNSS signal acquisition, signal detection and classification, Earth observation and monitoring, GNSS navigation and precise positioning, GNSS-denied environments and indoor navigation, GNSS anomaly detection and atmospheric effects, GNSS security, GNSS/INS integration, satellite selection, and Low-Earth-Orbit (LEO) satellite orbit determination and positioning. While most of these topics are out of the context for the current thesis, this survey provided insight into some of the more closely related works.

Considering GNSS navigation and precision, Kim and Bae used Long Short-Term Memory (LSTM) method to improve the accuracy of GNSS-RTK positioning. The authors applied position error from the absolute position, wheel speed sensor data, and yaw information as training data for the ML model [107]. Kuratomi implemented Decision Tree (DT) and Support Vector Machine (SVM) to estimate positioning error using features based on azimuth, elevation, constellation type, and CNR. It was reported that the SVM model reduced the RMSE by 31% compared to the DT model [108]. Zhang *et al.* investigated the prediction of urban GNSS satellite visibility and pseudorange error based on deep learning networks with LSTM. The ML model incorporated features such as satellite elevation and azimuth angle, CNR, individual pseudorange residual, and the root-sum-squares of pseudorange residuals from all available satellites. By testing the performance of the model in an urban area, the satellite visibility was predicted with 80.1% accuracy and pseudorange measurement error with an average difference of 4.9 m from reference [109]. Lyu and Gao proposed a weighting scheme for improving kinematic GNSS positioning in urban environments using a novel multi-feature SVM approach. It is based on the identification

of the most important features in GNSS data in urban environments and the intelligent classification of LoS and NLoS signals. The input features are calculated from the GNSS observations and are based on pseudorange, phase, and Doppler prediction data. The resulting scheme outperformed the traditional CNR based weight model by 65.4% and 85% in the horizontal and up direction. The weighting scheme was also capable of overcoming position error spikes at overcrossing and short tunnels [110].

Scenario recognition in GNSS-denied environments is also considered a major topic in seamless indoor and outdoor positioning. The goal is to use ML to identify current positioning environment and then apply appropriate strategies for a more accurate and reliable positioning [57]. For example, Xia *et al.* investigated scenario recognition with multi-constellation GNSS on a smartphone by exploiting the Recurrent Neural Network (RNN). The model was trained on position-independent features, which included the number of visible satellites and various statistical measures for satellite CNR. It was reported that the model could recognize isolated and transition areas with high overall accuracy of 98.65% [111]. Liu *et al.* developed a NLoS and multipath detection network using deep learning approach. The model was trained on datasets generated by a GNSS software receiver using an intermediate frequency signal from an indoor pseudo-satellite system. This model was then compared to SVM-based classification method, showing an improvement of up to 45% in overall classification accuracy [112]. Klus *et al.* proposed a Neural Network (NN) solution to boost positioning accuracy in urban areas by fusing beamformed RSS measurements from user equipment with GNSS positioning data. The author investigated a RSS fingerprinting model and a positioning fusion model that combines sequential outputs of the first model with available GNSS measurements. The first model demonstrated meter-level accuracy and the second model sub-meter accuracy in uncertainty-free scenario [113].

As can be seen from presented literature, various ML techniques can be applied for various different tasks. In the context of improving GNSS accuracy, there is an obvious tendency in using low-level GNSS measurement data such as pseudoranges, carrier phase measurements, Doppler information, satellite azimuth/elevation angles and more as input features for ML training. However, it can be argued that such data is not always accessible from COTS devices because of firmware or manufacturer limitations. Moreover, processing large amounts of data can be computationally intensive in training the model and applying in a real-time system. As Siemuri *et al.* identified, approximately 47% of the GNSS-related ML algorithms use some form of NN approach, making it the most popular method for developing an ML model [114]. However, it is also common knowledge that NN has a significant number of parameters (e.g., weights or biases) to be learned during the training, leading to optimization problems and being altogether computationally demanding. The latter is also reflected in a significant processing time delay as demonstrated by Xia *et al.* [111]. In the research it was identified how an RNN model, trained for scenario recognition, entailed significant time delays in identification and scenario switching decision, which could take up to 3 seconds. On the other hand, NLoS and multipath recognition algorithms, besides being computationally intensive, require complete knowledge of the geometry and physical characteristics of the reflecting surfaces [115].

An ML-based approach for GNSS uncertainty estimation should consider the aforementioned limitations. An ML model should be computationally fast to be applied in a high update-rate positioning system. Also, the model should contain features using high-level data that is straightforward to implement and obtainable from COTS devices.

3.3.2 Machine learning in UWB positioning

In recent years, machine learning has been applied extensively to improve UWB-based positioning. By incorporating low-level data such as channel state information, channel statistics, or ranging parameters, researchers have leveraged ML techniques to find patterns and dependencies between potential features and use this information for error detection and mitigation schemes in UWB positioning. The following paragraph describes state-of-the-art solutions combining both ML and UWB technology.

Liu *et al.* developed a framework to utilize cost-effective and robust UWB-based positioning, with extreme gradient boosting-based ML technique and coordinate filtering for precise distance measurements. The model incorporates various features extracted from UWB ranging information including estimated and corrected distances, impulse response parameters, and RSSI indicator. Feature selection was performed using correlation analysis, identifying low level of collinearity among the features. Consequently, all features and corresponding real distances were used in model training. The authors also compared the proposed model with XGBoost, Random Forest, and SVM ML models in terms of their performance and processing latency. It was found that the proposed model outperformed the other models in all regression performance metrics. While the proposed model and XGBoost exhibited comparable processing time, both significantly outperformed Random Forest and SVM in this regard. Real-world testing in building condition monitoring validated the framework's ability to measure distances with millimeter-level accuracy [116].

Bregar and Mohorčič presented two methods to mitigate positioning errors in indoor NLoS conditions using raw CIR from UWB sensors and Convolutional Neural Network (CNN). The first method exploited a CNN-based classifier to identify NLoS measurements and exclude unreliable ranging nodes. The second method utilized a CNN-based regression model to predict ranging errors and incorporate them in a weighted least squares estimation process. The latter approach, combining WLS and ranging error mitigation, demonstrated the best localization performance. Both models were trained on a dataset collected in both LoS and NLoS conditions. The authors also investigated the computational efficiency of the CNN-based NLoS classifier, considering different batch sizes on various hardware configurations. While acknowledging the increased computational demands due to high input dimensionality and CNN complexity, the authors suggest that network topology optimization and reduced CIR sample sizes can make these methods viable alternatives to traditional approaches [117].

Similarly, Niu *et al.* proposed a deep learning-based approach to mitigate UWB positioning errors in NLoS conditions. The method leverages CIR information to predict ranging errors and correct estimated distances before applying them to a least-squares-based 2D position estimation. The UWB tag collects CIR data from all anchors and transmits it to a host computer running the ML model. The model's output, representing corrected distance estimates, is then used to calculate the 2D position. The UWB localization system was tested in a real-life complex environment showing correctly mitigated ranging errors, thus resulting in a more accurate localization [118].

Fan *et al.* proposed an unsupervised ML approach using Expectation Maximization (EM) for Gaussian mixture models to discriminate between LoS and NLoS conditions. This algorithm assigns LoS/NLoS probability to each UWB signal through soft clustering of various CIR statistics [119]. Kim *et al.* utilized a LSTM model to classify UWB channel conditions based on the magnitude of impulse response. Additionally, the model estimates were integrated into the EKF to mitigate positioning degradation. Comparative analysis with WLS, non-augmented EKF and LS methods demonstrated the superior performance of the LSTM-EKF approach [120].

Wymeersch *et al.* proposed an approach for NLoS identification directly based on measured UWB waveforms and their statistics. The authors applied two classes of non-parametric regressors to estimate the ranging error and directly mitigate the bias incurred in both LoS and NLoS conditions. The first technique exploited the SVM regression to find a hyperplane approximating the ranging error as a function of UWB waveform statistics. The second method employed a Gaussian process to determine the posterior distribution of the training error [121].

Several other authors have incorporated Channel Statistics (CS) information with various ML schemes such as LSTM [120], Multiple Input Learning (MIL)-NN [122], CNN [123], deep learning [124], sparse pseudo-input Gaussian process [125], SVM [126], [127], [128], Decision Tree [129], Random Forest [130], LS-SVM [131] etc.

On the other hand, authors who have provided non-CS-based approaches, have also leveraged ML techniques. Silva and Hancke investigated NLoS identification using ranging residuals, considering the sum of squares of distance residuals as a suitable feature for a naive Bayes NLoS classifier [86]. Barral *et al.* applied SVM technique to train a model based on UWB RSS and ranging information for NLoS classification [19], [90]. Chang *et al.* bypassed low-level UWB information entirely and proposed optimizing the classical multilateration algorithm using different regression techniques, such as linear, high-order polynomial, Lasso, and Ridge regression [132].

As discussed in paragraph 3.1.2, CIR provides accurate information about possible signal obstructions and potential NLoS scenarios, making it the most prevalent source of information and features for various ML techniques, especially NN and its variations. While the works mentioned above have significantly improved the accuracy and precision for UWB-based positioning (particularly for static scenarios [116]), certain limitations should be considered. As noted in 3.1.2 and by other authors, extracting and processing CIR information is time-consuming and may not be suitable for systems requiring high position update rate (e.g., dynamic positioning) [19], [90]. Additionally, channel statistics must be collected for different types of environments, as a training dataset collected in an office building may not be suitable in describing an industrial area [78], [86]. However, in the context of the current paragraph, it should also be considered that complex ML algorithms may also pose limitations for real-time systems. As noted by Bregar and Mohorčič, their CNN model (with CIR-based input data) implies higher computing demands compared to classical approaches. On the other hand, SVM provides a more precise and robust NLoS identification by establishing a decision boundary according to the input features with support vectors. However, the manually selected vector feature, generated from the UWB signal propagation path loss model, might be inadequate to meet the identification requirement in various positioning scenarios [122]. Therefore, choosing an appropriate ML technique depends on the task and its limitations. Considering the importance of computational latency and the models application in a high update rate system, using a NN or SVM-based approach may be unfeasible. Moreover, as stated by Nessa *et al.* in a survey regarding application of ML techniques in indoor positioning, SVM-based methods are time-consuming and require a significant amount of memory when the number of support vectors become large. On the other hand, the decision tree-based indoor positioning provides better performance in improving accuracy when compared to NN or k-NN [9], [133].

3.4 Research gaps

According to the literature presented in previous sections, the following research gaps were identified.

- Compared to CS-based methods for enhancing UWB position estimation accuracy, the utilization of ranging residuals remains an under-explored area. Its primary advantage lies in its low latency, as residuals (and their features) are directly calculated from raw ranging data. In contrast, CS information necessitates separate extraction, incurring significant time delays, rendering it unfeasible for high-update-rate positioning systems. While ranging residuals and their features have been applied in NLoS identification and mitigation schemes, their application to uncertainty estimation (i.e., coordinate accuracy) has been overlooked to the best of author's knowledge. Additionally, as ranging residuals are essentially a collection of distances from the estimated position, various features can be considered to estimate the overall position quality (**RQ1**).
- An ensemble of features, describing ranging residuals and the true position offset, can be employed with different ML techniques to identify underlying dependencies. Furthermore, residual features can be combined with other measures, such as dilution of precision, the distance between LS and NLS estimates or the number of NLS iterations etc., which indirectly indicate positioning integrity (**RQ1**).
- It is crucial to consider the computational efficiency of different ML techniques, as the model must be sufficiently fast when used in a high-update-rate positioning system (**RQ1**).
- By exploiting ML to estimate position uncertainty, this information can be integrated into various filtering schemes as an indicator of measurement uncertainty. This approach diverges from traditional methods, that prioritize NLoS identification and mitigation (**RQ2**).
- GNSS uncertainty estimation has been extensively researched, with numerous authors proposing error mitigation schemes for both standalone and sensor fusion solutions. Many of these approaches leverage raw GNSS data in a tightly-coupled manner. However, such low-level data access is limited to devices that provide the necessary information. Loosely coupled GNSS solutions, on the other hand, utilize already processed high-level data in the form of NMEA messages. Such data is usually more accessible and can be used with most COTS devices. Similar to the application of UWB-based residual features, loosely coupled GNSS data offers an ensemble of features that can be used to characterize positioning uncertainty (**RQ3**).
- While authors proposing loosely coupled GNSS solutions often rely on a single feature for uncertainty estimation, an ML-based approach leveraging an ensemble of features can potentially yield more accurate uncertainty estimates (**RQ3**).
- As uncertainty estimation is a critical parameter in sensor fusion schemes, the application of ML models to this task represents a novel approach. By integrating uncertainty estimates from both UWB and GNSS models, improved performance may be achieved in seamless indoor-outdoor sensor fusion systems (**RQ4**).

These research gaps motivate further investigation in improving UWB and multi-sensor positioning, as presented in subsequent chapters of this thesis.

4 UWB positioning accuracy classification using ML

The following section describes the first contribution of this thesis, which is an approach for UWB coordinate accuracy classification using ML. In contrast to usual methods, such as using CS for positioning integrity estimation, this section investigates whether positioning errors could be estimated using a pre-trained ML model. By incorporating an ensemble of features, mainly based on ranging residuals, the ML model was trained and tested on real-life UWB measurements to predict UWB coordinate error.

This section is based on Publication I:

- M. Tommingas, S. Ulp, M. M. Alam, I. Mürsepp, and T. Laadung, “Estimating UWB Positioning Integrity Based on Ranging Residuals,” in *2023 24th International Conference on Applied Electromagnetics and Communications (ICECOM)*, pp. 1–5, IEEE, 2023

4.1 Background and motivation

Coordinate accuracy classification using ML was an attempt to investigate whether a pre-trained ML model can use information from ranging residuals to predict the coordinate integrity. It was also motivated by the fact that application of channel statistics for estimating ranging errors is unfeasible when applied in a high update rate positioning system (Section 3.1.2). On the other hand, by exploiting information based on ranging residuals (which are calculated from ranging information and estimated position), it is assumed that coordinate integrity can be estimated and processed with smaller latency compared to CS-based methods. However, the latter hypothesis was investigated in Publication II.

Features that could describe whether residuals could be used in estimating UWB-based positioning integrity have not been thoroughly researched. For example, some features used by Silva and Hancke for LoS/NLoS detection, were also used in the proposed ML model: SSQ, mean and standard deviation of distance residuals [86]. However, the proposed model incorporates also several additional statistical measures related to position estimation and geometrical uncertainty. Altogether 28 features, were divided into 5 categories:

- **Lengthened and shortened residuals.** As described in Section 2.3, a residual is the difference between a distance of an estimated coordinate and measured range from an anchor. Depending on the NLS solution, which considers all available ranges, the estimated position may appear closer or further away relative to the anchor, thus resulting in a lengthened or shortened residual. Therefore, following features could be established: number of lengthened/shortened residuals, sum of lengthened/shortened residuals, average of lengthened/shortened residuals, root mean square error (RMSE) of lengthened/shortened residuals.
- **Residual statistics:** Following statistical features were included: variance, standard deviation, SSQ, sum of absolute values, mean, absolute mean and RMSE of residuals.
- **Number of residuals in range.** Small residuals indicate proximity to the NLS solution, whereas large residuals imply erroneous measurements. The idea was to count the number of residuals in a preset range, as it can indicate whether the NLS algorithm uses accurate measurements as its input. Following ranges were chosen:

0...0.1 m, 0.1...0.2 m, 0.2...0.4 m, 0.4...0.8 m, 0.8...1.6 m, 1.6...3.2 m, 3.2...6.4 m, 6.4...12.8 m, 12.8...25.6 m, 25.6... ∞ m.

- **LS and NLS metrics.** These values are associated with estimated position calculations described in Section 2.2. Chosen parameters include: Euclidean distance between LS and NLS solutions, number of NLS iterations to convergence and NLS convergence tolerance. The latter is based on relative offset convergence criterion. This assures that the current parameter vector is less than 0.001% of the radius of the confidence region from the least squares solution [134].
- **Geometrical integrity of positioning.** In context of UWB positioning, DoP indicates geometric location distribution [72]. Using estimated coordinates of the tag, DoP parameter indirectly shows the level of geometrical uncertainty in an area relative to the anchors. In this ML model, Position Dilution of Precision (PDOP) was calculated with x , y and z coordinates.

4.2 ML model training

The ML model was trained on real-life measurement data gathered at an industrial site of Krah Pipes OÜ, which manufactures thermoplastic pipes as shown in Fig. 13. Regarding UWB-based ranging, the site presents a complex environment with constantly moving obstacles, which produce NLoS and multipath effects for RF signals. An Eliko RTLS UWB test system was set up inside the manufacturing facility by placing UWB anchors at preset locations. Tag's 3D true coordinates were measured in a local frame of reference with the Leica DISTO S910 measurement tool and assigned for 8 UWB anchors as well as 30 different locations around the facility. At each location approximately 300 data points were gathered. **It should be noted that the measurements were not gathered by the author but rather the collected raw ranging data was used by the author to calculate the features, position estimates, and response values¹.**

As mentioned in Section 2.5, the ML model was created using supervised learning, which means for each set of features and their values, a response value was assigned. The latter depends on the premeasured true coordinate and is set as the Euclidean distance between the true and estimated coordinate i.e., coordinate error. The task was to perform accuracy classification with preset accuracy classes:

Class 1: Distance between 0...0.2 m;

Class 2: Distance between 0.2...0.4 m;

Class 3: Distance between 0.4...0.8 m;

Class 4: Distance between 0.8... ∞ m.

These categories were chosen based on UWB performance studies [70], [71]. Class 1 presents positioning accuracy up to 0.2 m, which is also considered an approximate accuracy level for UWB-based positioning given in the literature. Other accuracy classes were set as a double value from the last step.

After dataset cleaning and shuffling, 80% of the data was used for training, while 20% was used for testing purposes. The classification model was trained using Extreme Gradient Boosting (XGBoost) in RStudio environment with an imported *xgboost* library [136], [137]. Initially, a sequence of boosted trees was constructed with *xgb.train* function to

¹The data was collected by Taavi Laadung and Sander Ulp [135].



Figure 13: Industrial site at Krah Pipes OÜ, which manufactures thermoplastic pipes in a complex industrial environment. Figure from Publication I.

determine the number of trees that produce the smallest prediction RMSE. The resulting 93 boosted trees were used in the final model training using *xgboost* function.

As stated in Section 2.5, XGBoost can also be used to extract features that contribute the most information in making the prediction. As can be seen in Fig. 14, the most important features out of the set of 28 are related to residuals as these features provide the biggest informational gain or importance. Lesser contribution is provided by PDOP, number of residuals in 0...0.1 m range, LS/NLS solution distance, number of NLS optimization iterations and mean of residuals. Significantly large residuals can indicate that the estimated position is further away from the individual measured distance from the anchor and the estimated position has a potentially large error. Similarly, small residuals indicate little change between the individual measured distance from an anchor and the distance to the estimated position.

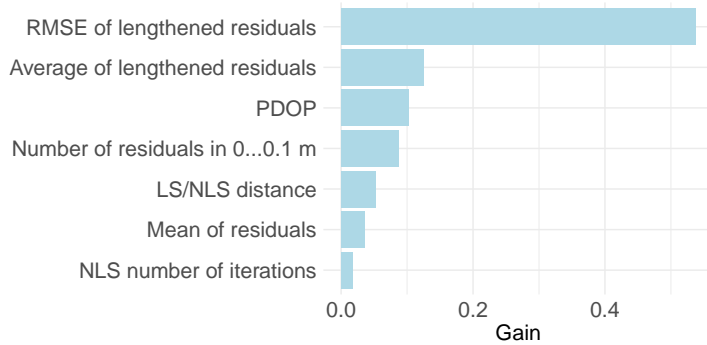


Figure 14: Top 7 features that provide most information as proposed by XGBoost algorithm. It can be seen that lengthened residuals are the most important features in describing estimated position integrity. Figure from Publication I.

4.3 ML model testing

Model testing was done on 20% of the mixed dataset that was not used in the training (total of 1748 samples). The goal was to evaluate model's classification performance by comparing true accuracy classes against predicted classes, which can be seen in Table 3. Furthermore, true positive rate (sensitivity) and true negative rate (specificity) were calculated for each class indicating the model's ability to correctly identify the estimated position within or outside their assigned accuracy class, respectively [138]. When comparing the references and predictions, it can be seen that all of the classes were mostly predicted correctly. While Class 2 and 3 had the worst true positive and true negative rates, these were still predicted correctly in over 62% of samples. These classes were also the most difficult to predict because measurement points (especially Class 2), were too close to Class 1 points in order to be distinguished with a high success rate. Overall, prediction of distinct classes was performed with a significant accuracy of 84%.

Table 3: Confusion matrix and prediction accuracy statistics. Prediction of different classes was done with varying accuracy. For example, if there were 76 points that belonged to Class 2, then according to the model these were labeled as Class 1. With a 89% Sensitivity (true positive rate) the model could identify most of the points belonging to Class 1. Class 2 was harder to predict with a Sensitivity of ca. 62%, since Class 2 points accuracy is very close to Class 1. However, if the measurement point did not belong to Class 1 or Class 2 then the prediction was made with a high Specificity (true negative rate) of ca. 87% and 91%, respectively. Table from Publication I.

		Reference Class			
		1	2	3	4
Predicted Class	1	1024	76	1	0
	2	109	240	14	0
	3	8	45	54	12
	4	0	5	17	143

Overall Accuracy: 0.8358 (ca. 84%)

Statistics by Class:

	Class 1	Class 2	Class 3	Class 4
Sensitivity	0.8975	0.6557	0.6279	0.9225
Specificity	0.8731	0.911	0.9608	0.9861

4.4 Discussion

In summary, the proposed XGBoost model performed UWB-based positioning accuracy classification with an overall high success percentage of ca. 84%, considering the narrow accuracy range of Class 1. It was shown that the model could also predict other accuracy classes with significant accuracy using features mainly based on UWB ranging residuals. Prediction sensitivity could possibly further be improved with a broader Class 1 range (e.g., 0...0.3 m). The ML algorithm distinctly separated the lengthening of residuals as one of the top features, that could describe the integrity of positioning. However, there are some additional actions to be considered when using an ML model to predict coordinate accuracy.

Firstly, while it was shown that XGBoost algorithm can output the most significant features for prediction, the initial list of features could be further reduced by using feature selection before training of the initial model. Secondly, as any decision tree-based model,

it can be further optimized with hyperparameter tuning, which was not done in this contribution. Parameters such as tree depth, learning rate, number of parallel trees etc., could be used to further enhance the performance of the model.

Lastly, there were not any measurements done in terms of computational latency to prove if prediction using the model is fast enough to be used in a high update rate positioning system. Rather, this contribution demonstrated that it is possible to use information based on ranging residuals to perform ML-based coordinate offset estimation. The latency tests for different ML and filtering schemes were performed in the subsequent chapter of this thesis (Publication II).

5 UWB end coordinate correction using ML

This chapter describes the methods used for estimating UWB coordinate accuracy with different ML models, evaluating their performance, and also applying their predictions to enhance coordinate accuracy and precision using different filtering schemes. The latter introduces Kalman filtering as the method of choice, because of its capability to incorporate measurement uncertainty estimation for weighting estimated coordinates. In this contribution, it is proposed that the coordinate error, estimated by the ML model, is used in the Kalman filter as a measure of uncertainty to further enhance positioning performance. Publication I showed that it is possible to use an ML model to estimate the coordinate offset class with significant accuracy by using features primarily related to ranging residuals. Publication II leverages this research by investigating different ML models and their application in coordinate filtering schemes.

This chapter is based on Publication II:

- M. Tommingas, M. M. Alam, I. Mürsepp, and S. Ulp, “UWB Positioning Integrity Estimation Using Ranging Residuals and ML Augmented Filtering,” *IEEE Journal of Indoor and Seamless Positioning and Navigation*, vol. 2, pp. 205–218, 2024

5.1 Background and motivation

Using ranging residuals for positioning error estimation provides an alternative to the usual methods incorporating CS-based information. One of the main motivations for this approach is the fact that range-based information can be accessed and processed faster than analyzing UWB channel characteristics. However, applying an ML model, which is trained with residual-based features, and incorporating it in coordinate offset predictions still adds to certain processing latency. Estimating this processing time was one of the goals for this research. Additional focus was on three distinct ML methods: **Regression Tree, Random Forest, and Extreme Gradient Boosting**. In essence, these methods are based on hierarchical decision-making using features described in Section 2.6.1. However, because of their distinct model structure and complexity, these models were expected to perform with different degrees of prediction accuracy and processing time. For example, a simple regression tree model would process data faster than a complex Random Forest. However, a more complex model would give a more accurate prediction. Lastly, the trained models were applied in an adaptive coordinate filtering scheme. By leveraging the error prediction by the ML model and applying it as measurement uncertainty in an Adaptive Kalman Filter (AKF), the proposed solution was compared with other filtering methods in terms of coordinate accuracy and precision.

5.2 ML model training

Similarly to Publication I, the ML models were trained using features mainly based on ranging residuals. These included: average sum of squares, root mean square, mean absolute deviation, standard deviation, number of residuals, mean and variance. These statistics were calculated for lengthened, shortened, and overall residual sets. Statistical equations also contained averaging to remove the dependence on the size of available residuals. Additional features, related with position calculation and geometrical uncertainty, were also included. These were: LS and NLS solution distance, number of NLS convergence iterations, and dilution of precision. All feature calculation equations are presented in Section 2.6.1.

In addition to the UWB measurement data used in Publication I, additional data was collected at Auroom Kastre factory (Fig. 23) and Eliko office rooms (Fig. 15). The latter was used to provide additional training data for the models, while Auroom data was solely used for testing purposes representing a dataset, which wasn't applied in training. Similarly to the measurement campaign done at Krah Pipes OÜ factory, the measurements were collected at different points around the area, consisting of stationary UWB measurements using a 10 Hz update rate resulting in approximately 300 ranging sequences at each point. Disto S910 laser measurement device was used to measure UWB tag's true coordinates [139]. These were later used for calculating the true coordinate error, which was set as the response variable in the supervised learning. All data processing and model



Figure 15: Eliko office rooms with a UWB positioning network. The visible UWB anchors are highlighted with red ellipses.

training was done in RStudio environment with appropriate ML libraries [137]. For each ML method, 10-fold Cross-Validation (CV) was performed to select hyper-parameters that provide a sufficiently small prediction error against the validation set. CV also helps to generalize the model and mitigate possible overfitting. Cross-validation was done using the *caret* library with *trainControl* functions [140]. The training dataset was separated into 10 segments with 1 segment being the validation set. After applying appropriate hyperparameters in an initial model, a combination of the most important features was selected for the final model.

5.2.1 Regression tree model training

Cross-validation was done with the *caret* library, using the *rpart2* function, which is intended to compare different regression tree depths in terms of prediction error [141]. As can be seen in Fig. 16-A, a tree depth of 7 is the minimum which provides the smallest prediction RMSE. Additionally, the tree was pruned using a Complexity Parameter (CP), which helps to find a balance between an overly complex model and accuracy. As can be seen in Fig. 16-B, the chosen tree size of 7 corresponds to a tree CP of 0.025. The resulting regression tree structure can be seen in Fig. 17. At each node and leaf, the prediction value and percentage (rounded to nearest integer) of training observations from a subsequent node can be observed. For example, the left-most terminal node shows the prediction of 0.23 m error encompassing 65% of all training samples. In creating regression trees

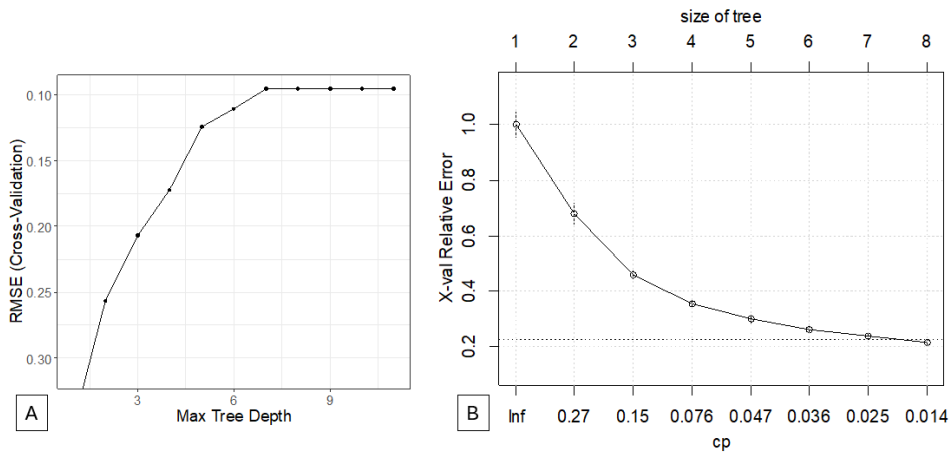


Figure 16: [A] Determining regression tree depth after 10-fold cross-validation. It can be seen that a tree depth of 7 is sufficient to provide the smallest prediction RMSE. [B] CP was used to further optimize the model. Tree depth of 7 corresponds to CP of 0.025. Figures from Publication II.

with the *rpart* library, features that provide the same goodness of split are removed [141]. Therefore, the resulting tree can differ significantly from cross-validated trees. Features that provide the best goodness of split in the regression tree model are shown in Fig. 18.

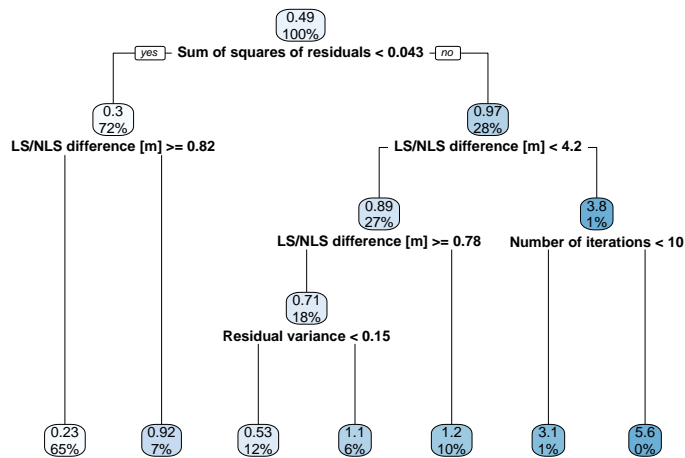


Figure 17: Final regression tree to be used in ML prediction. Figure from Publication II.

5.2.2 Random Forest model training

Cross-validation was done with *caret* library using *ranger* function, which allows to iterate over an increasing number of random predictors (function *mtry*) to find the suitable prediction RMSE. These predictors are a subset of features selected for building each tree [142]. As can be seen in Fig. 19-A, with 100 random trees, using more than 8 random predictors does not decrease prediction RMSE. The initial model was established using *ranger* function with all the features, a tree depth of 100, and 8 random predictors [143]. To re-

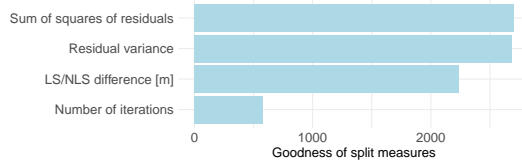


Figure 18: Features used in the final regression tree model, which are ordered based on the goodness of split in a regression tree. Figure from Publication II.

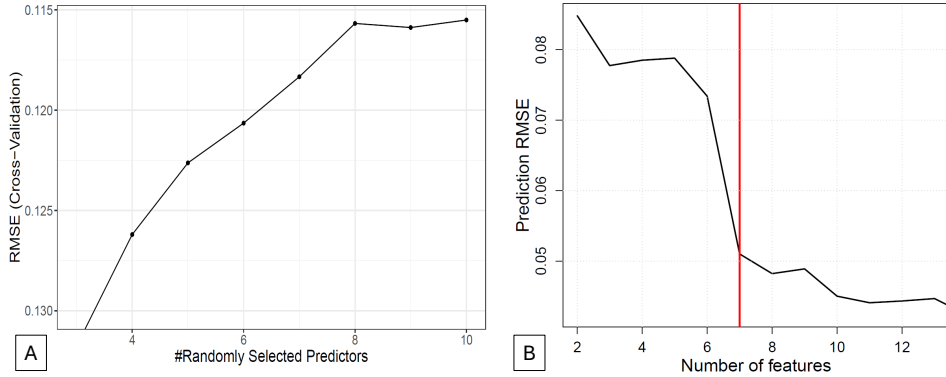


Figure 19: [A] Finding the optimal number of randomly selected Random Forest predictors using cross-validation with 100 random trees. It can be seen that using more than 8 randomly selected predictors, results in no decrease in cross-validation RMSE. [B] With 8 random predictors, using more than seven most important features results in no significant RMSE decrease in Random Forest prediction. Figures from Publication II.

duce complexity of the model, the *ranger* package provides *variable.importance* function to list the most important features based on node purity increase. By ordering these features based on their importance, new models were created and compared in terms of prediction accuracy. As can be seen in Fig. 19-B, choosing more than 7 most important features provides only a marginal decrease in prediction RMSE. Furthermore, choosing a higher number of features may not be useful for generalizing the model and can lead to overfitting. Most important Random Forest features can be seen in Fig. 20.

5.2.3 XGBoost model training

Cross-validation with hyper-parameter comparisons was done using *caret* package with *xgbTree* function. In this contribution tree depth and number of boosting iterations were considered. By comparing different hyperparameter values, it can be seen in Fig. 21-A that choosing more than 150 boosting iterations and a tree depth of 5, results in only a marginal decrease in prediction RMSE. To avoid an overly complex model, these values were chosen for the initial model, which was created with the *xgboost* library [136].

Features inherent in the model can be ordered in terms of their informational gain using *xgb.importance* function. By selecting consecutive combinations of most important features, different XGBoost models can be built and compared in terms of their prediction accuracy. As can be seen in Fig. 21-B, choosing more than 8 most important features presents in only a marginal decrease in prediction RMSE and therefore were chosen for the final model. These most important features are presented in Fig. 22.

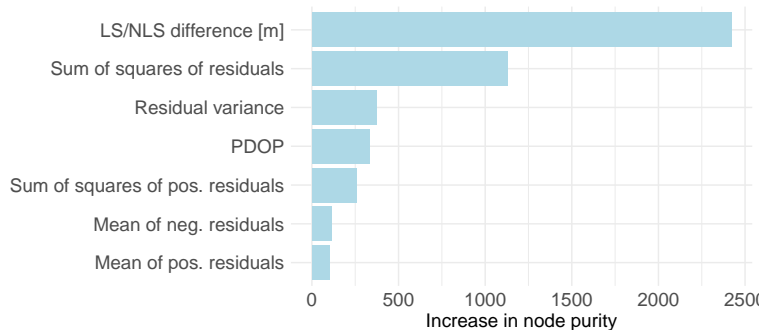


Figure 20: Set of features in the final model that provide the biggest node purity increase in Random Forest prediction. Figure from Publication II.

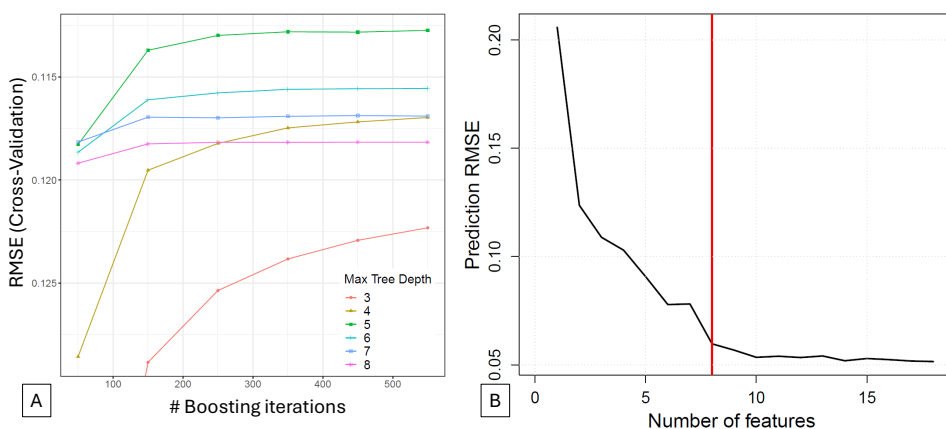


Figure 21: [A] Determining suitable XGBoost tree depth and number of boosting iterations using cross-validation. Tree depth 5 and 150 boosting iterations are chosen parameters for the model. Choosing a higher number of iterations results in no significant decrease in RMSE and might lead to overfitting. [B] Using more than 8 features has no significant impact on XGBoost prediction accuracy. Figures from Publication II.

5.3 Coordinate filtering

Coordinate filtering is used to smooth out noisy measurements and improve the overall accuracy and precision of the end coordinate. In the current context, the filter refines end coordinates, while considering the uncertainty of measurements (prediction) and previously filtered coordinates. While in a traditional Kalman Filter (KF) the process and measurement noise have fixed values, then in real-life applications it can be seen that measurement uncertainty is a dynamic value, affected by external factors such as NLoS or multipath. Therefore, it is preferable to know the measurement uncertainty at every ranging calculation. As shown in previous sections, the ML model aims to predict the end coordinate offset from true value, based on the input feature values. Since the direction of the error with regard to x , y , and z axes is unknown, this prediction can be considered as uncertainty in all three axes. By implementing the prediction as a dynamic measurement uncertainty in an AKF, it is hypothesized that positioning performance can be improved further.

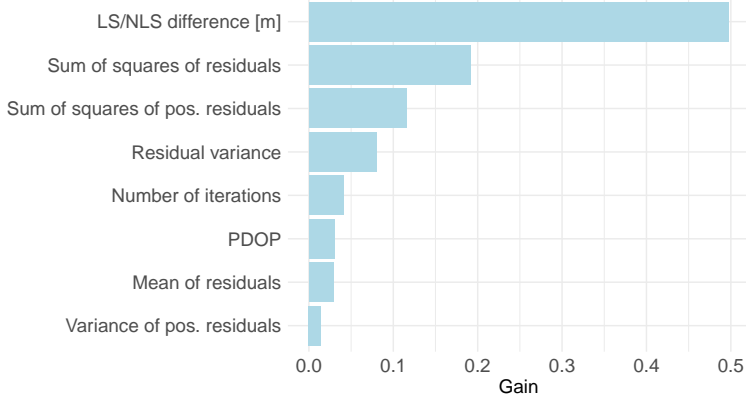


Figure 22: Features used in the final XGBoost model. Figure from Publication II.

5.3.1 KF and AKF filtering

In this work, the main difference between a KF and AKF is in the application of the \mathbf{R} matrix, which represents positioning measurement uncertainty. In KF, the diagonal elements of \mathbf{R} in (50) were set as fixed values $\text{diag}(0.01, 0.01, 0.01)$ corresponding to the precision of the DW1000 device [120] with:

$$\mathbf{R}_{\text{KF}} = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix} = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}. \quad (50)$$

However, AKF measurement uncertainty in (51) is updated at each iteration as the end coordinate is calculated with ML prediction $\hat{\mathbf{D}}_{\text{ML}}$ added to the variance of diagonal elements as

$$\mathbf{R}_{\text{AKF}} = \begin{bmatrix} 0.01 + \hat{D}_{\text{ML}} & 0 & 0 \\ 0 & 0.01 + \hat{D}_{\text{ML}} & 0 \\ 0 & 0 & 0.01 + \hat{D}_{\text{ML}} \end{bmatrix}. \quad (51)$$

In essence, the ML prediction drives the filtering process by dynamically changing measurement uncertainty i.e., weighting each incoming estimated coordinate with the associated uncertainty. In KF, EKF, and AKF, the process noise matrix \mathbf{Q} has constant values $\text{diag}(0.01, 0.01, 0.01)$. In the beginning of Alg. 1, the state transition matrix \mathbf{A} , state covariance \mathbf{P}_0 , and observation matrix \mathbf{H} are initialized as 3-by-3 identity matrices. $\hat{\mathbf{X}}_0$ represents the first converged NLS solution from (19), \mathbf{Z}_k is the measurement vector and \mathbf{I} is a 3-by-3 identity matrix.

5.3.2 EKF filtering

ML-driven AKF is also compared with the Extended Kalman Filter (EKF), which is capable of dealing with non-linear problems such as multilateration described in 2.2. In contrast to KF and AKF, which predict and correct coordinates, EKF makes state corrections using residuals between measured distances \mathbf{Z}_k and distances to the last estimated coordinates. In Alg. 1 state correction step $\mathbf{H}_k \hat{\mathbf{X}}_k^-$ is replaced with \mathbf{D}_k^- where:

Algorithm 1 Kalman filter algorithm

Input: $\hat{\mathbf{X}}_0, \mathbf{Z}_k, \mathbf{P}_0, \mathbf{Q}, \mathbf{R}$

Output: $\hat{\mathbf{X}}_k$

Initialize $\mathbf{A}, \mathbf{P}_0, \mathbf{H}, \mathbf{I}$

Prediction step

for $k = 1, \dots, \infty$

1: State prediction $\hat{\mathbf{X}}_k^- = \mathbf{A}\hat{\mathbf{X}}_{k-1}$

2: Covariance prediction $\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q}$

Correction step

3: Kalman gain $\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$

4: State correction $\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_k^- + \mathbf{K}_k (\mathbf{Z}_k - \mathbf{H}_k \hat{\mathbf{X}}_k^-)$

5: Covariance correction $\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-$

return $\hat{\mathbf{X}}_k, \mathbf{P}_k$

end for

$$\mathbf{D}_k^- = \begin{bmatrix} \sqrt{(x_k^- - x_1)^2 + (y_k^- - y_1)^2 + (z_k^- - z_1)^2} \\ \sqrt{(x_k^- - x_2)^2 + (y_k^- - y_2)^2 + (z_k^- - z_2)^2} \\ \vdots \\ \sqrt{(x_k^- - x_n)^2 + (y_k^- - y_n)^2 + (z_k^- - z_n)^2} \end{bmatrix} \quad (52)$$

with x_k^- , y_k^- and z_k^- representing coordinates from last iteration. Measurement vector \mathbf{Z}_k represents current iteration distance equations with added measurement noise

$$\mathbf{Z}_k = \begin{bmatrix} \sqrt{(x_k - x_1)^2 + (y_k - y_1)^2 + (z_k - z_1)^2} + v_1 \\ \sqrt{(x_k - x_2)^2 + (y_k - y_2)^2 + (z_k - z_2)^2} + v_2 \\ \vdots \\ \sqrt{(x_k - x_n)^2 + (y_k - y_n)^2 + (z_k - z_n)^2} + v_n \end{bmatrix}, \quad (53)$$

where \mathbf{v}_k represents measurement noise vector, which has covariance matrix \mathbf{R}_k as $\text{diag}(0.01, 0.01, 0.01)$. Process noise matrix \mathbf{Q} is also set as $\text{diag}(0.01, 0.01, 0.01)$.

With EKF, the entire NLS approximation process discussed in 2.2 may be bypassed and do linearization through the observation matrix \mathbf{H}_k , which is comprised of first-order partial derivatives [120]:

$$\mathbf{H}_k = \begin{bmatrix} \frac{\partial d_1(x_k, y_k, z_k)}{\partial x_k} & \frac{\partial d_1(x_k, y_k, z_k)}{\partial y_k} & \frac{\partial d_1(x_k, y_k, z_k)}{\partial z_k} \\ \frac{\partial d_2(x_k, y_k, z_k)}{\partial x_k} & \frac{\partial d_2(x_k, y_k, z_k)}{\partial y_k} & \frac{\partial d_2(x_k, y_k, z_k)}{\partial z_k} \\ \vdots & \vdots & \vdots \\ \frac{\partial d_n(x_k, y_k, z_k)}{\partial x_k} & \frac{\partial d_n(x_k, y_k, z_k)}{\partial y_k} & \frac{\partial d_n(x_k, y_k, z_k)}{\partial z_k} \end{bmatrix}, \quad (54)$$

where derivatives correspond to

$$\frac{\partial d_i(x_k, y_k, z_k)}{\partial x_k} = \frac{x_k - x_i}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2 + (z_k - z_i)^2}}, \quad (55)$$

$$\frac{\partial d_i(x_k, y_k, z_k)}{\partial y_k} = \frac{y_k - y_i}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2 + (z_k - z_i)^2}}, \quad (56)$$

$$\frac{\partial d_i(x_k, y_k, z_k)}{\partial z_k} = \frac{z_k - z_i}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2 + (z_k - z_i)^2}}. \quad (57)$$

In the context of coordinate calculation, skipping the NLS coordinate calculations (and convergence iterations) makes EKF computationally less demanding. On the other hand, a poor LS coordinate in the state vector can affect the filtering process and result in an inaccurate coordinate. Therefore, for comparison purposes, EKF was provided with a converged NLS coordinate as the initial state vector.

5.4 ML model testing and application for filtering

The proposed ML models and filtering schemes were tested using a real-life dataset gathered at an industrial site at Auroom Kastre factory (Fig. 23). Altogether 40 different measurement points were established around the ground floor. Similarly to the measurement campaigns at Krah Pipes and Eliko office, UWB ranging data were collected using 10 Hz update rate, resulting in approximately 300 measurement sequences per point. Additionally, the tag's true coordinates were measured with the Disto S910 laser measurement device for benchmarking. As stated in Section 5.2, Auroom dataset was not introduced during the training of the ML models. Therefore, it represents a suitable collection of unknown data to be used for model validation and filtering performance evaluation. ML performance



Figure 23: Manufacturing area inside Auroom Kastre factory. Red ellipses highlight visible UWB anchors. Figure from Publication II.

on predicting the test set response variables was summarized with the Cumulative error Distribution Function (CDF) in Fig. 24 and in Table 4. The latter includes commonly used regression performance indicators such as RMSE, Mean Square Error (MSE), and Mean Absolute Error (MAE) [144]. Expectedly, the Regression Tree model has the worst performance in terms of these metrics. This is due to the limitations arising from a single predictor tree as compared to significantly larger models of XGBoost and Random Forest. Moreover, the latter method provided the best regression metrics out of the three models.

Table 4: Performance of ML models on predicting the test set response variables. Table from Publication II.

	RMSE [m]	MSE [m ²]	MAE [m]
XGBoost	1.28	1.64	0.36
Regression tree	1.37	1.87	0.46
Random Forest	1.18	1.4	0.33

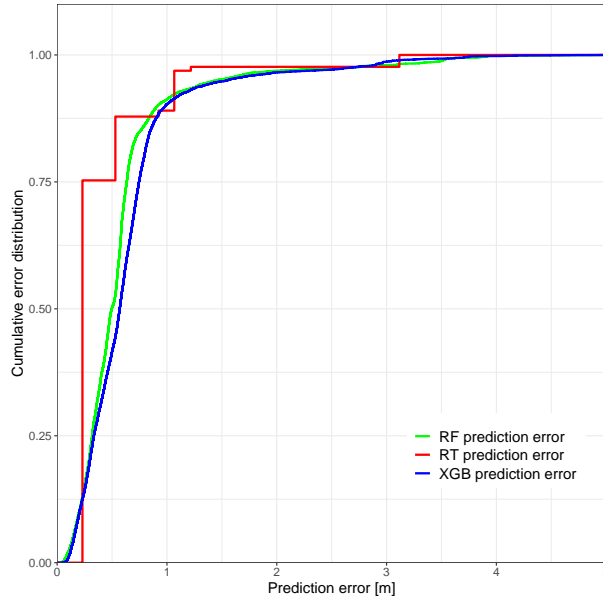


Figure 24: CDF of prediction errors. It can be seen that a regression tree provides more distinguished prediction error levels based on decisions from a single tree as shown in Fig. 17. RF and XGB predictions are more refined at the cost of more complex models. Figure from Publication II.

Next, the predictions of the models were applied in coordinate filtering schemes and the filtered end coordinates were compared with pre-measured true coordinates (x_T, y_T, z_T) . The following metrics were used to evaluate positioning accuracy and precision: Mean Location Error (MLE), RMSE, Distance Root Mean Square error (DRMS), Mean Radial Spherical Error (MRSE) and maximum error [78], [145]:

1) 2D metrics:

$$MLE_{2D} = \frac{\sum_{i=1}^n \sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2}}{n}, \quad (58)$$

$$RMSE_{2D} = \sqrt{\frac{\sum_{i=1}^n [(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2]}{n}}, \quad (59)$$

$$DRMS = \sqrt{\sigma_x^2 + \sigma_y^2}, \quad (60)$$

$$MAX_{2D} = \max_{i \in n} (\sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2}). \quad (61)$$

2) 3D metrics:

$$MLE_{3D} = \frac{\sum_{i=1}^n \sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2 + (z_T - \hat{z}_i)^2}}{n}, \quad (62)$$

$$RMSE_{3D} = \sqrt{\frac{\sum_{i=1}^n [(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2 + (z_T - \hat{z}_i)^2]}{n}}, \quad (63)$$

$$MRSE = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}, \quad (64)$$

$$MAX_{3D} = \max_{i \in n} (\sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2 + (z_T - \hat{z}_i)^2}). \quad (65)$$

Overall statistics summarizing all 40 measurement points can be seen in Tables 5 and 6. It can be seen that in 2D positioning, ML-driven filtering has approximately 0.1 m less MLE and 0.3 m smaller DRMS than compared to ordinary KF. In 3D positioning, the ML augmentation results in approximately 0.25 m smaller MLE and 0.6 m smaller MRSE than KF. While all AKF schemes performed at a similar level, it was seen that Random Forest had slightly better results in terms of overall maximum error in 2D and 3D positioning. It is also worth noting that NLS had smaller MLE than EKF in both 2D and 3D positioning. This can be explained with the solution convergence process inherent in the NLS optimization as explained in Section 2.2. In EKF algorithm no solution optimization was performed.

Table 5: Overall metrics for 2D positioning. Table from Publication II.

	MLE 2D [m]	RMSE 2D [m]	DRMS [m]	Max. error 2D [m]
NLS	0.46	0.95	0.85	11.16
KF	0.43	0.72	0.57	7.01
AKF + XGB	0.28	0.29	0.11	0.62
AKF + RF	0.28	0.29	0.1	0.55
AKF + RT	0.27	0.28	0.11	0.63
EKF	0.62	0.96	0.78	6.28

Lastly, proposed methods were compared in terms of elapsed processing time to investigate their feasibility in a high position update rate system. Benchmarking was done in the RStudio environment using built-in ML libraries *xgboost*, *ranger*, *rpart*, and *microbenchmark*. The hardware specification of the computer was Intel(R) Core(TM) i5-7300U CPU @ 2.60 GHz with 16 GB RAM. In Table 7 it can be seen the amount of delay ML adds to the filtering scheme. Ordinary Kalman filter performs the fastest while EKF being 3.5 times slower. However, ML prediction adds computational delay, with XGBoost and regression tree being approximately 18 times slower than KF and Random Forest being the slowest. The

Table 6: Overall metrics for 3D positioning. Table from Publication II.

	MLE 3D [m]	RMSE 3D [m]	MRSE [m]	Max. error 3D [m]
NLS	0.8	1.36	1.17	14.04
KF	0.74	1.05	0.8	8.78
AKF + XGB	0.48	0.5	0.18	0.94
AKF + RF	0.48	0.5	0.18	0.9
AKF + RT	0.51	0.53	0.2	1.07
EKF	2.86	3.26	1.94	11.73

latter is most probably due to prediction making process inherent in the Random Forest algorithm. As described in Section 2.5.2, prediction time is one of its limitations, as it requires querying each tree in the forest.

Table 7: Single iteration time for filtering and prediction. The ratio shows proportional relation between mean latencies of the filtering schemes. As the KF had the fastest processing time, the other filtering solutions were compared respectively. For example, the EKF algorithm had a 3.5 times higher single iteration latency than KF. The minimum elapsed time and mean time for KF is the same due to rounding. Table from Publication II.

	Min. time [ms]	Mean time [ms]	Max. time [ms]	Ratio
KF	0.04	0.04	0.06	1
AKF + XGB	0.68	0.71	0.72	17.75
AKF + RF	14.77	15.49	16.74	387.25
AKF + RT	0.68	0.72	0.9	18
EKF	0.12	0.14	0.21	3.5

Considering that all ML-driven filtering methods improved 2D and 3D positioning approximately on the same level, then in terms of latency, they added a significant delay compared to ordinary filtering. While these models could be applied in a high update rate (e.g., 10 or 20 Hz) positioning system, the Random Forest was significantly slower compared to RT and XGB models. Furthermore, XGBoost-augmented filtering performed slightly faster than a single regression tree. Such result may be related to the ML library's implementation, hardware specification, and efficiency of the code. The developed XGBoost model was also applied in the Eliko RTLS UWB positioning solution using the XGBoost C Package [146] with a prediction time delay of approximately 1 ms. The system hardware consisted of Intel(R) Xeon(R) W-2123 CPU @ 3.60 GHz with 16 GB RAM.

5.5 Discussion

In summary, this publication developed three distinct decision tree-based ML models with the aim to predict UWB positioning error. It was shown that using this prediction as a measurement uncertainty in a filtering scheme, the performance of UWB positioning can be improved when compared to traditional filtering methods. In 2D positioning, the MLE and RMSE improved ca. 10 cm and 40 cm, respectively. Furthermore, the ML prediction is sufficiently fast to be used in a real-life positioning system with XGB model showing 0.7 ms average latency. The performance of ML-augmented filtering motivated further investigation in its application in a UWB/GNSS sensor fusion solution as demonstrated in the subsequent chapter.

6 GNSS and UWB sensor fusion with ML-based uncertainty estimation

This chapter covers the methods used for sensor fusion of GNSS and UWB positioning systems using ML-based uncertainty estimations. In previous publications it was shown that ML can be effectively used for estimating positioning uncertainty and applied for improving the accuracy and precision of UWB end coordinate. Publication III leverages this knowledge and incorporates ML-based uncertainty estimation with GNSS positioning as well, with the aim of using two distinct ML models in a seamless indoor-outdoor positioning scheme. The estimated position uncertainties of both UWB and GNSS ML models are incorporated in adaptive sensor fusion and filtering, which refines the estimated end coordinate while considering the uncertainty estimation by the ML models at each position update. The entire proposed solution can be summarized with a flowchart in Fig. 25.

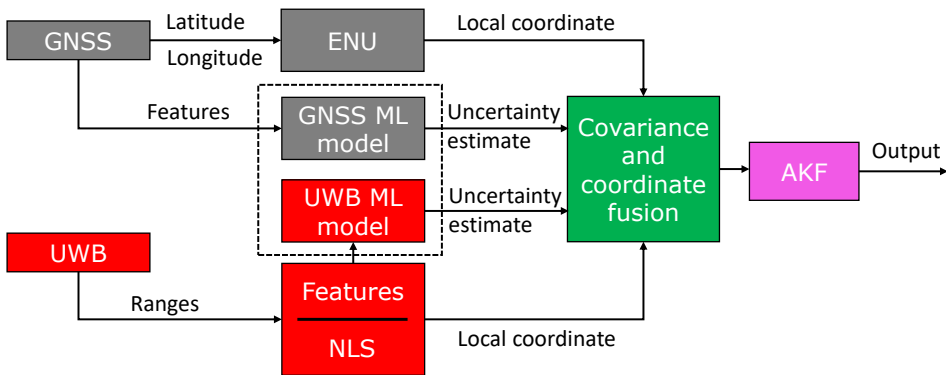


Figure 25: Flowchart of UWB and GNSS ML augmented sensor fusion as proposed by the author. The dashed box highlights the contribution in developing two distinct ML models for respective sensor uncertainty estimation. Sensor coordinates and their dynamically changing position uncertainties are then fused and filtered to produce the final coordinate at the output. Figure from Publication III.

This chapter is based on Publication III:

- M. Tommingas, T. Laadung, S. Varbla, I. Mürsepp, and M. Mahtab Alam, “UWB and GNSS Sensor Fusion Using ML-Based Positioning Uncertainty Estimation,” *IEEE Open Journal of the Communications Society*, vol. 6, pp. 2177–2189, 2025

6.1 Background and motivation

Determining the position of an object using a combination of positioning sensors whether indoors, outdoors, or in transitional environments, presents significant challenges as the system performance is highly dependent on the operational environment [147]. For instance, GNSS excels in open-sky conditions with clear satellite reception. However, indoor or dense urban environments severely attenuate satellite signals, degrading GNSS positioning accuracy significantly [8], [36]. While seamless indoor-outdoor positioning accuracy and precision may be enhanced by implementing additional sensors (e.g., inertial measurement unit or wheel sensor), the end coordinate still depends also on the accuracy of each positioning sensor. Estimating the positioning integrity at each position update is one of key components in achieving reliable coordinate, especially in transition areas, where the performance of both indoor and outdoor sensors may be compromised.

In the literature, a common method to measure the uncertainty of position is by using the Dilution of Precision (DoP) parameter, which indirectly shows the level of geometrical uncertainty in an area relative to servicing nodes (e.g., GNSS satellites or UWB anchors) [72]. However, DoP does not account for other factors that may contribute to positioning performance. For example, GNSS positioning is affected by several types of other error sources including receiver instrument issues, satellite signal propagation path (e.g., NLoS, multipath), and the space segment [21]. Considering that DoP considers only a small part of an entire ensemble of error sources, it is proposed to estimate positioning uncertainty with an alternative solution by using an ML model. In addition to considering geometrical uncertainty, several other features can be included such as the number of servicing satellites, quality and age of correction, deviation of positioning error, etc.

6.2 Data collection

For UWB positioning uncertainty estimation, this publication employs XGBoost ML model that was developed in Publication II. However, the current publication concentrates on the training of the GNSS ML model using features described in Section 2.6.2. The training data was collected during a measurement campaign conducted at an office building site as shown in Fig. 26. Firstly, the location of the GNSS RTK base station was established

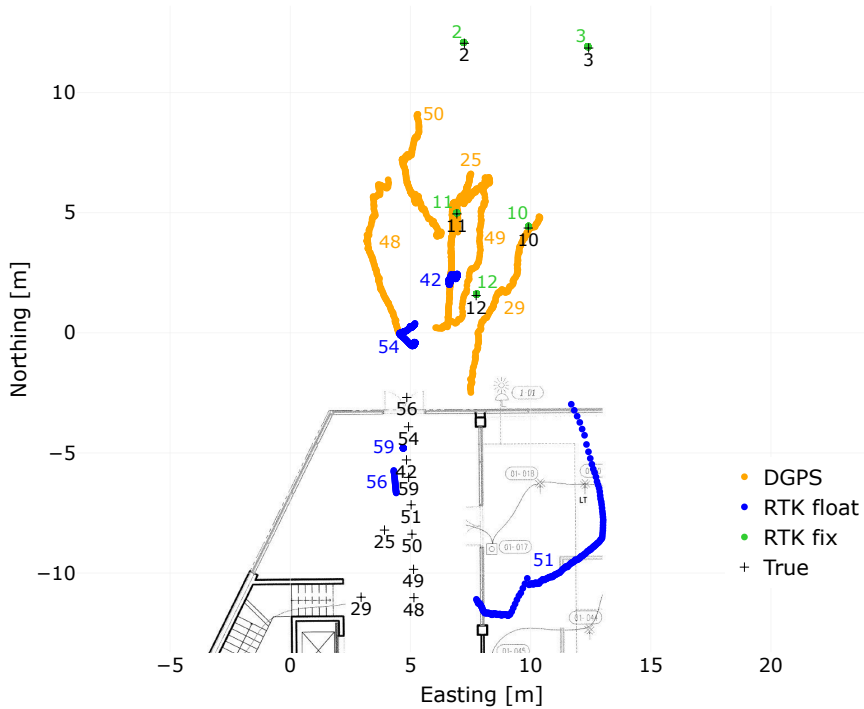


Figure 26: Layout of the GNSS measurement campaign at the Eliko office building. Static measurements were taken indoors, near-building, and in outdoor areas. For clarity, only 15 measurement points out of a total 60 measurements are shown in this figure. Each true coordinate is paired with a respective measurement. The blue and orange traces mark the highly inaccurate and imprecise DGPS and RTK float solutions taken indoors. Measurements that were taken closer to the building door, were also more accurate and precise, while points with RTK fix solution (marked with green) had the best performance. Figure from Publication III.

at a nearby geodetic location and assigned with global coordinates. The respective local coordinates were set as (0, 0). Next, the laser measurement device Disto S910 was set up near the building, so that the true coordinates of all measurement points can be gathered [139]. The goal was to take static GNSS measurements indoors, in semi-obstructed areas, and outdoors to mimic real-life dynamic changes in GNSS data. The measurements were collected for 30 seconds at 60 different measurement points with a 10 Hz update rate. The GNSS receiver collected the training data based on the same features as shown in Table 2. As the true coordinates were also measured at each point, the coordinate offset between the estimated GNSS position and the true coordinate was set as a response variable for each set of features.

During the measurement campaign, it was seen that correction quality changes relative to the operational area. For example, indoors with the unavailability of RTK, the main correction method was DGPS. Near the building edge, the main correction quality was RTK float. Only in clear-sky conditions, the RTK integer ambiguities were resolved, resulting in high accuracy RTK fix. More detailed description of GNSS correction qualities is given in Section 2.7. The entire training dataset can be viewed in terms of changes in correction quality as shown in Fig. 27. As expected, GNSS RTK provides the best positioning accuracy compared to other correction methods. The floating-point RTK had the largest accuracy range between approximately 1 m to 15 m. Indoors, at the furthest distance from the door, the main correction method was DGPS, which produced coordinates with varying offsets ranging from approximately 2 m to 25 m.

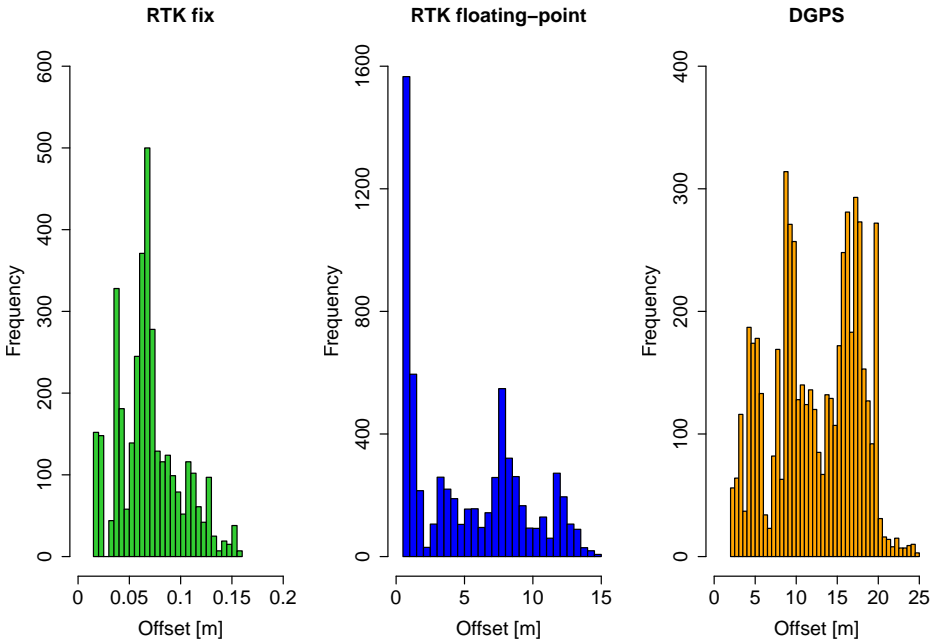


Figure 27: Histograms for all three GNSS correction qualities taken during the measurement campaign. Emphasis is on the distribution of coordinate offsets and their values with respect to each correction. It can be hypothesized that such distribution already provides insight in predicting the magnitude of coordinate error. Figure from Publication III.

6.3 ML model training

After data collection, several adjustments had to be made to ensure correct format of the dataset. Firstly, collected GNSS coordinates were transformed into the local frame of reference using ENU method described in Section 2.4. By calculating the Euclidean distance between the local coordinates and respective true coordinates, coordinate error was set as the response variable for each set of respective feature values. Next, to ensure unbiased training data, the entire dataset was sampled and divided into three equal-sized subsets according to correction quality: DGPS, RTK floating-point, and RTK fix. Lastly, the data was mixed and separated into training and testing sets with 80% and 20%, respectively.

To develop the GNSS ML model, supervised learning was done using *xgboost* library in RStudio environment [136], [137]. After data partitioning, 80 percent of the data was used for 10-fold cross-validation to select suitable hyperparameter values for the initial model. XGBoost was the ML method of choice for its high-performance metrics as shown in Publication II. Similarly to developing the UWB model, the chosen hyperparameters were the number of boosting iterations and tree depth [148]. RStudio provides appropriate cross-validation *train*, *xgbTree* and *trainControl* functions with the *caret* library [140]. The training dataset, which consisted of collected features and their response variables, was separated into 10 segments with 1 segment being the validation set. This approach helps to choose more generalized hyperparameter values [149]. The number of boosting iterations and tree depth were compared in terms of prediction RMSE as shown in Fig. 28. Hyperparameter values were deliberately limited, which can help avoid overfitting and an overly complex model [150]. Also, it can be seen that a model with a tree depth of 7 and 100 boosting iterations presents no significant increase in prediction performance. Using the chosen hyperparameters, the initial model with all the features was

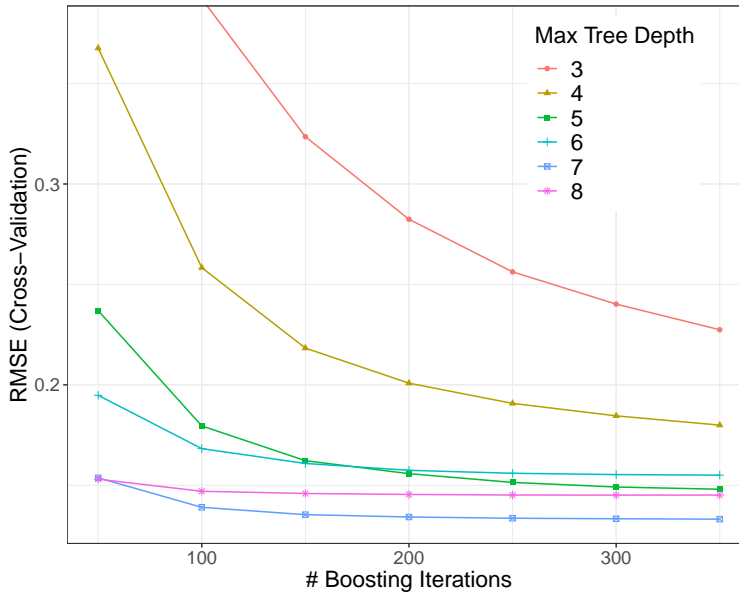


Figure 28: Prediction RMSE with different hyperparameter values. Tree depth and the number of boosting iterations were limited to 7 and 100 respectively as these values provide sufficient prediction accuracy and help avoid overfitting and an overly complex model. Figure from Publication III.

built with the *xgboost* function. XGBoost library's *xgb.importance*, outputs features that provide the most informational gain in making the prediction. Using the initial model, the inherent features were ranked in descending order. Next, by selecting a sequential combination of features, prediction RMSE was observed to select the number of features that provide a sufficiently small prediction RMSE. As shown in Fig. 29-A, more than 5 features provide no significant improvement in predicting test set response values. In contrast, choosing more features may lead to overfitting and an overly complex model [151]. The final selected features and their informational gain are shown in Fig. 29-B.

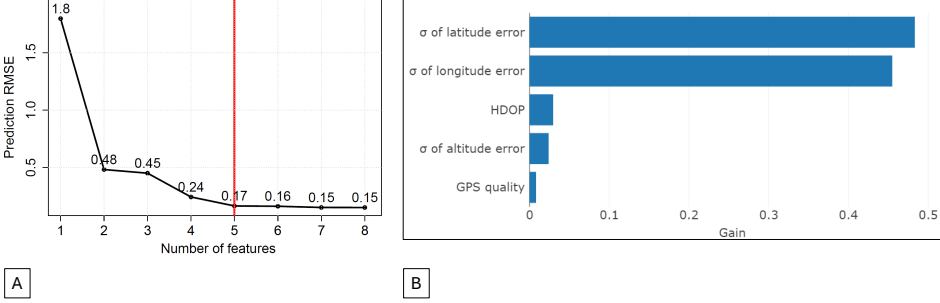


Figure 29: [A] The prediction RMSE as a function of the number of features. Using more than 5 features has no significant impact on prediction accuracy and may lead to overfitting of the model. [B] 5 features that provide the biggest informational gain in the XGBoost model. The gain quantifies how much a feature contributes in improving the models prediction. Figures from Publication III.

6.4 ML model testing

The final GNSS ML model was tested on the rest of 20% of data, which was not used in the training process. In Fig. 30-A, it can be seen that the ML model predicts GNSS response error in terms of different correction qualities with significant accuracy. The performance was evaluated with common regression metrics: RMSE, MSE, and MAE. Sample distributions of the test set and corresponding predictions can also be seen in Fig. 30-B.

6.5 Sensor fusion and filtering

In this work, complementary sensor fusion was used to combine the estimated UWB and GNSS coordinates and covariances [152]. Both UWB and GNSS ML models produce an uncertainty estimate, which is incorporated in the Kalman filter's covariance matrix of the respective sensor [153]. The predicted estimate ML_U of the UWB model and ML_G from the GNSS ML model were applied as:

$$\mathbf{R}_U = \begin{bmatrix} ML_U^2 & 0 \\ 0 & ML_U^2 \end{bmatrix}, \quad (66)$$

$$\mathbf{R}_G = \begin{bmatrix} ML_G^2 & 0 \\ 0 & ML_G^2 \end{bmatrix}. \quad (67)$$

Next, assuming measurements with normally distributed Probability Density Functions (PDF) a joint PDF \mathbf{R}_F was calculated as:

$$\mathbf{R}_F = (\mathbf{R}_G^{-1} + \mathbf{R}_U^{-1})^{-1}. \quad (68)$$

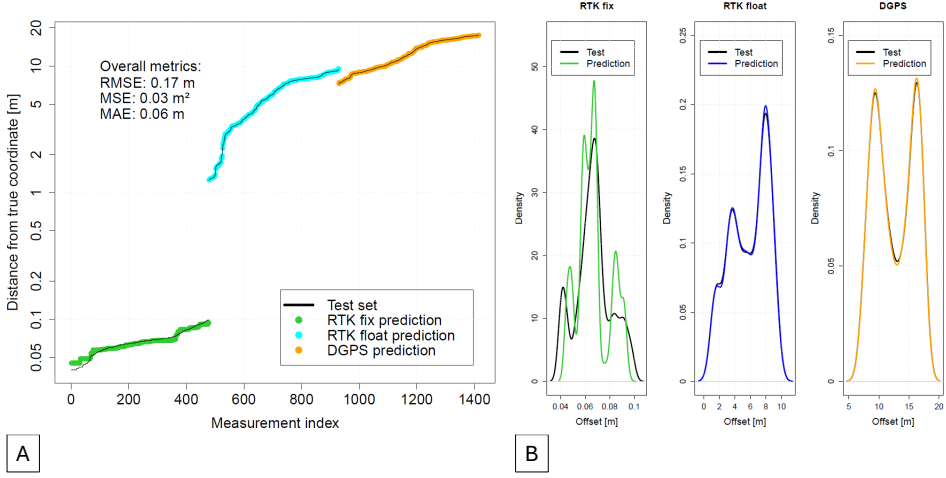


Figure 30: [A] Prediction error of GNSS ML model for different correction qualities. The vertical axis is presented in the logarithmic scale. [B] Sample density comparison of ML prediction and test set values for different correction qualities. Figures from Publication III.

Lastly, estimated coordinates $\hat{\mathbf{Z}}_U$ from UWB and $\hat{\mathbf{Z}}_G$ from GNSS, their respective covariances, and fused covariance are used to produce fused coordinates as:

$$\hat{\mathbf{Z}}_k = \mathbf{R}_F((\mathbf{R}_G^{-1}\hat{\mathbf{Z}}_G) + (\mathbf{R}_U^{-1}\hat{\mathbf{Z}}_U)). \quad (69)$$

The Adaptive Kalman Filter (AKF) is used to filter end coordinate estimates with fused covariances dynamically at each position update. The predictions of both ML models essentially drive the filtering process by dynamically changing measurement uncertainty i.e., whether to trust the measurement or the kinematic process. Since the proposed solution is meant to be applied to a moving object in 2D, the state transition matrix \mathbf{A} for position, velocity, and acceleration was established as:

$$\mathbf{A} = \begin{bmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} & 0 & 0 & 0 \\ 0 & 1 & \Delta t & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (70)$$

where Δt is measurement period of 0.1 s. And process noise matrix \mathbf{Q} as:

$$\mathbf{Q} = \begin{bmatrix} \frac{\Delta t^4}{4} & \frac{\Delta t^3}{2} & \frac{\Delta t^2}{2} & 0 & 0 & 0 \\ \frac{\Delta t^3}{2} & \Delta t^2 & \Delta t & 0 & 0 & 0 \\ \frac{\Delta t^2}{2} & \Delta t & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\Delta t^4}{4} & \frac{\Delta t^3}{2} & \frac{\Delta t^2}{2} \\ 0 & 0 & 0 & \frac{\Delta t^3}{2} & \Delta t^2 & \Delta t \\ 0 & 0 & 0 & \frac{\Delta t^2}{2} & \Delta t & 1 \end{bmatrix} \sigma_a^2, \quad (71)$$

where σ_a is random acceleration standard deviation with a heuristically chosen value of 10^{-4} m/s². As the filtering is done only for the x and y coordinates, the observation matrix is set up as:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \quad (72)$$

The order of steps inside the AKF is shown in Alg. 1. The initialization coordinates $\hat{\mathbf{X}}_0$ are extracted from the sensor, which has the lower uncertainty estimate based on the respective ML model. \mathbf{P}_0 represents the initial state covariance, which was set as $\mathbf{I} \cdot 100$, with \mathbf{I} being a 6-by-6 identity matrix.

6.6 Practical experiments and results

The proposed sensor fusion solution was tested with a moving testbed containing a GNSS RTK receiver, a UWB tag, a total station reflection prism, and a computer for data collection as shown in Fig. 31. The prism was used in conjunction with the stationary total station for measuring the true track. An additional computer was connected to the indoor UWB network, which consisted of 6 UWB anchors, and the total station data controller for real-time data collection. GNSS RTK base station was also set up nearby with premeasured geodetic coordinates [77]. During the test, the gathered data on the two computers were included with a Unix timestamp, and before the test, the computer clocks were synchronized against a time-server at *nettime.pool.ntp.org* with an approximate 2 ms offset.

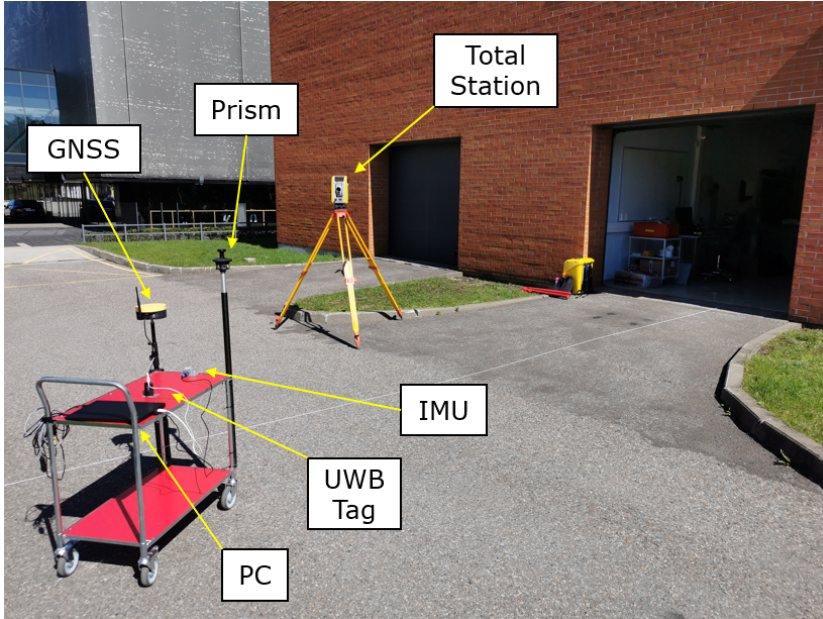


Figure 31: Test setup at the campus of Tallinn University of Technology. Figure from Publication III.

The initial total station setup was established using the resection method with three reference points [154]. These were acquired using a Trimble R12 GNSS receiver in RTK mode with three initializations (60 epochs per measurement). After establishing the initial base station, the three reference points were remeasured using the total station for

consistent coordinates, and the reference network was further densified. The improved network formed the basis for further validation measurements. By comparing the initial Trimble R12 and total station measurements, an approximate absolute accuracy of 10 mm could be assumed for validation surveys [3].

As can be seen in Table 8, the total station has a sample rate of 2.5 Hz. This presents a mismatch when coordinates from UWB and GNSS devices with 10 Hz sample rate are compared with the true track. In order to calculate performance metrics, sample rates of all systems must match at 10 Hz. Therefore, additional markers were added to the total station points through interpolation. Additionally, as seen in Fig. 31, the UWB tag and GNSS receiver were positioned with an offset regarding the reflection prism. Therefore, their output coordinates were rotated and shifted to match the location of the prism. Since IMU data was not used during the test, the direction of the trolley was calculated in post-processing using interpolated points of the total station.

Table 8: Positioning systems used in the test. Table from Publication III.

	Eliko RTLS UWB	Fieldbee L2 GNSS RTK	Trimble S6
Accuracy	0.2 m	0.01 m + 1 ppm CEP (RTK fix)	4 mm + 2 ppm and 2" angular
Sample rate	10 Hz	10 Hz	2.5 Hz
Method	AP-TWR	Real-time kinematic	LoS with reflection prism
Coordinate system	Local	Global	Global

An example of a test track is shown Fig. 32. The traverse started indoors with severely inaccurate GNSS position estimates with outliers approximately 60 m away. Similarly to the test campaign described in 6.2, only DGPS corrections were available. On the other hand, the UWB RTLS system provided stable coordinates with LoS from the servicing anchors. In the transition area, the UWB coordinates became expectedly more unstable, and the GNSS receiver applied first RTK float corrections, with coordinates converging on the transition area. After the GNSS receiver acquired the RTK fixed mode, the track was stable and accurate. Upon returning to the transition zone, the GNSS receiver fluctuated between RTK fix and floating-point modes. However, the receiver maintained a stable track even when using inferior correction qualities such as RTK-float or DGPS.

The test track data was then used for seamless sensor fusion solutions with different methods for estimating coordinate uncertainty. As stated in 6.1, one of the common methods for estimating the coordinate uncertainty in a sensor fusion system is by applying a dilution of precision parameter which describes geometrical uncertainty of an object relative to servicing nodes (e.g., satellites or anchors). In this work, HDOP was separately calculated for UWB coordinates, whereas GNSS provides HDOP in the NMEA message. At each position update, HDOP was used in the measurement uncertainty matrix of the Kalman filter instead of ML model estimates in (66) and (67). As can be seen in Fig. 33, using GNSS and UWB sensors with only HDOP as an uncertainty estimate poses certain limitations. The main difficulties appear at the beginning of the test, where the GNSS is most inaccurate. The HDOP provided by the GNSS receiver gives an incorrect estimate to a severely inaccurate coordinate and based on the weights in the R_U and R_G matrices, the Kalman filter estimates the end coordinate to lie approximately between the native UWB and GNSS coordinates. If the inferior correction modes, such as DGPS and RTK-floating

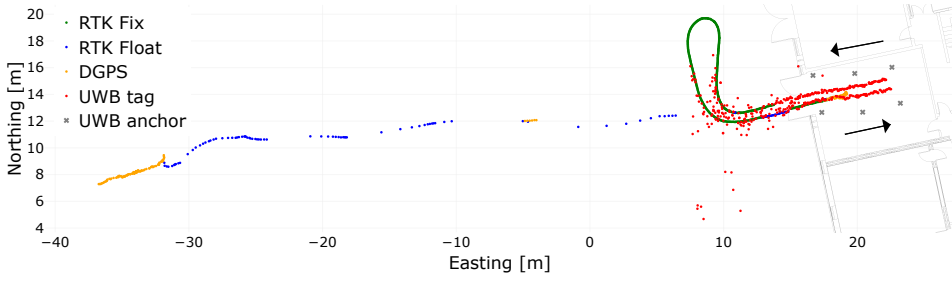


Figure 32: Coordinates from GNSS and UWB sensors taken along the indoor-outdoor-indoor movement path with arrows showing the movement direction. Traversing from the building, DGPS and RTK-float solutions are highly inaccurate, presenting a coordinate offset approximately 60 m from the starting point. Returning indoors, GNSS receiver fluctuates between different coordinate correction modes, while retaining a stable trajectory. Figure from Publication III.

point are not considered, then the fused track is much more stable and accurate as shown in Fig. 34-A. However, this solution entails several sensor dropouts, especially in the transition area, where typically RTK-floating point mode is used. Furthermore, when there was no RTK fix and no UWB data available, the Kalman filter solved the end coordinate as a stationary point.

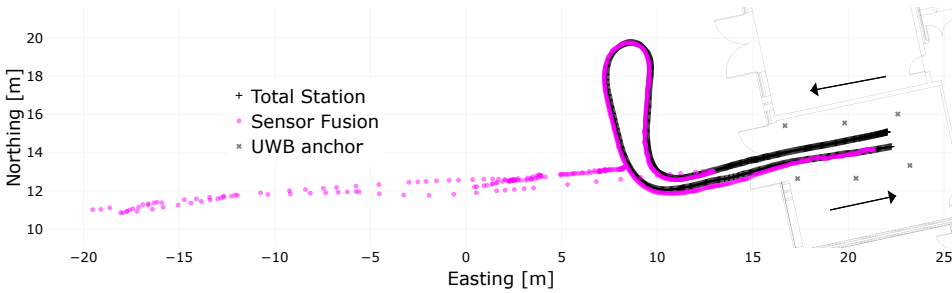


Figure 33: GNSS and UWB sensor fusion using HDOP-based uncertainty with AKF for benchmarking with all available GNSS corrections (DGPS, RTK float, and RTK fix). Figure from Publication III.

In contrast, the proposed ML model estimates the uncertainty based on training data, which already considers dynamically changing conditions. As was shown in Section 6.3, the training data consists of GNSS measurements with different correction qualities with respective features and true error response variables. Therefore, a properly trained ML model can consider a much broader variety of features than compared to DoP only. The benefit of incorporating ML models can be summarized in Fig. 34-B. At the beginning of the test, the GNSS ML model assigns a significant weight to the inaccurate GNSS coordinate, which results in the Kalman filter preferring the UWB coordinate instead. In the transition area with severely inaccurate DGPS and RTK float solutions, the ML model still assigns appropriate weights based on input features and suffers almost no dropouts. With RTK fix, the uncertainty is the smallest and the filtered end coordinate is the most stable and accurate. Returning indoors, it is notable that when transitioning from RTK fixed mode to RTK floating-point, the ML model assigns the latter with lower weights, than compared to RTK floating-point at the beginning of the test.

Table 9 shows overall sensor fusion results with different approaches. Positioning accuracy and precision were calculated with MLE (58), RMSE (59), and maximum error (61).

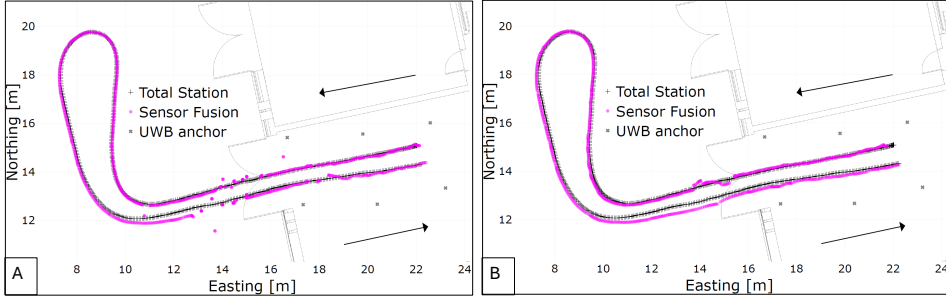


Figure 34: [A] GNSS RTK and UWB sensor fusion using HDOP-based uncertainty estimation with AKF for benchmarking with RTK fix only. [B] The proposed GNSS and UWB sensor fusion with ML-based uncertainty estimation with AKF using all available corrections. Figures from Publication III.

In Table 9 it can be seen how the HDOP-based approach with RTK fix has an MLE and RMSE at a similar level to ML-based estimation. However, the former method suffers from sensor dropouts in the transition area, resulting in a significant maximum error. Additionally, using HDOP with all available corrections (DGPS, RTK float, and RTK fix), results in an incorrect uncertainty estimation leading to a highly inaccurate fused coordinate. Lastly, the

Table 9: Comparison of different sensor fusion schemes. Table from Publication III.

	MLE [m]	RMSE [m]	Maximum error [m]
Proposed ML-based fusion with all corrections	0.16	0.18	0.49
Fusion with RTK fix and HDOP	0.14	0.19	1.29
Fusion with all corrections and HDOP	4.56	9.64	35.32

measurement uncertainties and their values are shown in Fig. 35. It illustrates how a raw GNSS HDOP compares to ML-based uncertainty estimations. As was shown in equations (68) and (69), the ML model estimate is reciprocal to the weight in AKF, resulting in a bias towards GNSS or UWB end coordinate.

6.6.1 Repeatability tests

The following section presents the results of additional tests that were not included in Publication III since these were carried out at a later date. The goal was to test the ML-augmented UWB-GNSS sensor fusion in terms of repeatability. The experiments were conducted on the same premises as the previous tests described in Section 6.6. However, the test track was set up for a different use case. The idea was to test the performance of the fused coordinate in three different scenarios: indoors, in the transition area, and outdoors to simulate forklift start-stop operations. The moving testbed made 4 stops along the track with 2 test runs, amounting to 8 stops as shown in Fig. 36. Similarly to previous tests, ML-augmented AKF was used for both GNSS and UWB sensor coordinates, with results shown in Fig. 37. The performance of fusion regarding the 8 distinct stopping points can be summarized in Table 10. Overall, the tests proved the feasibility of ML-augmented fusion in terms of repeatability and coordinate performance.

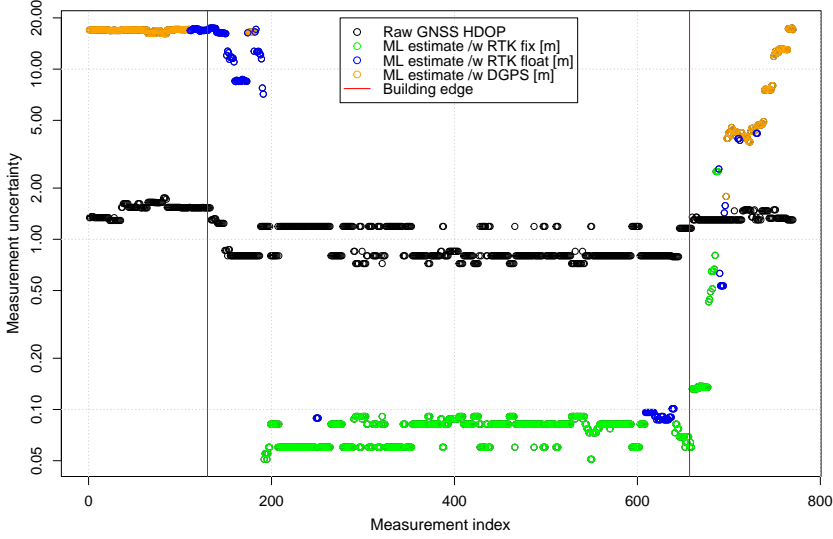


Figure 35: Distribution of measurement uncertainties during the test. It can be seen how ML estimates coordinate error in a significant range compared to raw GNSS HDOP values. Larger values (e.g., indoor DGPS) give smaller weight during coordinate filtering. On the other hand, smaller values (e.g., outdoor RTK fix), present a larger weight. HDOP is a unitless parameter.

Table 10: Positioning performance metrics during the repeatability tests.

	Stop 1	Stop 2	Stop 3	Stop 4	Stop 5	Stop 6	Stop 7	Stop 8
MLE [m]	0.30	0.07	0.11	0.17	0.10	0.07	0.14	0.32
RMSE [m]	0.31	0.07	0.12	0.17	0.11	0.07	0.14	0.34
MAX [m]	0.42	0.10	0.14	0.24	0.13	0.08	0.19	0.59

6.7 Discussion

In this publication, it was shown how ML-based positioning uncertainty estimation can be used in a seamless indoor-outdoor sensor fusion scheme. The goal was to apply dynamically changing ML-based weights to the coordinates of GNSS and UWB sensors and use this information for adaptive coordinate filtering. The resulting solution proved to outperform traditional solutions based on dilution of precision as a sole measure of coordinate uncertainty. While the initial tests showed promising results with ca. 80 cm smaller maximum error and almost no sensor dropouts, it can be suggested that this solution can be improved further. Firstly, the ML model can always be trained on additional data to make it more generalized. Furthermore, the UWB ML model was trained on positioning data gathered mostly in the vicinity of UWB anchors. However, additional data could be gathered similarly to the GNSS measurement campaign with measurements mimicking a potential indoor-outdoor route of the sensor. Secondly, the GNSS device used in this publication provided only a limited ensemble of features, when compared to tightly coupled solutions. Lastly, the proposed solution can possibly be improved with IMU for further stability and robustness.

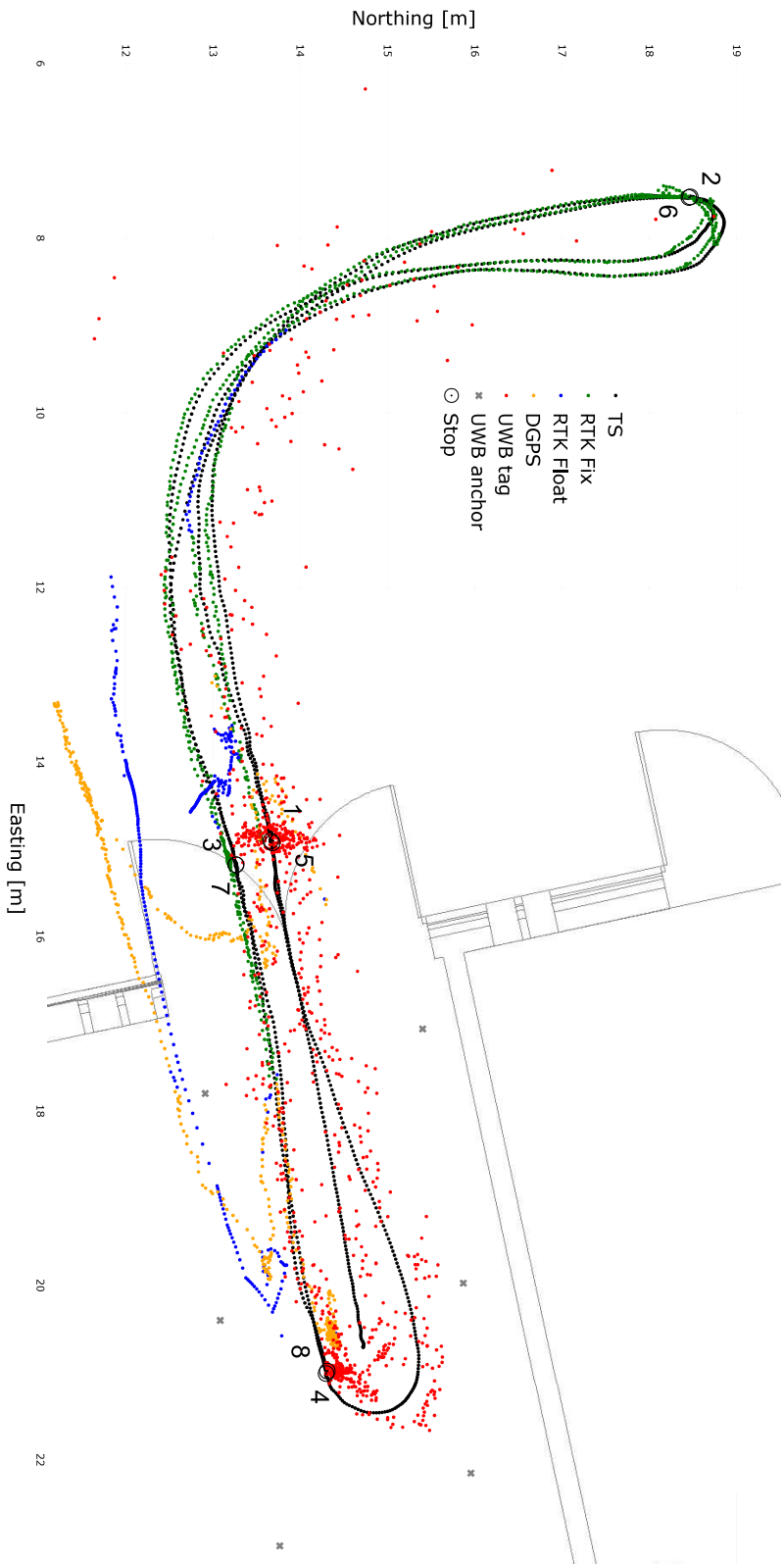


Figure 36: Layout of coordinates from individual sensors along the twice repeated test track with 8 stop points. Similarly to Fig. 32, a significant offset from the true track by DGPS and RTK float coordinates can be seen. Additionally, in the transition area there is a sparse UWB coordinate point cloud, due to the significant distance from UWB anchors resulting in a larger positioning uncertainty.



Figure 37: Layout of fused UWB-GNSS coordinates with ML-based uncertainty estimations along the twice repeated test track with 8 stop points. It can be seen how outliers and severely inaccurate measurements are mitigated and filtered according to the ML-augmented AKF algorithm. As expected, points 2 and 6 are most accurate and precise, relying only on GNSS RTK fix solutions. In the transition area as well as indoors, the point clouds were more sparse. Coordinate performance for these points are presented in Table 10.

7 Conclusion

This chapter presents an overview of the conducted research in the presented contributions, main findings, and key takeaways. Secondly, it provides answers to the initial research questions established in the introduction. Lastly, some potential directions for future research are presented.

7.1 Summary

The main goal of the presented contributions was to enhance positioning performance by leveraging information based on raw positioning data and incorporating them as features in distinct ML models. It diverges from typical solutions where channel statistics are used for ranging and positioning integrity assessment. Considering a high position update rate system, evaluating the channel statistics at each position update is currently an unfeasible solution. Instead, raw ranging information was employed in calculating various features as input for an ML model, which in turn attempts to predict end coordinate uncertainty. The latter is a key indicator in driving the coordinate filtering process, which aims to improve overall coordinate accuracy and precision. Leveraging the capabilities of various ML techniques, distinct models were trained and tested in all contributions of this thesis.

The first article investigated ML-based coordinate accuracy classification for UWB positioning using features of ranging residuals, position optimization metrics, and geometrical integrity values as features for an ML model. By incorporating real-life UWB ranging measurement data for training, it was found that the XGBoost ML model performed UWB-based positioning error classification with an overall high accuracy of 84%. Based on four classes of coordinate offsets with distinct ranges, the model could predict if the current coordinate belongs to a certain accuracy class. Furthermore, the XGBoost library distinguished ranging residuals as the one of the most important information sources for making the classification.

Leveraging the potential of using an ML model as a method for coordinate uncertainty estimations, the second publication concentrated on developing distinct decision tree-based ML models (Regression Tree, XGBoost, and Random Forest) for comparison purposes and incorporating their estimates in a coordinate filtering scheme to enhance the performance of UWB end coordinate. Two additional measurement campaigns were conducted to augment the initial dataset from Publication I and test the developed models on a separate real-life dataset. The predictions of the models were incorporated in coordinate filtering schemes to compare against non-ML filtering methods. The results showed significant improvement in both 2D and 3D positioning metrics. Additionally, it was shown that ML model prediction is sufficiently fast to be considered in a real-life high update rate positioning system.

The third publication investigated the application of UWB and GNSS ML-based uncertainty estimation in a seamless indoor-outdoor positioning scheme. Although this contribution applied the UWB XGBoost ML model from Publication II, the main focus was to investigate possible use of ML for GNSS-based positioning as well. By leveraging information extracted during a real-life GNSS measurement campaign, an ML model was trained to estimate positioning uncertainty similarly to the UWB model. Using the predictions of both UWB and GNSS ML models as measurement uncertainty, an adaptive Kalman Filtering scheme was developed for seamless indoor-outdoor positioning. The resulting solution proved to outperform traditional solutions based on dilution of precision as a sole measure of coordinate uncertainty with a significantly reduced maximum error and almost no sensor dropouts.

7.2 Research questions

This section provides answers to research questions from Section 1.3.

RQ1 How to assess UWB positioning uncertainty without the knowledge of channel statistics and is the alternative approach in estimating the uncertainty feasible to be used in a high update rate positioning system?

Incorporating information gathered from channel statistics is one of the most common methods to evaluate UWB ranging information and use this knowledge to mitigate or discard erroneous measurements. However, as an alternative solution, ranging- and positioning-based information can be used as well. Most notably, ranging residuals indirectly reflect erroneous measurements, which in turn affect the end coordinate. It was shown that using an ensemble of various non-channel statistics related features, UWB positioning uncertainty can be estimated with significant accuracy.

Range- and position-based information has a distinct advantage when used in a high update rate positioning solution, as this information is gathered and calculated from raw ranging measurements. This leads to a much smaller delay compared to channel statistics-based evaluations. It was shown that an ML-based approach is feasible to be applied in a positioning system with a high position update rate. For example, the XGBoost-augmented solution performed a single filtering and prediction with an average of 0.7 ms latency.

RQ2 How to improve UWB coordinate performance with supervised learning?

The ML models were developed with the initial aim in predicting end coordinate uncertainty using ranging- and positioning-based information. However, the main benefit comes from incorporating this estimate as a measurement uncertainty in an adaptive filtering scheme to mitigate erroneous measurements and outliers. Extensive UWB measurement campaigns were conducted with the aim of gathering UWB training data. The collected data along with true coordinates were used in supervised learning to develop distinct models to predict the coordinate uncertainty. By using ML-based estimates with coordinate filtering, the end coordinate accuracy and precision were improved with an overall reduction of 2D MLE and RMSE of approximately 10 cm and 40 cm, respectively. In 3D positioning, the same respective metrics were improved by approximately 25 cm and 50 cm.

RQ3 How to achieve a more comprehensive GNSS positioning uncertainty estimation?

Dilution of Precision (DoP) is a common metric used for GNSS positioning uncertainty estimation. However, as GNSS positioning performance is affected also by other factors than geometrical uncertainty, then DoP is not a comprehensive parameter. Alternatively, GNSS can output distinct features that indirectly reflect positioning performance such as: correction quality, number of satellites, age of correction, pseudorange error etc. By using ML techniques, a GNSS positioning uncertainty estimation model was proposed, which leverages inherent information from the GNSS receiver. By considering dynamically changing features at each position update, a more accurate uncertainty estimate is provided.

RQ4 How to augment multi-sensor fusion with ML for improved indoor-outdoor positioning?

Estimating measurement uncertainty accurately plays a key role when using coordinate filtering. Moreover, to achieve a reliable and seamless sensor fusion between a UWB indoor sensor network and an outdoor GNSS network, the uncertainty has to be accurately estimated at each position update. By leveraging the capabilities of Kalman filtering, the uncertainty estimates can be applied as weights for individual sensor coordinate outputs. As the ML models were trained on dynamic changes of real-life data, the model would give a prediction and bias toward using the position estimate of one of the sensors. This would

result in ML-augmented sensor fusion, where the model prediction essentially drives the filtering process, resulting in a more stable and accurate end coordinate. By implementing the solution on a real-life testbed moving between indoor and outdoor areas, it was observed that ML-augmented sensor fusion experienced almost no sensor dropouts with a mean and maximum positioning error of approximately 16 cm and 50 cm, respectively.

7.3 Future work

While the investigations and experiments in ML-assisted positioning presented in this thesis have yielded promising results, there is still potential for further improvement. Additional measurement and testing campaigns across different positioning environments, along with the integration of supplementary sensors, could further enhance positioning performance.

As noted for all the contributions in the thesis, ML models can always be improved with almost limitless quantities of positioning data. For example, Publication II concentrated on improving the accuracy and precision of scattered stationary UWB tag positions. However, the measurement campaign conducted in Publication III hinted that in indoor-outdoor positioning it would be preferable to gather training data similar to real-life movement paths. This would fare a better representation of dynamically changing feature values and possibly yield a better estimate for positioning uncertainty in LoS and NLoS conditions.

Another potential research topic would be to investigate the efficiency of non-decision tree-based ML algorithms (e.g., SVM, NN) for supervised learning. For instance, in a low update rate positioning system, prioritizing accuracy over prediction latency, these models could be compared with existing approaches.

Additionally, while the test campaign in Publication III confirmed the potential of using ML-based uncertainty estimation for seamless UWB-GNSS positioning, there is still further testing to be done in terms of model verification. Currently, the fusion solution was only applied to a single location. However, additional measurement data should be gathered in areas with different UWB anchor layouts and obstructions.

Another direction would introduce augmentation with additional hardware. Fusing GNSS and UWB sensors with IMU is a common method to further enhance the robustness of seamless positioning especially in the transition areas, where both UWB and GNSS performance is usually severely degraded. IMU can be used to suppress outliers and perform dead-reckoning to a certain extent in situations where both UWB and GNSS positioning information may be unavailable. Fusing all three sensors with ML-based uncertainty estimation is a potential topic to be investigated.

Real-life tests are essential in validating the proposed solutions. Indoor storage, manufacturing or production areas are typically cluttered with obstructions, presenting a challenging environment for accurate localization. Furthermore, as these areas usually extend also outdoors, seamless positioning and location awareness are essential for intelligent management. Consequently, the previously mentioned solutions would find application and testing in monitoring the position of industrial assets, thereby enhancing the efficiency of logistical and industrial processes.

List of Figures

1	A 2D trilateration (a variant of multilateration) scheme showing three reference points at (x_i, y_i) and distances d_i to the object at coordinates (x, y) . As the ranges intersect at only one point, the object's coordinates can be calculated. This figure illustrates an ideal scenario, without ranging error. Figure from Publication I.	15
2	An illustration of a 2D TDoA lateration scheme without ranging noise. Time-synchronized reference points measure the signal propagation time differences from an object. These differences, converted to distance differences (hyperbolas), are used with the reference point coordinates to determine the object's location at (x, y)	16
3	A 2D AoA scheme without ranging noise. The directions α_i of the signal from (x, y) are measured by directional antenna arrays at (x_i, y_i) , which are then used to estimate the position.....	17
4	A 2D trilateration scheme showing fixed coordinates at (x_i, y_i) and inaccurate pseudoranges d_i . The tag's location can be estimated in the vicinity of three overlapping circles at (\hat{x}, \hat{y}) . Figure from Publication I.	19
5	An inaccurately estimated position at (\hat{x}, \hat{y}) can result in an offset Δd_i between a measured range d_i and distance \hat{d}_i . The latter is calculated between the reference point (x_i, y_i) and the estimated position. The estimated position can also be located beyond a ranging measurement, resulting in a negative residual.....	21
6	In the Gauss-Krüger projection, a tangent transverse cylinder, which touches along a meridian (yellow line), is used to project the ellipsoidal model of the Earth. The scale distortions grow rapidly with increasing distance from the central meridian, so the projection width is limited to 3 to 6 degrees [47].	22
7	In the UTM projection, the cylinder touches along two standard parallels on the Earth's surface, reducing distortion near the central meridian. This figure is from [22].....	23
8	The ENU method maps spherical coordinates onto a flat tangent plane located at the project site. It is assumed that the flat plane represents a part of the surface on the sphere in a sufficiently small area.	24
9	In the Regression Tree, the input value is traversed through a series of nodes, where data features m are compared to preset thresholds t_n at each node resulting in prediction \hat{y}_n	26
10	A simplified example of the Random Forest structure. Instead of using a single regression tree, Random Forest relies on multiple trees generated using bootstrapped training data. Each tree produces an individual prediction, which is then used to average the result.	27
11	A general architecture of gradient boosting used in the XGBoost algorithm. The colored boxes are different features, which make up a decision/regression tree. Each subsequent tree improves the previous tree according to errors reflected by the prediction residuals. The summation of all trees results in a prediction estimate.	28

12	Differential correction-based GNSS positioning. While both DGNSS and RTK utilize corrections transmitted from a stationary GNSS base station, they differ in their method. DGNSS relies on pseudorange differences for coordinate corrections, whereas RTK incorporates carrier-phase observations.	34
13	Industrial site at Krah Pipes OÜ, which manufactures thermoplastic pipes in a complex industrial environment. Figure from Publication I.	50
14	Top 7 features that provide most information as proposed by XGBoost algorithm. It can be seen that lengthened residuals are the most important features in describing estimated position integrity. Figure from Publication I.	50
15	Eliko office rooms with a UWB positioning network. The visible UWB anchors are highlighted with red ellipses.	54
16	[A] Determining regression tree depth after 10-fold cross-validation. It can be seen that a tree depth of 7 is sufficient to provide the smallest prediction RMSE. [B] CP was used to further optimize the model. Tree depth of 7 corresponds to CP of 0.025. Figures from Publication II.	55
17	Final regression tree to be used in ML prediction. Figure from Publication II.	55
18	Features used in the final regression tree model, which are ordered based on the goodness of split in a regression tree. Figure from Publication II.	56
19	[A] Finding the optimal number of randomly selected Random Forest predictors using cross-validation with 100 random trees. It can be seen that using more than 8 randomly selected predictors, results in no decrease in cross-validation RMSE. [B] With 8 random predictors, using more than seven most important features results in no significant RMSE decrease in Random Forest prediction. Figures from Publication II.	56
20	Set of features in the final model that provide the biggest node purity increase in Random Forest prediction. Figure from Publication II.	57
21	[A] Determining suitable XGBoost tree depth and number of boosting iterations using cross-validation. Tree depth 5 and 150 boosting iterations are chosen parameters for the model. Choosing a higher number of iterations results in no significant decrease in RMSE and might lead to overfitting. [B] Using more than 8 features has no significant impact on XGBoost prediction accuracy. Figures from Publication II.	57
22	Features used in the final XGBoost model. Figure from Publication II.	58
23	Manufacturing area inside Auroom Kastre factory. Red ellipses highlight visible UWB anchors. Figure from Publication II.	60
24	CDF of prediction errors. It can be seen that a regression tree provides more distinguished prediction error levels based on decisions from a single tree as shown in Fig. 17. RF and XGB predictions are more refined at the cost of more complex models. Figure from Publication II.	61
25	Flowchart of UWB and GNSS ML augmented sensor fusion as proposed by the author. The dashed box highlights the contribution in developing two distinct ML models for respective sensor uncertainty estimation. Sensor coordinates and their dynamically changing position uncertainties are then fused and filtered to produce the final coordinate at the output. Figure from Publication III.	64

26	Layout of the GNSS measurement campaign at the Eliko office building. Static measurements were taken indoors, near-building, and in outdoor areas. For clarity, only 15 measurement points out of a total 60 measurements are shown in this figure. Each true coordinate is paired with a respective measurement. The blue and orange traces mark the highly inaccurate and imprecise DGPS and RTK float solutions taken indoors. Measurements that were taken closer to the building door, were also more accurate and precise, while points with RTK fix solution (marked with green) had the best performance. Figure from Publication III.	65
27	Histograms for all three GNSS correction qualities taken during the measurement campaign. Emphasis is on the distribution of coordinate offsets and their values with respect to each correction. It can be hypothesized that such distribution already provides insight in predicting the magnitude of coordinate error. Figure from Publication III.	66
28	Prediction RMSE with different hyperparameter values. Tree depth and the number of boosting iterations were limited to 7 and 100 respectively as these values provide sufficient prediction accuracy and help avoid overfitting and an overly complex model. Figure from Publication III.	67
29	[A] The prediction RMSE as a function of the number of features. Using more than 5 features has no significant impact on prediction accuracy and may lead to overfitting of the model. [B] 5 features that provide the biggest informational gain in the XGBoost model. The gain quantifies how much a feature contributes in improving the models prediction. Figures from Publication III.	68
30	[A] Prediction error of GNSS ML model for different correction qualities. The vertical axis is presented in the logarithmic scale. [B] Sample density comparison of ML prediction and test set values for different correction qualities. Figures from Publication III.	69
31	Test setup at the campus of Tallinn University of Technology. Figure from Publication III.	70
32	Coordinates from GNSS and UWB sensors taken along the indoor-outdoor-indoor movement path with arrows showing the movement direction. Traversing from the building, DGPS and RTK-float solutions are highly inaccurate, presenting a coordinate offset approximately 60 m from the starting point. Returning indoors, GNSS receiver fluctuates between different coordinate correction modes, while retaining a stable trajectory. Figure from Publication III.	72
33	GNSS and UWB sensor fusion using HDoP-based uncertainty with AKF for benchmarking with all available GNSS corrections (DGPS, RTK float, and RTK fix). Figure from Publication III.	72
34	[A] GNSS RTK and UWB sensor fusion using HDoP-based uncertainty estimation with AKF for benchmarking with RTK fix only. [B] The proposed GNSS and UWB sensor fusion with ML-based uncertainty estimation with AKF using all available corrections. Figures from Publication III.	73
35	Distribution of measurement uncertainties during the test. It can be seen how ML estimates coordinate error in a significant range compared to raw GNSS HDoP values. Larger values (e.g., indoor DGPS) give smaller weight during coordinate filtering. On the other hand, smaller values (e.g., outdoor RTK fix), present a larger weight. HDoP is a unitless parameter.	74

36	Layout of coordinates from individual sensors along the twice repeated test track with 8 stop points. Similarly to Fig. 32, a significant offset from the true track by DGPS and RTK float coordinates can be seen. Additionally, in the transition area there is a sparse UWB coordinate point cloud, due to the significant distance from UWB anchors resulting in a larger positioning uncertainty.....	75
37	Layout of fused UWB-GNSS coordinates with ML-based uncertainty estimations along the twice repeated test track with 8 stop points. It can be seen how outliers and severely inaccurate measurements are mitigated and filtered according to the ML-augmented AKF algorithm. As expected, points 2 and 6 are most accurate and precise, relying only on GNSS RTK fix solutions. In the transition area as well as indoors, the point clouds were more sparse. Coordinate performance for these points are presented in Table 10.	76

List of Tables

1	Comparison of techniques for improving GNSS positioning performance....	13
2	Features from NMEA messages [73].	33
3	Confusion matrix and prediction accuracy statistics. Prediction of different classes was done with varying accuracy. For example, if the there were 76 points that belonged to Class 2, then according to the model these were labeled as Class 1. With a 89% Sensitivity (true positive rate) the model could identify most of the points belonging to Class 1. Class 2 was harder to predict with a Sensitivity of ca. 62%, since Class 2 points accuracy is very close to Class 1. However, if the measurement point did not belong to Class 1 or Class 2 then the prediction was made with a high Specificity (true negative rate) of ca. 87% and 91%, respectively. Table from Publication I. ...	51
4	Performance of ML models on predicting the test set response variables. Table from Publication II.	61
5	Overall metrics for 2D positioning. Table from Publication II.	62
6	Overall metrics for 3D positioning. Table from Publication II.	63
7	Single iteration time for filtering and prediction. The ratio shows proportional relation between mean latencies of the filtering schemes. As the KF had the fastest processing time, the other filtering solutions were compared respectively. For example, the EKF algorithm had a 3.5 times higher single iteration latency than KF. The minimum elapsed time and mean time for KF is the same due to rounding. Table from Publication II. ...	63
8	Positioning systems used in the test. Table from Publication III.	71
9	Comparison of different sensor fusion schemes. Table from Publication III. .	73
10	Positioning performance metrics during the repeatability tests.	74

References

- [1] M. Tommingas, S. Ulp, M. M. Alam, I. Mürsepp, and T. Laadung, "Estimating UWB Positioning Integrity Based on Ranging Residuals," in *2023 24th International Conference on Applied Electromagnetics and Communications (ICECOM)*, pp. 1–5, IEEE, 2023.
- [2] M. Tommingas, M. M. Alam, I. Mürsepp, and S. Ulp, "UWB Positioning Integrity Estimation Using Ranging Residuals and ML Augmented Filtering," *IEEE Journal of Indoor and Seamless Positioning and Navigation*, vol. 2, pp. 205–218, 2024.
- [3] M. Tommingas, T. Laadung, S. Varbla, I. Mürsepp, and M. Mahtab Alam, "UWB and GNSS Sensor Fusion Using ML-Based Positioning Uncertainty Estimation," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 2177–2189, 2025.
- [4] N. H. A. Wahab, N. Sunar, S. H. Ariffin, K. Y. Wong, Y. Aun, *et al.*, "Indoor positioning system: A review," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 6, 2022.
- [5] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1067–1080, 2007.
- [6] L. Vaccari, A. M. Coruzzolo, F. Lolli, and M. A. Sellitto, "Indoor positioning systems in logistics: A review," *Logistics*, vol. 8, no. 4, p. 126, 2024.
- [7] S. Hayward, K. van Lopik, C. Hinde, and A. West, "A survey of indoor location technologies, techniques and applications in industry," *Internet of Things*, p. 100608, 2022.
- [8] A. Seszyk, S. Ioannou, and M. Raspopoulos, "A survey of 3D indoor localization systems and technologies," *Sensors*, vol. 22, no. 23, p. 9380, 2022.
- [9] A. Nessa, B. Adhikari, F. Hussain, and X. N. Fernando, "A survey of machine learning for indoor positioning," *IEEE access*, vol. 8, pp. 214945–214965, 2020.
- [10] T. Savić, X. Vilajosana, and T. Watteyne, "Constrained localization: A survey," *IEEE Access*, vol. 10, pp. 49297–49321, 2022.
- [11] T. Laadung, *Active-Passive Two-Way Ranging Protocol for Positioning Systems*. PhD thesis, TalTech, 2023.
- [12] A. Alarifi, A. Al-Salman, M. Alsaleh, A. Alnafessah, S. Al-Hadhrami, M. A. Al-Ammar, and H. S. Al-Khalifa, "Ultra wideband indoor positioning technologies: Analysis and recent advances," *Sensors*, vol. 16, no. 5, p. 707, 2016.
- [13] F. Despau, A. Van den Bossche, K. Jaffrès-Runser, and T. Val, "N-TWR: An accurate time-of-flight-based N-ary ranging protocol for Ultra-Wide band," *Ad Hoc Networks*, vol. 79, pp. 1–19, 2018.
- [14] M. A. Al-Ammar, S. Alhadhrami, A. Al-Salman, A. Alarifi, H. S. Al-Khalifa, A. Alnafessah, and M. Alsaleh, "Comparative survey of indoor positioning technologies, techniques, and algorithms," in *2014 International Conference on Cyberworlds*, pp. 245–252, IEEE, 2014.

- [15] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019.
- [16] T. Kim Geok, K. Zar Aung, M. Sandar Aung, M. Thu Soe, A. Abdaziz, C. Pao Liew, F. Hossain, C. P. Tso, and W. H. Yong, "Review of indoor positioning: Radio wave technology," *Applied Sciences*, vol. 11, no. 1, p. 279, 2020.
- [17] R. F. Brena, J. P. García-Vázquez, C. E. Galván-Tejada, D. Muñoz-Rodríguez, C. Vargas-Rosales, and J. Fangmeyer, "Evolution of indoor positioning technologies: A survey," *Journal of Sensors*, vol. 2017, 2017.
- [18] F. Zhang, L. Yang, Y. Liu, Y. Ding, S.-H. Yang, and H. Li, "Design and implementation of real-time localization system (RTLS) based on UWB and TDoA algorithm," *Sensors*, vol. 22, no. 12, p. 4353, 2022.
- [19] V. Barral, C. J. Escudero, and J. A. García-Naya, "NLOS classification based on RSS and ranging statistics obtained from low-cost UWB devices," in *2019 27th European Signal Processing Conference (EUSIPCO)*, pp. 1–5, IEEE, 2019.
- [20] E. D. Kaplan and C. Hegarty, *Understanding GPS/GNSS: principles and applications*. Artech house, 2017.
- [21] R. Zekavat and R. M. Buehrer, *Handbook of position location: theory, practice, and advances*. John Wiley & Sons, 2019.
- [22] J. V. Sickle, *GPS for land surveyors*. Taylor and Francis Group, 2008.
- [23] NovAtel Inc., "An Introduction to GNSS." <https://novatel.com/an-introduction-to-gnss>, 2015. Second Edition.
- [24] Y. Liu, S. Wang, W. Zuo, and M. Cen, "Indoor and outdoor seamless positioning method for open environment," in *2021 33rd Chinese Control and Decision Conference (CCDC)*, pp. 3206–3211, IEEE, 2021.
- [25] R. S. Campos and L. Lovisolo, *RF positioning: fundamentals, applications, and tools*. Artech House, 2015.
- [26] J. Guillory, D. Truong, and J.-P. Wallerand, "Multilateration with self-calibration: uncertainty assessment, experimental measurements and Monte-Carlo simulations," *Metrology*, vol. 2, no. 2, pp. 241–262, 2022.
- [27] P. Moravek, D. Komosny, M. Simek, and J. Muller, "Multilateration and flip ambiguity mitigation in ad-hoc networks," *Przegląd Elektrotechniczny*, vol. 2012, no. 05b, pp. 222–229, 2012.
- [28] C. McElroy, D. Neiryck, and M. McLaughlin, "Comparison of wireless clock synchronization algorithms for indoor location systems," in *2014 IEEE International Conference on Communications Workshops (ICC)*, pp. 157–162, IEEE, 2014.
- [29] S. Gezici and H. V. Poor, "Position estimation via ultra-wide-band signals," *Proceedings of the IEEE*, vol. 97, no. 2, pp. 386–403, 2009.
- [30] L. Taponecco, A. A. D'Amico, and U. Mengali, "Joint TOA and AOA estimation for UWB localization applications," *IEEE Transactions on Wireless Communications*, vol. 10, no. 7, pp. 2207–2217, 2011.

- [31] T. Gigl, G. J. Janssen, V. Dizdarevic, K. Witrisal, and Z. Irahauten, "Analysis of a UWB indoor positioning system based on received signal strength," in *2007 4th Workshop on Positioning, Navigation and Communication*, pp. 97–101, IEEE, 2007.
- [32] Y. Liu, Z. Yang, X. Wang, and L. Jian, "Location, localization, and localizability," *Journal of computer science and technology*, vol. 25, pp. 274–297, 2010.
- [33] J. J. Caffery, "A new approach to the geometry of TOA location," in *Vehicular Technology Conference Fall 2000. IEEE VTS Fall VTC2000. 52nd Vehicular Technology Conference (Cat. No. O0CH37152)*, vol. 4, pp. 1943–1949, IEEE, 2000.
- [34] Y. T. Chan and K. C. Ho, "A simple and efficient estimator for hyperbolic location," *IEEE transactions on signal processing*, vol. 42, no. 8, pp. 1905–1915, 1994.
- [35] G. Shen, R. Zetik, and R. S. Thoma, "Performance comparison of TOA and TDOA based location estimation algorithms in LOS environment," in *2008 5th workshop on positioning, navigation and communication*, pp. 71–78, IEEE, 2008.
- [36] M. Elsanhoury, P. Mäkelä, J. Koljonen, P. Välisuo, A. Shamsuzzoha, T. Mantere, M. Elmusrati, and H. Kuusniemi, "Precision positioning for smart logistics using ultra-wideband technology-based indoor navigation: A review," *IEEE Access*, vol. 10, pp. 44413–44445, 2022.
- [37] I. Guvenc, S. Gezici, F. Watanabe, and H. Inamura, "Enhancements to linear least squares localization through reference selection and ML estimation," in *2008 IEEE Wireless Communications and Networking Conference*, pp. 284–289, IEEE, 2008.
- [38] I. Guvenc, C.-C. Chong, and F. Watanabe, "NLOS identification and mitigation for UWB localization systems," in *2007 IEEE Wireless Communications and Networking Conference*, pp. 1571–1576, IEEE, 2007.
- [39] X. Li and Y. Wang, "Research on a factor graph-based robust UWB positioning algorithm in NLOS environments," *Telecommunication Systems*, vol. 76, pp. 207–217, 2021.
- [40] K. Zhang, C. Shen, Q. Zhou, H. Wang, Q. Gao, and Y. Chen, "A combined GPS, UWB and MARG locationing algorithm for indoor and outdoor mixed scenario," *Cluster computing*, vol. 22, no. 3, pp. 5965–5974, 2019.
- [41] J. Zhang, Z. Weng, and Y. Guo, "Distribution center location model based on gauss-kruger projection and gravity method," in *Journal of Physics: Conference Series*, vol. 1972, p. 012075, IOP Publishing, 2021.
- [42] M. Specht, "The evaluation of the positioning accuracy of the EGNOS and DGPS systems based on the long-term measurements in the years 2006-2014," *Polish Cartographical Review*, vol. 47, no. 2, pp. 99–108, 2015.
- [43] V. Di Pietra, P. Dabove, and M. Piras, "Loosely coupled GNSS and UWB with INS integration for indoor/outdoor pedestrian navigation," *Sensors*, vol. 20, no. 21, p. 6292, 2020.
- [44] V. Di Pietra, P. Dabove, and M. Piras, "Seamless navigation using UWB-based multisensor system," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pp. 1079–1084, IEEE, 2020.

- [45] P. Lv, M. Wang, B. Jin, and Y. Shen, "Seamless indoor and outdoor positioning of vehicles based on RTK/UWB," in *2022 6th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, pp. 1–8, IEEE, 2022.
- [46] Ł. Rykała, A. Typiak, and R. Typiak, "Research on developing an outdoor location system based on the ultra-wideband technology," *Sensors*, vol. 20, no. 21, p. 6171, 2020.
- [47] J. P. Snyder, *Map Projections - A Working Manual*, vol. 1395. US Government Printing Office, 1987.
- [48] W. Jiang, Z. Cao, B. Cai, B. Li, and J. Wang, "Indoor and outdoor seamless positioning method using UWB enhanced multi-sensor tightly-coupled integration," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10633–10645, 2021.
- [49] T. Meyer, "Grid, ground, and globe: Distances in the GPS era," *Surveying and Land Information Science*, vol. 62, no. 3, pp. 179–202, 2002.
- [50] M. W. A. Khan, E.-S. Lohan, and R. Piché, "Statistical sensor fusion of ultra wide band ranging and real time kinematic satellite navigation," in *2015 International Conference on Localization and GNSS (ICL-GNSS)*, pp. 1–6, IEEE, 2015.
- [51] X. Li, Z. Wu, Z. Shen, Z. Xu, X. Li, S. Li, and J. Han, "An indoor and outdoor seamless positioning system for low-cost UGV using PPP/INS/UWB tightly coupled integration," *IEEE Sensors Journal*, 2023.
- [52] T. Liu, B. Li, L. Yang, J. Qiao, W. Chen, *et al.*, "Tightly coupled integration of GNSS/UWB/VIO for reliable and seamless positioning," *IEEE Transactions on Intelligent Transportation Systems*, 2023.
- [53] G. Macgougan, K. O'Keefe, and R. Klukas, "Tightly-coupled GPS/UWB positioning," in *2009 IEEE International Conference on Ultra-Wideband*, pp. 381–385, 2009.
- [54] C. Wang, A. Xu, X. Sui, Y. Hao, Z. Shi, and Z. Chen, "A Seamless Navigation System and Applications for Autonomous Vehicles Using a Tightly Coupled GNSS/UWB/INS/Map Integration Scheme," *Remote Sensing*, vol. 14, no. 1, p. 27, 2021.
- [55] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *GPS - theory and practice*. Springer-Verlag, 1997.
- [56] M. Szabova and F. Duchon, "Survey of GNSS coordinates systems," *European Scientific Journal*, vol. 12, no. 24, 2016.
- [57] A. Siemuri, K. Selvan, H. Kuusniemi, P. Valisuo, and M. S. Elmusrati, "A systematic review of machine learning techniques for GNSS use cases," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 6, pp. 5043–5077, 2022.
- [58] W.-Y. Loh, "Classification and regression trees," *Wiley interdisciplinary reviews: data mining and knowledge discovery*, vol. 1, no. 1, pp. 14–23, 2011.
- [59] O. M. Badawy and M. A. B. Hasan, "Decision tree approach to estimate user location in WLAN based on location fingerprinting," in *2007 National Radio Science Conference*, pp. 1–10, IEEE, 2007.

- [60] C. Aggarwal, "Data mining the text book," 2015.
- [61] H. Yazdani, M. Doostizadeh, and F. Aminifar, "Forecast-aided power and flexibility trading of prosumers in peer to peer markets," *IET Renewable Power Generation*, vol. 17, no. 4, pp. 920–934, 2023.
- [62] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.
- [63] J. M. Rudd *et al.*, "An empirical study of downstream analysis effects of model pre-processing choices," *Open journal of statistics*, vol. 10, no. 5, pp. 735–809, 2020.
- [64] M. Y. Khan, A. Qayoom, M. S. Nizami, M. S. Siddiqui, S. Wasi, and S. M. K.-u.-R. Raazi, "Automated Prediction of Good Dictionary EXamples (GDEX): A Comprehensive Experiment with Distant Supervision, Machine Learning, and Word Embedding-Based Deep Learning Techniques," *Complexity*, vol. 2021, no. 1, p. 2553199, 2021.
- [65] Y. Li, C. Zou, M. Berecibar, E. Nanini-Maury, J. C.-W. Chan, P. Van den Bossche, J. Van Mierlo, and N. Omar, "Random forest regression for online capacity estimation of lithium-ion batteries," *Applied energy*, vol. 232, pp. 197–210, 2018.
- [66] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.
- [67] GitHub project webpage, "eXtreme Gradient Boosting." <https://github.com/dmlc/xgboost>. Accessed January 9, 2025.
- [68] A. Asselman, M. Khaldi, and S. Aammou, "Enhancing the prediction of student performance based on the machine learning XGBoost algorithm," *Interactive Learning Environments*, vol. 31, no. 6, pp. 3360–3379, 2023.
- [69] Y. Wang, Z. Pan, J. Zheng, L. Qian, and M. Li, "A hybrid ensemble method for pulsar candidate classification," *Astrophysics and Space Science*, vol. 364, pp. 1–13, 2019.
- [70] W. Chantaweesomboon, C. Suwatthikul, S. Manatrinon, K. Athikulwongse, K. Kaemarungsi, R. Ranron, and P. Suksompong, "On performance study of UWB real time locating system," in *2016 7th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES)*, pp. 19–24, 2016.
- [71] K. Mannay, N. Benhadjyoussef, M. Machhout, and J. Urena, "Location and positioning systems: Performance and comparison," in *2016 4th International Conference on Control Engineering & Information Technology (CEIT)*, pp. 1–6, IEEE, 2016.
- [72] Y. Guo, W. Li, G. Yang, Z. Jiao, and J. Yan, "Combining Dilution of Precision and Kalman Filtering for UWB Positioning in a Narrow Space," *Remote Sensing*, vol. 14, no. 21, p. 5409, 2022.
- [73] NovAtel, "GNSS Logs - NovAtel Documentation Portal." https://docs.novatel.com/OEM7/Content/Logs/Core_Logs.htm?tocpath=Commands%20%2526%20Logs%20%2527CGNSS%20Logs%20%2527C____0. Accessed September 19, 2024.
- [74] A. Stateczny, C. Specht, M. Specht, D. Brčić, A. Jugović, S. Widźgowski, M. Wiśniewska, and O. Lewicka, "Study on the positioning accuracy of GNSS/INS systems supported by DGPS and RTK receivers for hydrographic surveys," *Energies*, vol. 14, no. 21, p. 7413, 2021.

- [75] E. Fredeluces, T. Ozeki, N. Kubo, and A. El-Mowafy, "Modified RTK-GNSS for challenging environments," *Sensors*, vol. 24, no. 9, p. 2712, 2024.
- [76] H. Ma, Q. Zhao, S. Verhagen, D. Psychas, and X. Liu, "Assessing the performance of multi-GNSS PPP-RTK in the local area," *Remote sensing*, vol. 12, no. 20, p. 3343, 2020.
- [77] F. eFarmer BV, "Fieldbee L2 GNSS RTK base station." <https://www.fieldbee.com/wp-content/uploads/2020/09/FieldBee-L2-base-station-datasheet.pdf>. Accessed October 15, 2024.
- [78] B. Silva and G. P. Hancke, "Ranging error mitigation for through-the-wall non-line-of-sight conditions," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 11, pp. 6903–6911, 2020.
- [79] P.-C. Chen, "A non-line-of-sight error mitigation algorithm in location estimation," in *WCNC. 1999 IEEE Wireless Communications and Networking Conference (Cat. No. 99TH8466)*, vol. 1, pp. 316–320, IEEE, 1999.
- [80] X. Li, "An iterative NLOS mitigation algorithm for location estimation in sensor networks," *Proc. 15th IST Mobile Wireless Commun. Summit*, pp. 1–5, 2006.
- [81] L. Jiao, J. Xing, X. Zhang, J. Zhang, and C. Zhao, "LCC-Rwgh: A NLOS error mitigation algorithm for localization in wireless sensor network," in *2007 IEEE international conference on control and automation*, pp. 1354–1359, IEEE, 2007.
- [82] J. Xing, J. Zhang, L. Jiao, X. Zhang, and C. Zhao, "A robust wireless sensor network localization algorithm in NLOS environment," in *2007 IEEE International Conference on Control and Automation*, pp. 3244–3249, IEEE, 2007.
- [83] L. Jiao, J. Xing, and F. Y. Li, "Performance comparison of residual related algorithms for ToA positioning in wireless terrestrial and sensor networks," in *2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology*, pp. 278–283, IEEE, 2009.
- [84] Y.-T. Chan, W.-Y. Tsui, H.-C. So, and P.-c. Ching, "Time-of-arrival based localization under NLOS conditions," *IEEE Transactions on vehicular technology*, vol. 55, no. 1, pp. 17–24, 2006.
- [85] D. Liu, M.-C. Lee, C.-M. Pun, and H. Liu, "Analysis of wireless localization in nonline-of-sight conditions," *IEEE transactions on vehicular technology*, vol. 62, no. 4, pp. 1484–1492, 2013.
- [86] B. J. Silva and G. P. Hancke, "Non-line-of-sight identification without channel statistics," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, pp. 4489–4493, IEEE, 2020.
- [87] M. J. Bocus and R. J. Piechocki, "Passive unsupervised localization and tracking using a multi-static UWB radar network," in *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 01–06, IEEE, 2021.
- [88] S. Venkatesh and R. Buehrer, "Non-line-of-sight identification in ultra-wideband systems based on received signal statistics," *IET Microwaves, Antennas & Propagation*, vol. 1, no. 6, pp. 1120–1130, 2007.

- [89] E. García, P. Poudereux, Á. Hernández, J. Ureña, and D. Gualda, "A robust UWB indoor positioning system for highly complex environments," in *2015 IEEE International Conference on Industrial Technology (ICIT)*, pp. 3386–3391, IEEE, 2015.
- [90] V. Barral, C. J. Escudero, J. A. García-Naya, and P. Suárez-Casal, "Environmental cross-validation of NLOS machine learning classification/mitigation with low-cost UWB positioning systems," *Sensors*, vol. 19, no. 24, p. 5438, 2019.
- [91] S. Guo, Y. Zhang, X. Gui, and L. Han, "An improved PDR/UWB integrated system for indoor navigation applications," *IEEE Sensors Journal*, vol. 20, no. 14, pp. 8046–8061, 2020.
- [92] D.-H. Kim, G.-R. Kwon, J.-Y. Pyun, and J.-W. Kim, "NLOS identification in UWB channel for indoor positioning," in *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1–4, IEEE, 2018.
- [93] S. Wu, Y. Ma, Q. Zhang, and N. Zhang, "NLOS error mitigation for UWB ranging in dense multipath environments," in *2007 IEEE Wireless Communications and Networking Conference*, pp. 1565–1570, IEEE, 2007.
- [94] G. Falco, M. Pini, and G. Marucco, "Loose and tight GNSS/INS integrations: Comparison of performance assessed in real urban scenarios," *Sensors*, vol. 17, no. 2, p. 255, 2017.
- [95] J. M. Hansen, T. A. Johansen, N. Sokolova, and T. I. Fossen, "Nonlinear observer for tightly coupled integrated inertial navigation aided by RTK-GNSS measurements," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 3, pp. 1084–1099, 2018.
- [96] R. Langley, "NMEA 0183: A GPS receiver," *GPS world*, vol. 6, no. 7, pp. 54–57, 1995.
- [97] A. Dyukov, S. Choy, and D. Silcock, "Accuracy of speed measurements using GNSS in challenging environments," *Asian Journal of Applied Sciences*, vol. 3, no. 6, 2015.
- [98] K. Song, J. Li, M. Ren, and S. Liu, "Indoor and Outdoor Fusion Positioning System Based on UWB/GNSS/IMU Multi-Sensor Technology," in *Proceedings of the 2024 International Conference on Autonomous Driving and Intelligent Sensing Technology*, pp. 48–54, 2024.
- [99] Y. Yang, L. Song, and T. Xu, "Robust estimator for correlated observations based on bifactor equivalent weights," *Journal of geodesy*, vol. 76, pp. 353–358, 2002.
- [100] R. F. Keefe, A. M. Wempe, R. M. Becker, E. G. Zimbelman, E. S. Nagler, S. L. Gilbert, and C. C. Caudill, "Positioning methods and the use of location and activity data in forests," *Forests*, vol. 10, no. 5, p. 458, 2019.
- [101] M. Vasudha and G. Raju, "Comparative evaluation of IRNSS performance with special reference to positional accuracy," *Gyroscopy and Navigation*, vol. 8, pp. 136–149, 2017.
- [102] A. Liu, J. Wang, S. Lin, and X. Kong, "A dynamic UKF-based UWB/wheel odometry tightly coupled approach for indoor positioning," *Electronics*, vol. 13, no. 8, p. 1518, 2024.

- [103] L. Yao, M. Li, T. Xu, X. Dai, T. Jiang, P. Dai, S. Wang, and J. Xing, "GNSS/UWB/INS indoor and outdoor seamless positioning algorithm based on federal filtering," *Measurement Science and Technology*, vol. 35, no. 1, p. 015135, 2023.
- [104] B. Zhu, X. Tao, J. Zhao, M. Ke, H. Wang, and W. Deng, "An integrated GNSS/UWB/DR/VMM positioning strategy for intelligent vehicles," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 10842–10853, 2020.
- [105] Z. Sun, W. Gao, X. Tao, S. Pan, P. Wu, and H. Huang, "Semi-Tightly Coupled Robust Model for GNSS/UWB/INS Integrated Positioning in Challenging Environments," *Remote Sensing*, vol. 16, no. 12, p. 2108, 2024.
- [106] M. Specht, "Experimental studies on the relationship between HDOP and position error in the GPS system," *Metrology and Measurement Systems*, pp. 17–36, 2022.
- [107] H.-U. Kim and T.-S. Bae, "Deep learning-based GNSS network-based real-time kinematic improvement for autonomous ground vehicle navigation," *Journal of Sensors*, vol. 2019, no. 1, p. 3737265, 2019.
- [108] A. Kuratomi, "GNSS position error estimated by machine learning techniques with environmental information input," 2019.
- [109] G. Zhang, P. Xu, H. Xu, and L.-T. Hsu, "Prediction on the urban GNSS measurement uncertainty based on deep learning networks with long short-term memory," *IEEE Sensors Journal*, vol. 21, no. 18, pp. 20563–20577, 2021.
- [110] Z. Lyu and Y. Gao, "An SVM based weight scheme for improving kinematic GNSS positioning accuracy with low-cost GNSS receiver in urban environments," *Sensors*, vol. 20, no. 24, p. 7265, 2020.
- [111] Y. Xia, S. Pan, W. Gao, B. Yu, X. Gan, Y. Zhao, and Q. Zhao, "Recurrent neural network based scenario recognition with multi-constellation GNSS measurements on a smartphone," *Measurement*, vol. 153, p. 107420, 2020.
- [112] Q. Liu, Z. Huang, and J. Wang, "Indoor non-line-of-sight and multipath detection using deep learning approach," *GPS Solutions*, vol. 23, no. 3, p. 75, 2019.
- [113] R. Klus, J. Talvitie, and M. Valkama, "Neural network fingerprinting and GNSS data fusion for improved localization in 5G," in *2021 International Conference on Localization and GNSS (ICL-GNSS)*, pp. 1–6, IEEE, 2021.
- [114] A. Siemuri, H. Kuusniemi, M. S. Elmusrati, P. Välisuo, and A. Shamsuzzoha, "Machine learning utilization in GNSS — use cases, challenges and future applications," in *2021 International Conference on Localization and GNSS (ICL-GNSS)*, pp. 1–6, IEEE, 2021.
- [115] V. Navarro, R. Grieco, B. Soja, M. Nugnes, G. Klopotek, G. Tagliaferro, L. See, R. Falzarano, R. Weinacker, and J. VenturaTraveset, "Data fusion and machine learning for innovative GNSS science use cases," in *Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021)*, pp. 2656–2669, 2021.
- [116] Y. Liu, L. Liu, L. Yang, L. Hao, and Y. Bao, "Measuring distance using ultra-wideband radio technology enhanced by extreme gradient boosting decision tree (XGBoost)," *Automation in Construction*, vol. 126, p. 103678, 2021.

- [117] K. Bregar and M. Mohorčič, "Improving indoor localization using convolutional neural networks on computationally restricted devices," *IEEE Access*, vol. 6, pp. 17429–17441, 2018.
- [118] Z. Niu, H. Yang, L. Zhou, M. F. Taha, Y. He, and Z. Qiu, "Deep learning-based ranging error mitigation method for UWB localization system in greenhouse," *Computers and Electronics in Agriculture*, vol. 205, p. 107573, 2023.
- [119] J. Fan and A. S. Awan, "Non-line-of-sight identification based on unsupervised machine learning in ultra wideband systems," *IEEE Access*, vol. 7, pp. 32464–32471, 2019.
- [120] D.-H. Kim, A. Farhad, and J.-Y. Pyun, "UWB positioning system based on LSTM classification with mitigated NLOS effects," *IEEE Internet of Things Journal*, vol. 10, no. 2, pp. 1822–1835, 2022.
- [121] H. Wymeersch, S. Maranò, W. M. Gifford, and M. Z. Win, "A machine learning approach to ranging error mitigation for UWB localization," *IEEE transactions on communications*, vol. 60, no. 6, pp. 1719–1728, 2012.
- [122] J. Wei, H. Wang, S. Su, Y. Tang, X. Guo, and X. Sun, "NLOS identification using parallel deep learning model and time-frequency information in UWB-based positioning system," *Measurement*, vol. 195, p. 111191, 2022.
- [123] Z. Cui, Y. Gao, J. Hu, S. Tian, and J. Cheng, "LOS/NLOS identification for indoor UWB positioning based on Morlet wavelet transform and convolutional neural networks," *IEEE Communications Letters*, vol. 25, no. 3, pp. 879–882, 2020.
- [124] S. Angarano, V. Mazzia, F. Salvetti, G. Fantin, and M. Chiaberge, "Robust ultra-wideband range error mitigation with deep learning at the edge," *Engineering Applications of Artificial Intelligence*, vol. 102, p. 104278, 2021.
- [125] X. Yang, "NLOS mitigation for UWB localization based on sparse pseudo-input Gaussian process," *IEEE Sensors Journal*, vol. 18, no. 10, pp. 4311–4316, 2018.
- [126] J. B. Kristensen, M. M. Ginard, O. K. Jensen, and M. Shen, "Non-line-of-sight identification for UWB indoor positioning systems using support vector machines," in *2019 IEEE MTT-S International Wireless Symposium (IWS)*, pp. 1–3, IEEE, 2019.
- [127] T. Wang, K. Hu, Z. Li, K. Lin, J. Wang, and Y. Shen, "A semi-supervised learning approach for UWB ranging error mitigation," *IEEE Wireless Communications Letters*, vol. 10, no. 3, pp. 688–691, 2020.
- [128] A. G. Ferreira, D. Fernandes, S. Branco, A. P. Catarino, and J. L. Monteiro, "Feature selection for real-time NLOS identification and mitigation for body-mounted UWB transceivers," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–10, 2021.
- [129] A. Musa, G. D. Nugraha, H. Han, D. Choi, S. Seo, and J. Kim, "A decision tree-based NLOS detection method for the UWB indoor location tracking accuracy improvement," *International Journal of Communication Systems*, vol. 32, no. 13, p. e3997, 2019.

- [130] C. L. Sang, B. Steinhagen, J. D. Homburg, M. Adams, M. Hesse, and U. Rückert, "Identification of NLOS and multi-path conditions in UWB localization using machine learning methods," *Applied Sciences*, vol. 10, no. 11, p. 3980, 2020.
- [131] W. Li, T. Zhang, and Q. Zhang, "Experimental researches on an UWB NLOS identification method based on machine learning," in *2013 15th IEEE International Conference on Communication Technology*, pp. 473–477, IEEE, 2013.
- [132] C.-C. Chang, H.-W. Wang, Y.-X. Zeng, and J.-D. Huang, "Using machine learning approaches to improve ultra-wideband positioning," *Journal of Internet Technology*, vol. 22, no. 5, pp. 1021–1031, 2021.
- [133] J. Yim, "Introducing a decision tree-based indoor positioning technique," *Expert Systems with Applications*, vol. 34, no. 2, pp. 1296–1302, 2008.
- [134] D. M. Bates and D. G. Watts, *Nonlinear regression analysis and its applications*. Wiley, 1988.
- [135] T. Laadung and S. Ulp, "UWB Active-Passive Two-Way Ranging Protocol Data Captured in an Industrial Environment," 2023.
- [136] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou, M. Li, J. Xie, M. Lin, Y. Geng, Y. Li, and J. Yuan, *xgboost: Extreme Gradient Boosting*, 2022. R package version 1.6.0.1.
- [137] R Core Team, *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2021.
- [138] S. P. Rana, M. Dey, H. U. Siddiqui, G. Tiberi, M. Ghavami, and S. Dudley, "UWB localization employing supervised learning method," in *2017 IEEE 17th International Conference on Ubiquitous Wireless Broadband (ICUWB)*, pp. 1–5, IEEE, 2017.
- [139] Leica Geosystems AG, "Leica DISTO S910 User Manual." <https://shop.leica-geosystems.com/sites/default/files/2019-04/leica-disto-s910-user-manual-805080-808183-806677-en.pdf>. Accessed July 16, 2021.
- [140] Kuhn and Max, "Building Predictive Models in R Using the caret Package," *Journal of Statistical Software*, vol. 28, no. 5, p. 1–26, 2008.
- [141] T. Therneau and B. Atkinson, *rpart: Recursive Partitioning and Regression Trees*, 2022. R package version 4.1.19.
- [142] Z. Jiao, P. Hu, H. Xu, and Q. Wang, "Machine learning and deep learning in chemical health and safety: a systematic review of techniques and applications," *ACS Chemical Health & Safety*, vol. 27, no. 6, pp. 316–334, 2020.
- [143] M. N. Wright and A. Ziegler, "ranger: A fast implementation of random forests for high dimensional data in C++ and R," *Journal of Statistical Software*, vol. 77, no. 1, pp. 1–17, 2017.
- [144] J. S. Kushwah, A. Kumar, S. Patel, R. Soni, A. Gawande, and S. Gupta, "Comparative study of regressor and classifier with decision tree using modern tools," *Materials Today: Proceedings*, vol. 56, pp. 3571–3576, 2022.

- [145] G. Laveti, G. S. Rao, K. J. Rani, A. Nalinee, and A. M. Babu, "GPS receiver SPS accuracy assessment using LS and LQ estimators for precise navigation," in *2014 Annual IEEE India Conference (INDICON)*, pp. 1–5, IEEE, 2014.
- [146] xgboost developers, "XGBoost C Package." <https://xgboost.readthedocs.io/en/stable/c.html>, 2022. Accessed November 15, 2023.
- [147] N. El-Sheimy and Y. Li, "Indoor navigation: State of the art and future trends," *Satellite Navigation*, vol. 2, no. 1, p. 7, 2021.
- [148] S. Demir and E. K. Sahin, "An investigation of feature selection methods for soil liquefaction prediction based on tree-based ensemble algorithms using AdaBoost, gradient boosting, and XGBoost," *Neural Computing and Applications*, vol. 35, no. 4, pp. 3173–3190, 2023.
- [149] H. M. Aydin, M. A. Ali, and E. G. Soyak, "The analysis of feature selection with machine learning for indoor positioning," in *2021 29th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, IEEE, 2021.
- [150] R. G. Leiva, A. F. Anta, V. Mancuso, and P. Casari, "A novel hyperparameter-free approach to decision tree construction that avoids overfitting by design," *IEEE Access*, vol. 7, pp. 99978–99987, 2019.
- [151] J. Reunanen, *Overfitting in feature selection: Pitfalls and solutions*. PhD thesis, Aalto University, 2012.
- [152] A. Becker, *Kalman Filter from the Ground Up*. KalmanFilter.NET, 2023. First Edition.
- [153] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, 1960.
- [154] T. GmbH, "Resection computations in trimble access." <https://help.trimblegeospatial.com/TrimbleAccess/latest/en/PDFs/Access-Resection-Computations.pdf>. Accessed October 4, 2024.
- [155] Google AI, "Gemini 2.0 Flash (Large Language Model)." <https://ai.google.dev/models/gemini>, 2023. Accessed April 4, 2025.

Acknowledgements

I would like to thank Eliko Tehnoloogia Arenduskeskus OÜ and Thomas Johann Seebeck Department of Electronics at Tallinn University of Technology for giving me the opportunity to participate in the PhD programme.

I acknowledge with gratitude my esteemed supervisors, Professor Muhammad Mahtab Alam and Ivo Mürsepp from Tallinn University of Technology for their guidance, critical feedback and support. From Eliko I would like to thank Taavi Laadung and Sander Ulp for their supervision and providing valuable knowledge in operating with UWB systems.

I would also like to thank Professor Yannick Le Moullec and Aleksei Fjodorov for their valuable feedback and constructive discussion throughout the studies.

Further acknowledgements regarding financial support are as follows:

- European Union's Horizon 2020 Research programme under grant agreement No. 101058505;
- Internet of Intelligent Things project of Estonian IT Academy program and Estonian Research Council under Grant PUT-PRG424;
- Development of an industrial digital control system based on precise positioning technology ELIKO TAK and Atemix Automatika, nr. 2014-2020.4.02.21-0311;
- Increasing the knowledge intensity of Ida-Viru entrepreneurship.

Finally, the author would like to acknowledge the use of Google Gemini language model for minor language polishing, grammar checks, and stylistic improvements [155].

Abstract

Enhancing UWB and Multi-Sensor Positioning with ML-based Uncertainty Estimation

This thesis presents novel approaches for enhancing positioning performance of Ultra-Wideband (UWB) and multi-sensor positioning systems using Machine Learning (ML). Several models are proposed that leverage positioning-related information that indirectly reflects positioning uncertainty. The latter is a key indicator when considering different filtering schemes. In essence, a correct uncertainty estimate would give an appropriate weight for an accurate or inaccurate coordinate, resulting in mitigated outliers and overall enhanced performance of the coordinate.

The first contribution of this thesis investigates features based on UWB ranging residuals and positioning information for position integrity estimation. Usually, UWB-based positioning performance is enhanced through the use of Channel Statistics (CS). However, in practice this is not a feasible approach as gathering CS data takes too much time when incorporated in a high update rate UWB positioning system. In contrast, this contribution leverages information based on ranging residuals as they indirectly reflect errors in ranging and therefore in the end coordinate. A set of features, describing different statistics of ranging residuals, was employed in an Extreme Gradient Boosted (XGBoost) ML algorithm to train a model for end coordinate offset classification. These features included those used in literature as well as several novel ones. Notably, the dataset in this contribution was collected during a measurement campaign in a complex industrial environment with constantly changing Line-of-Sight/Non-Line-of-Sight (LoS/NLoS) conditions. Based on the results of a test set, the trained model could predict errors in the range of 0...0.2 m with an accuracy of 90% and an overall accuracy of 84%. These metrics show that using the proposed features, it is possible to predict UWB end coordinate integrity with high accuracy.

The second contribution focuses on the use of UWB ranging residuals and positioning-based information for coordinate integrity estimation and their use in a filtering scheme as a means of coordinate correction. By leveraging the insight gained from the previous research, this contribution investigates the potential of applying coordinate offset estimation to improve the overall performance of UWB end coordinates. Furthermore, this research aims to compare different ML models in terms of their prediction accuracy and processing delay. In addition to the training data from the first contribution, the models are augmented with training data from a supplementary real-life measurement campaign. Additionally, the estimates of these models are incorporated in an Adaptive Kalman Filtering (AKF) scheme as an input for measurement uncertainty. The solution was tested on a UWB measurement test dataset gathered at an industrial site, which was not used during the training. The overall results showed significant improvement in 2D and 3D positioning metrics using ML-augmented filtering when compared to non-ML-assisted filtering. On average, the end coordinates in the test set had approximately 10 cm smaller mean location error (MLE) and 40 cm smaller root mean square error (RMSE) in 2D positioning. In terms of 3D positioning, the MLE was reduced by approximately 20 cm and RMSE by 50 cm. Furthermore, the presence of outliers was reduced significantly as the maximum offset error decreased by several meters. Lastly, it is shown that ML-augmented filtering is sufficiently fast to be considered in a high update rate positioning system. The results showed that using the proposed residual features in an ML model provides a feasible approach to predict UWB positioning uncertainty and by employing it with a coordinate filtering scheme, the end coordinate can be considerably improved compared to non-ML-assisted filtering.

The final contribution investigates the possibilities of applying ML-based uncertainty estimates for a seamless indoor-outdoor sensor fusion solution using UWB and Global Navigation Satellite System (GNSS) sensors. While this contribution also applies the UWB XGBoost ML model from previous research, the main focus was to investigate the possible application of ML for GNSS-based positioning. It was motivated by the fact that usually, GNSS positioning integrity is described through Dilution of Precision (DoP), which reflects the level of geometrical uncertainty between a GNSS receiver and the serving satellites. However, the integrity of GNSS position also depends on other factors such as signal quality, pseudorange error, or the number of servicing satellites. It is proposed that an ensemble of GNSS features can be incorporated in an ML model for a more comprehensive and accurate uncertainty estimate. Using real-life GNSS measurement data gathered from areas with different degrees of positioning quality, a model is proposed to estimate the GNSS positioning performance. Both UWB and GNSS models are then applied in an adaptive coordinate filtering scheme with the predictions serving as input for individual sensor measurement uncertainty. The proposed solution was tested on a real-life testbed in indoor and outdoor areas. The results showed that the ML-augmented sensor fusion outperforms the usual method in relying only on DoP as a measure of positioning uncertainty with a mean positioning error of 0.16 m and a maximum error of approximately 0.5 m.

This thesis shows the possibilities of applying ML models to significantly improve the positioning performance of UWB and GNSS sensors. In summary, there is clear potential for further investigations of ML in positioning solutions as well as incorporating other sensors for ML-augmented multi-sensor schemes.

Kokkuvõte

Ülilairiba ja mitme sensoriga positsioneerimissüsteemide täpsuse parandamine masinõppe meetodil

Käesolev doktoritöö esitleb uuenduslikke meetodeid ülilairiba (*Ultra-Wideband, UWB*) ja globaalse satelliitnavigatsioonisüsteemi (*Global Navigation Satellite System, GNSS*) tehnoloogiatel põhinevate süsteemide positsioneerimistäpsuse parandamisel. Masinõppel (*Machine Learning, ML*) põhinevad mudelid rakendavad UWB ja GNSS sensoritest kogutud informatsiooni mõõtemääramatuse hindamiseks, mis on oluline komponent lõpp-koordinaadi filtreerimisel. Lõppkoordinaat arvutatakse mudeli prognoosi ja filtreerimise tulemusena, vähendades erindite (*outlier*) mõju ning parandades üldist koordinaadi täpsust ja punktipilve hajuvust.

Esmaalt pakub doktoritöö välja masinõppe mudeli UWB sensori koordinaadi täpsuse hindamisel. Tavaliselt hinnatakse UWB koordinaatide kvaliteeti raadiokanali statistika põhjal. Siiski on selle peamiseks puuduseks võrdlemisi pikk viiteaeg, et seda saaks rakendada kõrge sagedusega positsioneerimissammuga süsteemides. Alternatiivina pakub antud töö välja mudeli, mis on treenitud UWB kaugus- ja positsioneerimisinfo põhjal. Kasutades sisendinfo erinevaid statistilisi väärtusi masinõppe mudeli tunnusena, suudab arendatud mudel hinnata UWB koordinaadi ebatäpsust. Mudeli treeningandmed koguti reaalsest tööstuskeskkonnast, kus oli varasemalt ülesseatud UWB positsioneerimisvõrk. Treenitud mudelit testiti reaalse positsioneerimisinfo peal, eesmärgiga hinnata iga arvutatud UWB koordinaadi täpsust. Antud mudel suutis tuvastada koordinaadivigu vahemikus 0...0.2 m ligikaudu 90% täpsusega ja üldine täpsus kõigi teiste täpsusklasside tuvastamisel oli ligikaudu 84%. Need tulemused näitavad, et ML mudelit on võimalik edukalt rakendada UWB koordinaadi täpsuse hindamisel.

Järgmine töö toetub eelnevalt saavutatud tulemustele ja masinõppe potentsiaalsele rakendamisele UWB koordinaatide täpsuse hindamisel. Erinevalt eelnevast klassifitseerimismudelist, rakendatakse antud töös UWB kaugus- ja positsioneerimisinfot koordinaadi vea hindamiseks ning saadud väärtust kasutatakse adaptiivses Kalmani filtris (*Adaptive Kalman Filter, AKF*) mõõtemääramatuse hinnanguna. Lisaks võrreldakse erineva keerukusega masinõppe mudeleid nende täpsuse ja arvutuskiiruse osas. Masinõppe mudeleid täiustati lisa treeningandmetega ja kogu lahendust testiti eraldiseisvas UWB võrgus, mis oli samuti ülesseatud tööstusalal. Välja pakutud lahenduse tulemused näitasid märgatavat paranemist koordinaadi täpsuse ja hajuvuse osas. Võrreldes mitte-ML filtreerimisega vähenes üldine horisontaaltasapinna keskmine asukoha viga ligikaudu 10 cm ja punktipilve ruutkeskmine viga 40 cm võrra. Kolmemõõtmelise positsioneerimise puhul vähenesid samad statistilised väärtused ligikaudu 20 cm ja 50 cm võrra. Lisaks testiti antud töös mudeli prognoosi kiirust koos filtreerimisega. Saadud tulemuste põhjal võib väita, et välja pakutud lahendus on piisavalt väikese viitega, et rakendada seda tiheda positsioneerimissammuga süsteemides.

Viimasena uuritakse ka masinõppe rakendamist mitmiksensoritega, pakkudes välja UWB ja GNSS tehnoloogiatel põhineva ühendpositsioneerimissüsteemi. Antud lahendus võimaldab objekti asukoha määramist ühtse süsteemi abil nii siseruumides paikneva UWB võrgu abil, väliskeskkonnas GNSS sensoriga kui ka nõ. ülemineku alades, kus mõlema sensori kvaliteet on tavaliselt kompromiteeritud. Arvestades juba eelnevalt väljatöötatud UWB ML mudeliga, siis antud töös keskendutakse pigem GNSS mudelile. Sarnaselt UWB-ga, väljastab ka GNSS sensor infot, mida võib kaudselt seostada positsioneerimiskvaliteediga. Kui tavaliselt kasutatakse GNSS positsioneerimismääramatuse hindamise jaoks geomeetrilise määramatuse indikaatorit, siis võib väita, et tegemist ei ole piisavalt paindliku

hindamismöödikuga. Nimelt sõltub GNSS vastuvõtja positsioneerimistäpsus ka teistest teguritest nagu: teenindavate satelliitide arv, baasjaama korrektsiooni kvaliteet, pseudokauguste viga jpm. Kuna eelmainitud tunnused esinevad ka GNSS vastuvõtja poolt väljastatud sõnumites, siis rakendatakse neid tegureid ka väljapakutud masinõppe mudelis. Toetudes GNSS seadmetega tehtud mõõtmistel kogutud andmetele, loodi mudel mis võimaldab hinnata positsioneerimismääramatust nii hea kui ka halva leviga keskkondades. Nii UWB kui GNSS ML mudelid rakendati ühendsüsteemi loomisel ja adaptiivsel koordinaadi filtreerimisel AKF-ga, kus igal positsiooni uuendusel arvestatakse ML mudelite poolt prognoositud määramatusega. Võrreldes geomeetrilise määramatuse kasutamisega, näitas ML mudeliga täiendatud ühtse positsioneerimissüsteemi lahendus märgatavat paranemist nii koordinaadi keskmise vea (u. 0.16 m) ja maksimaalse vea osas (u. 0.5 m).

Antud doktoritöös uuriti võimalusi UWB ja GNSS tehnoloogial põhinevate positsioneerimissüsteemide positsioneerimistäpsuse parandamiseks erinevate masinõppe meetoditega. Teostatud uurimused ja katsete tulemused näitavad selget potentsiaali masinõppepõhisel lähenemisel, kus väljatöötatud mudelid suudavad määramatuse hindamisel arvestada mitmete positsioneerimistäpsust iseloomustavate teguritega. Edasised uuringud võiksid olla seotud ML mudelite täiustamisega ning ühendsüsteeme võiks täiendada ka teiste sensoritega (nt. inertsiaalandidurid), parandades seeläbi positsioneerimise stabiilsust ja täpsust.

Appendix 1

I

M. Tommingas, S. Ulp, M. M. Alam, I. Mürsepp, and T. Laadung, "Estimating UWB Positioning Integrity Based on Ranging Residuals," in *2023 24th International Conference on Applied Electromagnetics and Communications (ICECOM)*, pp. 1–5, IEEE, 2023

Estimating UWB Positioning Integrity Based on Ranging Residuals

Mihkel Tommingas^{1,2}, Sander Ulp², Muhammad Mahtab Alam¹, Ivo Mürsepp¹, Taavi Laadung^{1,2}

Thomas Johann Seebeck Department of Electronics

¹ *Tallinn University of Technology*

² *Eliko Tehnoloogia Arenduskeskus OÜ*

Tallinn, Estonia

{mihkel.tommingas, muhammad.alam, ivo.muursepp, taavi.laadung}@taltech.ee, sander.ulp@eliko.ee

Abstract—This paper investigates features based on Ultra-Wideband (UWB) ranging residuals used for coordinate integrity prediction. Usually, UWB-based positioning integrity is evaluated using channel statistics (CS). However, in practice this is not a feasible approach as gathering CS data takes too much time compared to the position update rate of an UWB system. In contrast to this approach, a set of features based on UWB ranging residuals are used in a machine learning (ML) algorithm to train a model for accurate integrity prediction. These features include those used in literature as well as proposed novel features. The trained model could predict measurements in the range of 0...0.2 m with the accuracy of 90% and having an overall accuracy of 84%. The results show that using the proposed residual features it is possible to predict UWB coordinate calculation integrity with high confidence. Lastly, the dataset used in this paper was collected during a measurement campaign in a complex industrial environment with constantly changing line-of-sight/non-line-of-sight (LOS/NLOS) conditions.

Index Terms—UWB, coordinate accuracy, integrity, ranging residuals, machine learning, XGBoost, feature importance

I. INTRODUCTION

A range-based Ultra-Wideband (UWB) positioning system consists of fixed beacons (anchors), which are used to measure distances to a mobile node (tag) using two-way-ranging (TWR) approach. Final position of the tag is estimated using all measured distances from anchors. UWB systems are generally considered robust and accurate by providing precise positioning in the presence of multipath effects and being less prone to interference [1]. On the other hand, the performance of UWB positioning depends on UWB anchor layout geometry, availability of anchors as well as absence of signal propagation impairments [2], [3]. Therefore, a typical industrial environment, with its constantly changing line-of-sight/non-line-of-sight (LOS/NLOS) conditions for radio wave

propagation, poses a challenging setting for UWB-based positioning.

Since positioning integrity is dependent on line-of-sight or non-line-of-sight conditions between the UWB anchor and tag, then majority of literature focuses on information gathered from channel statistics (CS). For example, channel impulse response (CIR) is a popular indicator used in NLOS detection and error mitigation schemes [4]–[6]. However, the gathering of CIR information requires a significant amount of data, hence originating a long latency of about one second just to transfer the measurements. This leads to an additional delay for estimating the range, making this approach unpractical when the estimated position needs to be updated at a high rate [7]. Additionally, it is noted that apart from long latency issues, the use of channel statistics is unfeasible in a constantly changing and harsh industrial environment [8]. A number of authors propose their positioning solutions in NLOS environments without the use of CS. For example, a factor graph-based positioning algorithm along with robust Tukey kernel was used to improve positioning accuracy [2]. Parameters like received signal strength (RSS) and ranging statistics have also been used for NLOS classification [7]. Other authors have tried to mitigate NLOS effect by applying residual weighting algorithms at a cost of higher computational complexity [9], [10].

In contrast to other works, this paper investigates position estimation integrity regardless of whether it is affected by LOS or NLOS conditions. Furthermore, the data was collected in a measurement campaign at an industrial site with dynamically changing obstacles, which affect radio frequency (RF) signal propagation and strength. Using a set of features on ranging residuals, a machine learning algorithm uses these features to predict the accuracy of the position estimation.

UWB ranging residuals can be evaluated by various characteristics such as standard deviation (SD) or root mean square error (RMSE). However, these measures alone may not suffice to describe whether a set of ranging residuals belongs to an accurate position estimation of an UWB tag. Therefore, this paper investigates ranging residuals and their potential characteristics in a more in-depth manner. In addition to the features presented in the literature, additional parameters such as those related to least squares (LS) and non-linear

This project has received funding from the European Union's Horizon 2020 Research programme under grant agreement No. 101058505 and in part by the Internet of Intelligent Things project of Estonian IT Academy program and Estonian Research Council under Grant PUT-PRG424. Additional support was received under the project of "Development of an industrial digital control system based on precise positioning technology ELIKO TAK and Atemix Automatika" nr. 2014-2020.4.02.21-0311.

least squares (NLS) optimization and geometrical dilution of precision.

Features that could potentially describe ranging residual behavior have not been thoroughly researched. For example, Silva and Hancke used sum of the squares of distance residuals (SSDR), mean and standard deviation and maximum distance as residual features for LOS/NLOS detection [8]. Additionally, Li and Wang used SSDR as a filter to discard invalid position estimates [2]. However, many other features could also be used for UWB positioning integrity i.e., to check whether a position estimation is reliable or not.

In this paper, XGBoost software library (*gbtree* booster) was used to train the model as well as to identify relevant features that contribute the most in making the prediction. This method has been implemented before to extract relevant UWB channel parameters and using this information to improve UWB-based positioning [11]. As previously stated, using CS is not in the scope of the current article but rather analyzing individual calculated ranges and position estimation. By using a large set of different features based on ranging residuals along with true distances, XGBoost was used to conduct supervised learning on a set of training data to predict the accuracy class of test set measurements. An additional goal was to train a general model that could also be used in future measurement campaigns on other sites.

The paper is organized as follows: Section II describes the theory of ranging residuals, how these are calculated and features that could potentially describe overall positioning integrity. Additionally, this section gives an overview of data collection and XGBoost algorithm. Section III shows the results in classification performance and feature importance. The article is concluded in Section IV.

II. ANALYSIS OF RANGING RESIDUALS AND DATA COLLECTION

A. Position estimation and residuals

For simplicity, following theory in this subsection is considering two-dimensional positioning. In the scope of this paper, position estimation of the tag is considered as a two-step process. In step 1 a set of circle equations (1) is established to solve the problem of multilateration:

$$(x_i - x)^2 + (y_i - y)^2 = d_i^2, \quad i = 1, 2, \dots, N, \quad (1)$$

where (x_i, y_i) is the known coordinate of i -th anchor and d_i is the true distance between tag and i -th anchor. Position of the tag (x, y) can be found by performing linearization on (1) and applying the LS method. Firstly, an anchor (x_r, y_r) with the shortest distance to the tag d_r is taken as a reference point [12]. Next, the non-linear expressions in all available circle equations are expanded as

$$x_i^2 - 2x_i x + x^2 + y_i^2 - 2y_i y + y^2 = d_i^2, \quad i = 1, 2, \dots, N \quad (2)$$

and the reference point (x_r, y_r) equation

$$x_r^2 - 2x_r x + x^2 + y_r^2 - 2y_r y + y^2 = d_r^2 \quad (3)$$

is subtracted from the rest of the expressions. The goal is to rearrange the terms with regards to unknowns x and y in a way that satisfies following linear model (4) as demonstrated by Guvenç, Chong and Watanabe [13]:

$$A\theta = b, \quad (4)$$

where

$$A = -2 \begin{bmatrix} x_1 - x_r & y_1 - y_r \\ x_2 - x_r & y_2 - y_r \\ \vdots & \vdots \\ x_{N-1} - x_r & y_{N-1} - y_r \end{bmatrix}, \quad (5)$$

$$\theta = \begin{bmatrix} x \\ y \end{bmatrix} \quad (6)$$

and

$$b = \begin{bmatrix} d_1^2 - d_r^2 - x_1^2 + x_r^2 - y_1^2 + y_r^2 \\ d_2^2 - d_r^2 - x_2^2 + x_r^2 - y_2^2 + y_r^2 \\ \vdots \\ d_{N-1}^2 - d_r^2 - x_{N-1}^2 + x_r^2 - y_{N-1}^2 + y_r^2 \end{bmatrix}. \quad (7)$$

Tag's position θ has the following LS solution:

$$\theta = (A^T A)^{-1} A^T b. \quad (8)$$

It should be considered that the anchors in Fig. 1 cannot be positioned in a straight line as this may result in a flip ambiguity [14]. Under ideal conditions, without any measurement errors, d_i are equal to true distances and the LS model provides a solution at the intersection of the three circles. However, in real-life applications, ranging measurements contain errors

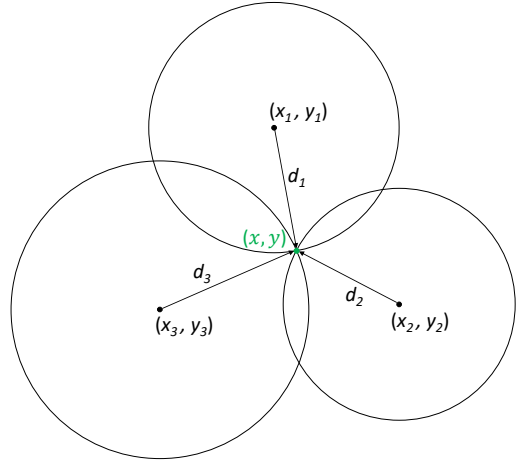


Fig. 1. Example of an error free 2-D trilateration scheme with three anchors using only x and y coordinates. By solving the ranging equations a unique solution is provided at the intersection of the three circles. Because of the absence of measurement noise, there are no ranging residuals.

caused by NLOS and ranging noise, thus producing varying position coordinates [2], [8].

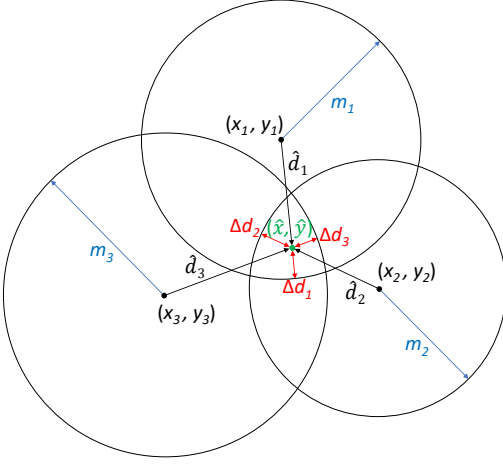


Fig. 2. Example of a 2-D trilateration scheme in UWB-based positioning with inaccurate range measurements. Difference between the position estimate and actual measured range results in a residual Δd_i that can be used in estimating positioning integrity.

For example, in Fig. 2 it can be seen that tags position is located somewhere in the vicinity of the three ranges and therefore cannot be pinpointed exactly. In this scenario, the tags location can be only estimated, thus the approximate position of the LS solution is determined somewhere in the area that is overlapped by three circles. Thus, the LS solutions could further be optimized using a Non-Linear Least Squares (NLS) model. As a step 2 of position estimation, Gauss-Newton optimization algorithm was applied using the initial guess provided by LS solution along with previously measured anchor coordinates and individual distances measured between anchors and tag. The estimated position is found by minimizing the objective function:

$$\hat{x}, \hat{y} = \underset{x, y}{\operatorname{argmin}} \sum_{i=1}^N ((x_i - x)^2 + (y_i - y)^2 - d_i^2)^2. \quad (9)$$

By comparing the individual measured anchor and tag distances m_i with the distances from estimated coordinate \hat{d}_i , the ranging residuals Δd_i are defined as:

$$\Delta d_i = m_i - \hat{d}_i. \quad (10)$$

It must be noted that in this work, functions (1) and (9) were augmented with the expression $(z_i - z)^2$ for position estimation in 3-D space.

B. Features of residuals

Position estimation can be used to indirectly reflect the quality of UWB observations as sum of ranging residuals

could be compared against a preset threshold to filter invalid positioning results [2]. However, this approach would lead to a situation where a single large residual among other small residuals would lead to a discarded position. Therefore, this paper proposes a set of 28 features to be used in a machine learning algorithm with a purpose of determining the most important features that could best classify accurate or inaccurate measurements. These features could be divided into 5 categories:

- **LS and NLS metrics.** These values are associated with position estimation as discussed in the previous chapter. Chosen parameters include: Euclidean distance between LS and NLS solutions, number of NLS iterations to convergence and NLS convergence tolerance¹;
- **Lengthened and shortened residuals.** As described by (10), a residual is the difference between distances of estimated coordinate and measured range from an anchor. Depending on the NLS solution, which considers all available ranges, the estimated position may appear closer or further away relative to the anchor, thus resulting in a lengthened or shortened residual. Therefore, following features could be established: number of lengthened/shortened residuals, sum of lengthened/shortened residuals, average of lengthened/shortened residuals, RMSE of lengthened/shortened residuals;
- **Residual statistics:** Following statistical features were included: variance, standard deviation, sum of squares (SSQ), sum of absolute values, mean, absolute mean and root mean square error (RMSE);
- **Number of residuals in range.** Small residuals indicate proximity to the NLS solution, whereas large residuals imply erroneous measurements. By counting the number of residuals in a preset range, it can be assumed whether the NLS algorithm uses accurate measurements as its input. Following ranges were chosen based on overall accuracy of UWB positioning [16], [17]: 0...0.1 m, 0.1...0.2 m, 0.2...0.4 m, 0.4...0.8 m, 0.8...1.6 m, 1.6...3.2 m, 3.2...6.4 m, 6.4...12.8 m, 12.8...25.6 m, 25.6... ∞ m;
- **Geometrical integrity of positioning.** Dilution of precision (DOP) indicates geometric location distribution in an indoor positioning system [18]. Using estimated coordinates of the tag, DOP parameter indirectly shows the level of geometrical uncertainty in an area relative to the anchors. In this article position dilution of precision (PDOP) was used as it depends on x , y and z coordinates.

C. Data collection

UWB measurement data was collected at an industrial site of Krah Pipes OÜ (Fig. 3), which manufactures thermoplastic pipes [19]. Regarding UWB ranging, the site presents a complex environment with constantly moving objects, which produce NLOS and multipath effects for RF signals. The

¹Convergence tolerance, is based on relative offset convergence criterion. This assures that the current parameter vector is less than 0.001% of the radius of the confidence region from the least squares point [15].

Eliko real-time locating system (RTLS) was set up inside the manufacturing facility by placing UWB anchors and a tag at preset locations. Based on DecaWave DW1000 chip, the RTLS was set to operate on UWB channel 4 [20]. Ground truth coordinates were measured in a local frame of reference with the Leica DISTO S910 measurement tool and assigned for 8 UWB anchors as well as 30 different tag locations around the facility. The measurement tool was positioned at a mezzanine floor in order to have LOS with all measurement points. By using an update rate of 10 Hz, each location was measured for 30 seconds, resulting in approximately 300 ranging sequences per location.



Fig. 3. Industrial site at Krah Pipes OÜ, which manufactures thermoplastic pipes in a complex industrial environment.

D. Data preparation and machine learning

All 30 measurement points from the measurement campaign were assembled into one dataset, which contained premeasured anchor coordinates, true coordinates of measurement points and individual distances measured by anchors. The task was to perform supervised machine learning i.e., classification on a data with preset accuracy classes:

- Class 1:** Distance between $0 \dots 0.2$ m;
- Class 2:** Distance between $0.2 \dots 0.4$ m;
- Class 3:** Distance between $0.4 \dots 0.8$ m;
- Class 4:** Distance between $0.8 \dots \infty$ m.

These categories were chosen based on UWB performance studies [16], [17]. In this paper, Class 1 presents positioning accuracy up to 0.2 m, which is also an approximate accuracy level for UWB-based positioning given in the literature. All other classes were set as a double value from the last step.

After dataset cleaning and shuffling, 80% of data was used for training, while 20% was used for testing purposes. XGBoost was selected as the ML algorithm of choice as it has been seen to dominate structured and tabular data sets

on classification, regression and predictive modeling problems [21]. XGBoost is a large-scale general-purpose gradient boosting library. Classification and regression tree (CART) is the basic component of the gradient boosted decision tree (GBDT) model. The final prediction results in summation of predictions of multiple regression trees. The XGBoost algorithm is composed of a series of base classifiers such as: decision tree, k-nearest neighbors (KNN), support vector machine (SVM) and logistic regression. These are linearly superimposed, so that they work together to optimize the algorithm [4].

R Studio package "XGBoost" with default settings (*gbtree* booster) was used to train the model and extract relevant features. Since the true coordinate in all 30 measurement points was known, then training data was labeled with aforementioned 4 categories using Euclidean distance between true coordinate and NLS estimation. The machine learning model was trained on a dataset of 28 features along with true distance values. As a result, the model can be used in classification of the test data. By using a relatively large set of measurements, an additional goal was to train a general model that could also be used on other sites in future measurement campaigns.

III. RESULTS

The XGBoost algorithm produced the predictive model and a list of features that contributed the most in terms of predicting positioning integrity i.e., the accuracy or inaccuracy of measurements. As can be seen on Fig. 4 the most important features out of the set of 28 seem to be related to lengthened residuals as these provide the biggest statistical gain. Less contribution is provided by dilution of precision, number of residuals in $0 \dots 0.1$ m range, LS/NLS distance, mean of ranging residuals and number of NLS optimization iterations.

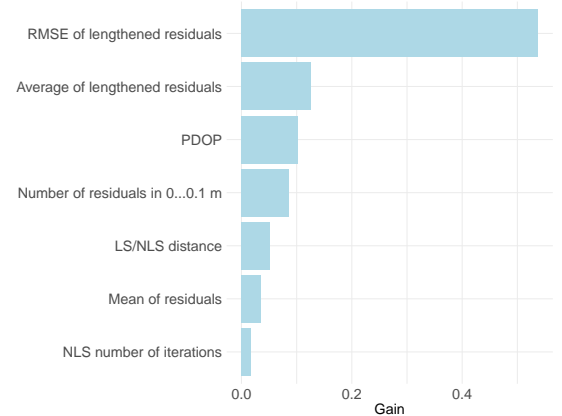


Fig. 4. Top 7 features that provide biggest contribution or gain as proposed by XGBoost algorithm. Gain represents the contribution of a feature in construction of the boosted decision trees within the model. When compared to other features, then higher gain implies bigger impact in prediction process. It can be seen that lengthened residuals are the most important features in describing estimated position integrity.

The lengthening of residuals means that the estimated position is further away from the individual measured distance from the anchor. The higher the RMSE of lengthened residuals, the bigger the offset from true coordinate.

Next, by using these features in a prediction model, classification could be performed on the test data. Performance can be seen in Table I using a confusion matrix and prediction statistics. Overall, prediction of distinct classes was performed with a high accuracy of ca. 84%. Class 2 was harder to predict because measurement points, which were estimated in the range of 0.2...0.4 m from true coordinate, were too close to Class 1 points to discern these with a high success rate.

TABLE I
CONFUSION MATRIX AND PREDICTION STATISTICS

		Reference Class			
		1	2	3	4
Predicted Class	1	1024	76	1	0
	2	109	240	14	0
	3	8	45	54	12
	4	0	5	17	143

Overall Accuracy: 0.8358 (ca. 84%)

Statistics by Class:

	Class 1	Class 2	Class 3	Class 4
Sensitivity	0.8975	0.6557	0.6279	0.9225
Specificity	0.8731	0.911	0.9608	0.9861

For example, there were 76 points that belonged to Class 2 and according to the model these points were labeled as Class 1. With a 90% sensitivity (true positive rate) the model could predict most of Class 1 measurements. Class 2 was harder to predict with a sensitivity of ca. 66%. However, if the measurement point did not belong to Class 1 or Class 2 then the prediction was made with a high specificity (true negative rate) of ca. 87% and 91% respectively. Sensitivity could potentially be improved with a broader Class 1 range (e.g., 0...0.3 m).

IV. CONCLUSION

This paper investigated if UWB-based ranging residuals could be used for estimating overall UWB positioning integrity. The residuals were examined through a set of features and comparing them with preset positioning accuracy classes. In addition to features already established in the literature, several other parameters were also included. By using a supervised machine learning algorithm (provided by XGBoost library) it was found that lengthened ranging residuals provide the biggest contribution and statistical gain in classification. With an overall accuracy of 84%, the trained model could successfully identify 90% of measurements in the range of 0...0.2 m. However, adjacent accuracy class, presenting distances of 0.2...0.4 m from true coordinate were identified with 66% accuracy. It can be concluded that the trained model could evaluate UWB positioning integrity with relatively high accuracy by using the input of lengthened residuals as its main indicator. Furthermore, this model could be applied in possible future measurement campaigns.

REFERENCES

- [1] S. Hayward, K. van Lopik, C. Hinde, and A. West, "A survey of indoor location technologies, techniques and applications in industry," *Internet of Things*, p. 100608, 2022.
- [2] X. Li and Y. Wang, "Research on a factor graph-based robust UWB positioning algorithm in NLOS environments," *Telecommunication Systems*, vol. 76, pp. 207–217, 2021.
- [3] I. Guvenc and C.-C. Chong, "A survey on TOA based wireless localization and NLOS mitigation techniques," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 3, pp. 107–124, 2009.
- [4] S. Li and X. Zhang, "Research on orthopedic auxiliary classification and prediction model based on XGBoost algorithm," *Neural Computing and Applications*, vol. 32, pp. 1971–1979, 2020.
- [5] J. Kulmer, S. Hinteregger, B. Grosswindhager, M. Rath, M. S. Bakr, E. Leitinger, and K. Witrals, "Using DecaWave UWB transceivers for high-accuracy multipath-assisted indoor positioning," in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2017, pp. 1239–1245.
- [6] D.-H. Kim, A. Farhad, and J.-Y. Pyun, "UWB positioning system based on LSTM classification with mitigated NLOS effects," *IEEE Internet of Things Journal*, 2022.
- [7] V. Barral, C. J. Escudero, and J. A. García-Naya, "NLOS classification based on RSS and ranging statistics obtained from low-cost UWB devices," in *2019 27th European Signal Processing Conference (EU-SIPCO)*. IEEE, 2019, pp. 1–5.
- [8] B. J. Silva and G. P. Hancke, "Non-line-of-sight identification without channel statistics," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2020, pp. 4489–4493.
- [9] P.-C. Chen, "A non-line-of-sight error mitigation algorithm in location estimation," in *WCNC. 1999 IEEE Wireless Communications and Networking Conference (Cat. No. 99TH8466)*, vol. 1. IEEE, 1999, pp. 316–320.
- [10] L. Jiao, J. Xing, and F. Y. Li, "Performance comparison of residual related algorithms for TOA positioning in wireless terrestrial and sensor networks," in *2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology*. IEEE, 2009, pp. 278–283.
- [11] Y. Liu, L. Liu, L. Yang, L. Hao, and Y. Bao, "Measuring distance using ultra-wideband radio technology enhanced by extreme gradient boosting decision tree (XGBoost)," *Automation in Construction*, vol. 126, p. 103678, 2021.
- [12] I. Guvenc, S. Gezici, F. Watanabe, and H. Inamura, "Enhancements to linear least squares localization through reference selection and ML estimation," in *2008 IEEE Wireless Communications and Networking Conference*. IEEE, 2008, pp. 284–289.
- [13] I. Guvenc, C.-C. Chong, and F. Watanabe, "NLOS identification and mitigation for UWB localization systems," in *2007 IEEE Wireless Communications and Networking Conference*. IEEE, 2007, pp. 1571–1576.
- [14] P. Moravek, D. Komosny, M. Simek, and J. Muller, "Multilateration and flip ambiguity mitigation in ad-hoc networks," *Przeglad Elektrotechniczny*, vol. 2012, no. 05b, pp. 222–229, 2012.
- [15] D. M. Bates and D. G. Watts, *Nonlinear regression analysis and its applications*. Wiley, 1988.
- [16] K. Mannay, N. Benhadjyoussef, M. Machhout, and J. Urena, "Location and positioning systems: Performance and comparison," in *2016 4th International Conference on Control Engineering & Information Technology (CEIT)*. IEEE, 2016, pp. 1–6.
- [17] W. Chantaweesomboon, C. Suwattikul, S. Manatrinon, K. Athikulwongse, K. Kaemarungsi, R. Ranron, and P. Suksompong, "On performance study of UWB real time locating system," in *2016 7th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES)*, 2016, pp. 19–24.
- [18] Y. Guo, W. Li, G. Yang, Z. Jiao, and J. Yan, "Combining Dilution of Precision and Kalman Filtering for UWB Positioning in a Narrow Space," *Remote Sensing*, vol. 14, no. 21, p. 5409, 2022.
- [19] "Welcome to Krah Pipes - Krah Pipes — krah-pipes.ee," <http://www.krah-pipes.ee/eng/>, 2023, [Accessed 15-Jun-2023].
- [20] Decawave Ltd, "DW1000 Datasheet," <https://www.qorvo.com/products/d/da007946>, 2023, [Accessed 15-Jun-2023].
- [21] S. Ramraj, N. Uzir, R. Sunil, and S. Banerjee, "Experimenting XGBoost algorithm for prediction and classification of different datasets," *International Journal of Control Theory and Applications*, vol. 9, no. 40, pp. 651–662, 2016.

Appendix 2

II

M. Tommingas, M. M. Alam, I. Mürsepp, and S. Ulp, "UWB Positioning Integrity Estimation Using Ranging Residuals and ML Augmented Filtering," *IEEE Journal of Indoor and Seamless Positioning and Navigation*, vol. 2, pp. 205–218, 2024

UWB Positioning Integrity Estimation Using Ranging Residuals and ML Augmented Filtering

Mihkel Tommingas, Muhammad Mahtab Alam, *Senior Member, IEEE*, Ivo Mürsepp, Sander Ulp

Abstract—This article investigates the use of Ultra-Wideband (UWB) ranging residuals for coordinate integrity estimation and their use in a filtering scheme. Typically, UWB system accuracy is improved using channel statistics to detect and mitigate non-line of sight (NLOS) effects between UWB sensors and the object to be located, potentially improving the end coordinate solution. However, in practice when considering UWB system with a high positioning update rate, this is not a feasible approach, as gathering and processing CS data takes too much time. In contrast to this approach, this article proposes a set of features based on UWB ranging residuals that could be used as an alternative in integrity assessment. By using machine learning (ML), the most important features were extracted from the initial set and then used to train and validate a model for UWB coordinate error prediction. Lastly, the prediction was applied in an adaptive Kalman filtering (AKF) scheme as an input for measurement uncertainty. Model testing was done using UWB measurement test dataset gathered at an industrial site. The overall results showed significant improvement in 2D and 3D positioning metrics of ML-augmented filtering when compared to non-ML filtering. On average, the end coordinates in the test set had ca. 10 cm smaller mean location error and ca. 40 cm smaller dispersion in 2D positioning. Additionally, the presence of outliers was reduced significantly as the maximum error offset decreased by several meters. Although ML augmented filtering is computationally slower than non-ML filtering (e.g., ordinary and extended Kalman filter), it is still faster than using channel statistics for UWB integrity estimation. The results show that using the proposed residual features in an ML model provides a feasible approach to predict UWB positioning integrity and use it as a measure of uncertainty in a coordinate filtering scheme.

Index Terms—end coordinate correction and filtering, machine learning (ML), ranging residuals, Ultra-Wideband (UWB) positioning.

This project has received funding from the European Union's Horizon 2020 Research programme under grant "5G-TIMBER" agreement No. 101058505, Just Transition Fund (JTF) project "Increasing the knowledge intensity of Ida-Viru entrepreneurship" (co-funded by the European Union) and by the Estonian Research Council under Grant PUT-PRG424. Additional support was received under the project of "Development of an industrial digital control system based on precise positioning technology ELIKO TAK and Atemix Automatika" nr. 2014-2020.4.02.21-0311.

Mihkel Tommingas, Muhammad Mahtab Alam and Ivo Mürsepp are with the Thomas Johann Seebeck Department of Electronics, Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia (e-mail: mihkel.tommingas@taltech.ee, muhammad.alam@taltech.ee, ivo.muursepp@taltech.ee).

Mihkel Tommingas and Sander Ulp are with the OÜ Eliko Tehnoloogia Arenduskeskus, Aiandi 13/1, 12918 Tallinn, Estonia (e-mail: mihkel.tommingas@eliko.ee, sander.ulp@eliko.ee).

I. INTRODUCTION

CREATING an Ultra-Wideband (UWB) positioning solution that provides reliable location information in a difficult industrial environment is a challenging task. For example, storage areas, whether indoor or outdoor, are usually littered with objects that obstruct radio frequency (RF) signal propagation, thus affecting the estimated coordinate of an object to be positioned. UWB-based systems are considered more robust in the presence of multipath effects and are less susceptible to interference as compared to other RF-signal-based positioning systems [1], [2], [3]. UWB system employs RF signals with a large bandwidth to be used in a wireless positioning scheme [4]. By using a two-way-ranging (TWR) approach, distances between fixed UWB nodes (anchors) and a mobile UWB node (tag) are measured and the final position of the tag is estimated based on these distances [2], [5]. Although a robust positioning solution, the ranging still relies on wireless RF signals. Thus, the accuracy and precision (i.e., integrity) of the end coordinate is affected but not limited by factors such as the number of servicing anchors, their vicinity to the tag, impairments caused by non-line of sight (NLOS) and suitable anchor layout geometry [1], [6], [7].

Problems related to NLOS detection and mitigation are an extensively researched topic in UWB-based positioning [5]. While the number of anchors and their spatial geometry can be adjusted according to the operating area, NLOS appears dynamically with constantly changing obstructions between the anchors and a moving tag. According to the works published by various authors, it can be seen that detection of NLOS and combating multipath effects is usually done by analyzing the characteristics of the RF propagation channel (i.e., channel state information (CSI)) [8], [9], [10], [11]. For example, channel impulse response (CIR), which describes the propagation path of a signal, can be used to assess the amplitude and phase of a particular multipath component [12]. Although this information is effective for NLOS detection, then CIR entails also some constraints. Certain authors have noted that gathering CIR information requires a significant amount of data, hence causing a latency of approximately one second just to transfer the measurements [13]. Taking into account UWB positioning solution with a high position update rate, the transfer and processing of CIR information becomes unpractical. Additionally, it is stated that CSI has to be collected for different types of environments, as a dataset describing a residential environment might not be suitable

for a harsh industrial environment [5]. Finally, considering machine learning (ML) based positioning algorithms that must be trained on real ranging or positioning data, using such an approach in conjunction with CSI increases computational complexity even further [2].

In essence, this article proposes positioning integrity assessment without the knowledge of CSI. In the literature, alternative methods have been used before. Barral et al. used received signal strength (RSS) and range information for ML-based LOS/NLOS detection and classification [13], [14]. Liu et al. investigated NLOS detection and mitigation using sum of squares (SSQ) of distance residuals. A large SSQ compared to a certain threshold would indicate inconsistency in localization [15]. Similarly, Silva and Hancke used SSQ of distance residuals for NLOS identification [5]. A residual test was proposed by Chan et al. in order to determine and identify the number of LOS base stations [16]. However, the current article expands on the analysis of ranging residuals further by adding features that describe their statistical and quantitative properties. Additionally, aspects related to end coordinate calculations and geometrical dilution of precision (DOP) were also included. The goal was to include characteristics that describe UWB positioning integrity whether affected by LOS/NLOS or varying anchor geometry. Therefore, this paper considers real-life measurement data that already contains both LOS and NLOS ranging measurements and it is assumed that end coordinate error is predicted regardless of tag's LOS/NLOS conditions or its position relative to anchors.

In contrast to most related works being done with simulated data, the current article considers real-life measurements gathered from three different indoor environments. The calculated features from raw ranging data were then used in three distinct ML algorithms: regression tree (RT), random forest (RF), and XGBoost (XGB) [17], [18], [19]. These methods were used to produce three different models, which could estimate the offset from the true coordinate. An additional objective was to evaluate whether there was any significant gain to be had from using a more complex machine-learning algorithm. Finally, the three different predictions were used as a measure of uncertainty in a coordinate filtering scheme in an Adaptive Kalman Filter (AKF), which was compared with the non-adaptive (KF) and extended Kalman filter (EKF).

The paper is organized as follows: Section II describes the theory behind end coordinate estimation and ranging residuals. Additionally, it is explained how residuals and their features are calculated. Section III gives an overview of data collection and processing with different ML algorithms along with coordinate filtering schemes. Section IV contains results by comparing the presented coordinate calculation methods. The article is concluded in Section V.

II. COORDINATE ESTIMATION METHODS AND FEATURES

A. End coordinate estimation

Estimating the coordinates of the tag with regard to surrounding anchors presents a problem of multilateration. In Fig. 1 it can be seen how an object on coordinates (\hat{x}, \hat{y}) is located at certain distances from all surrounding anchors

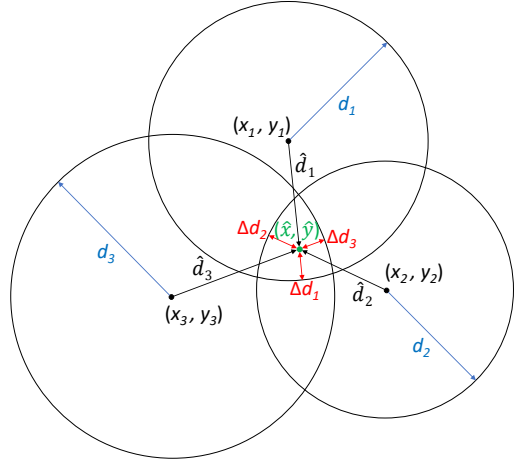


Fig. 1. Example of a 2-D trilateration scheme in UWB-based positioning with inaccurate range measurements. The difference between the distance to the estimated position \hat{d}_i and the actual measured range d_i results in a residual Δd_i that can be used in estimating positioning integrity.

denoted with (x_i, y_i) . By using known distance measurements d_i from each individual anchor, the tag's position can be estimated. Usually, for a single solution in 2D space, at least three-, and in 3D space four anchors are required [20]. It should also be considered that the anchors in Fig. 1 should not be positioned in a straight line as this may result in a flip ambiguity with possible solutions on either side of the line [21]. Under ideal conditions, without any measurement errors, $d_i = \hat{d}_i$ and the least squares (LS) model provides a solution at the intersection of the three circles [22]. However, in real-life applications, ranging measurements contain errors caused by NLOS propagation and ranging noise, thus producing varying position estimates [2], [5].

B. End coordinate calculation

In this article, the end coordinate calculation of the tag is considered as a two-step process. Firstly, estimating the initial position of the tag and then optimizing the solution with a non-linear least squares (NLS) approach. Both involve solving the multilateration problem using ranging measurements discussed in the previous section. Additionally, this article considers positioning in 3D space. In Step 1 a set of circle equations (1) is used to find the initial estimate of the tag's position $(\hat{x}, \hat{y}, \hat{z})$:

$$(x_i - \hat{x})^2 + (y_i - \hat{y})^2 + (z_i - \hat{z})^2 = d_i^2, \quad i = 1, 2, \dots, N, \quad (1)$$

where (x_i, y_i, z_i) is the known coordinate of the i -th anchor and d_i is the measured distance between the tag and the i -th anchor. The initial guess of the tag $(\hat{x}, \hat{y}, \hat{z})$ can be found by performing linearization on (1) and applying the LS method. Firstly, an anchor (x_r, y_r, z_r) with the shortest measured distance to the tag d_r is taken as a reference point

[23]. Next, the non-linear expressions in all available circle equations N are expanded as:

$$x_i^2 - 2x_i\hat{x} + \hat{x}^2 + y_i^2 - 2y_i\hat{y} + \hat{y}^2 + z_i^2 - 2z_i\hat{z} + \hat{z}^2 = d_i^2, \quad i = 1, 2, \dots, N \quad (2)$$

and the reference anchor (x_r, y_r, z_r) equation:

$$x_r^2 - 2x_r\hat{x} + \hat{x}^2 + y_r^2 - 2y_r\hat{y} + \hat{y}^2 + z_r^2 - 2z_r\hat{z} + \hat{z}^2 = d_r^2 \quad (3)$$

is subtracted from the rest of the expressions. The goal is to rearrange the terms with regards to unknowns \hat{x} , \hat{y} and \hat{z} in a way that satisfies the following linear model (4) as demonstrated by Guvenc, Chong, and Watanabe [24]:

$$A\theta = b, \quad (4)$$

where

$$A = -2 \begin{bmatrix} x_1 - x_r & y_1 - y_r & z_1 - z_r \\ x_2 - x_r & y_2 - y_r & z_2 - z_r \\ \vdots & \vdots & \vdots \\ x_{N-1} - x_r & y_{N-1} - y_r & z_{N-1} - z_r \end{bmatrix}, \quad (5)$$

$$\theta = \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} \quad (6)$$

and

$$b = \begin{bmatrix} d_1^2 - d_r^2 - x_1^2 + x_r^2 - y_1^2 + y_r^2 - z_1^2 + z_r^2 \\ d_2^2 - d_r^2 - x_2^2 + x_r^2 - y_2^2 + y_r^2 - z_2^2 + z_r^2 \\ \vdots \\ d_{N-1}^2 - d_r^2 - x_{N-1}^2 + x_r^2 - y_{N-1}^2 + y_r^2 - z_{N-1}^2 + z_r^2 \end{bmatrix}. \quad (7)$$

Lastly, the estimated tag's position θ has the following LS solution:

$$\theta = (A^T A)^{-1} A^T b. \quad (8)$$

As shown in an example in Fig. 1, it can be seen that the tag is estimated somewhere within the area overlapped by three circles. The sum of the squares of distance errors can further be minimized using the NLS approach [25]. As a step 2 of position estimation, the Gauss-Newton optimization algorithm was applied. The LS solution provides an initial estimate, along with previously measured anchor coordinates and individual distances measured between anchors and tag. The estimated position is found by minimizing the objective function:

$$\hat{x}, \hat{y}, \hat{z} = \argmin_{x, y, z} \sum_{i=1}^N ((x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2 - d_i^2)^2 \quad (9)$$

where x , y and z represent the coordinates that provide the smallest error. Since there are various methods to solve this non-linear multilateration problem, this article applies linearization using Taylor series with the Gauss-Newton iteration procedure. Renaming the initial guess from the LS solution (8) as (x_G, y_G, z_G) , the measured distances d_i are approximated

through first-order Taylor series expansion as demonstrated by Guillory, Truong, and Wallerand [20]:

$$\begin{aligned} d_i(\hat{x}, \hat{y}, \hat{z}) &\approx d_i(x_G, y_G, z_G) + \frac{\partial d_i(\hat{x}, \hat{y}, \hat{z})}{\partial \hat{x}} \Big|_{x_G, y_G, z_G} \Delta x + \\ &+ \frac{\partial d_i(\hat{x}, \hat{y}, \hat{z})}{\partial \hat{y}} \Big|_{x_G, y_G, z_G} \Delta y + \frac{\partial d_i(\hat{x}, \hat{y}, \hat{z})}{\partial \hat{z}} \Big|_{x_G, y_G, z_G} \Delta z \\ &\approx d_i(x_G, y_G, z_G) + \\ &+ \frac{\hat{x} - x_i}{\sqrt{(x_i - \hat{x})^2 + (y_i - \hat{y})^2 + (z_i - \hat{z})^2}} \Big|_{x_G, y_G, z_G} \Delta x + \\ &+ \frac{\hat{y} - y_i}{\sqrt{(x_i - \hat{x})^2 + (y_i - \hat{y})^2 + (z_i - \hat{z})^2}} \Big|_{x_G, y_G, z_G} \Delta y + \\ &+ \frac{\hat{z} - z_i}{\sqrt{(x_i - \hat{x})^2 + (y_i - \hat{y})^2 + (z_i - \hat{z})^2}} \Big|_{x_G, y_G, z_G} \Delta z \\ &\approx d_i(x_G, y_G, z_G) + \frac{\hat{x} - x_i}{d_i(\hat{x}, \hat{y}, \hat{z})} \Big|_{x_G, y_G, z_G} \Delta x + \\ &+ \frac{\hat{y} - y_i}{d_i(\hat{x}, \hat{y}, \hat{z})} \Big|_{x_G, y_G, z_G} \Delta y + \frac{\hat{z} - z_i}{d_i(\hat{x}, \hat{y}, \hat{z})} \Big|_{x_G, y_G, z_G} \Delta z \\ &\approx d_i(x_G, y_G, z_G) + \frac{x_G - x_i}{d_i(x_G, y_G, z_G)} \Delta x + \\ &+ \frac{y_G - y_i}{d_i(x_G, y_G, z_G)} \Delta y + \frac{z_G - z_i}{d_i(x_G, y_G, z_G)} \Delta z, \end{aligned} \quad (10)$$

where Δx , Δy and Δz are equal to $\hat{x} - x_G$, $\hat{y} - y_G$ and $\hat{z} - z_G$ respectively. Considering that Δx , Δy and Δz are multiplied to first-order derivatives when:

$$J_i = \begin{bmatrix} \frac{x_G - x_i}{d_i(x_G, y_G, z_G)} & \frac{y_G - y_i}{d_i(x_G, y_G, z_G)} & \frac{z_G - z_i}{d_i(x_G, y_G, z_G)} \end{bmatrix}, \quad (11)$$

then (10) can be rearranged into matrix form:

$$\Delta d_{NLS} = J \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}, \quad (12)$$

with Δd_{NLS} representing the difference between measured and estimated distances. The error corrections Δx , Δy and Δz can be found by solving the Normal Equation as shown in (8) and substituting values accordingly:

$$\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = (J^T J)^{-1} J^T \Delta d_{NLS}. \quad (13)$$

Using the error correction vector, the initial guess coordinates x_G , y_G and z_G are updated with Gauss-Newton iteration until a convergence criterion has been reached (e.g., until the error correction vector is sufficiently small [20]). After reaching a pre-determined threshold, the final position estimation results as:

$$\begin{bmatrix} x_G + \Delta x \\ y_G + \Delta y \\ z_G + \Delta z \end{bmatrix} = \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix}. \quad (14)$$

Similarly to Fig. 1, it can be seen that the difference between an individual measured distance d_i , and distance \hat{d}_i calculated from the estimated coordinate $(\hat{x}, \hat{y}, \hat{z})$, results in a residual Δd_i as:

$$\Delta d_i = d_i - \hat{d}_i. \quad (15)$$

C. Features

This paragraph describes features used in the ML model training. As stated in the introduction, this article considers both previously used features in the literature as well as several novel ones. As ranging residuals could indirectly reflect the end coordinate integrity, several statistical metrics such as the residual mean or sample variance have been added. Additional features have also been included regarding end coordinate calculations and positioning geometry.

1) *Residual statistics*: Depending on the location of the estimated solution, \hat{d}_i may be longer or shorter compared to the individual measured distance d_i resulting in a positive or negative residual. A significant change in the magnitude of a residual may indicate that UWB propagation path is affected by an obstruction. Therefore, residual statistics were calculated for three different sets: positive, negative and overall residuals. Additionally, statistical equations were averaged to remove the dependence on the size of available residuals. The following statistics were calculated:

average sum of squares (SSQ)

$$SSQ = \frac{\sum_{i=1}^n \Delta d_i^2}{n}, \quad (16)$$

root mean square (RMS)

$$RMS = \sqrt{\frac{SSQ}{n}}, \quad (17)$$

mean

$$\bar{x} = \frac{\sum_{i=1}^n \Delta d_i}{n}, \quad (18)$$

mean absolute deviation (MAD)

$$MAD = \frac{\sum_{i=1}^n |\Delta d_i - \bar{x}|}{n}, \quad (19)$$

standard deviation

$$s = \sqrt{\frac{\sum_{i=1}^n (\Delta d_i - \bar{x})^2}{n}}, \quad (20)$$

and variance

$$v = s^2, \quad (21)$$

where n represents the number of residuals in a corresponding positive, negative or overall set (also used as a feature).

2) *LS and NLS metrics*: These values are associated with position calculation as discussed in chapter II-B. The chosen parameters include Euclidean distance ΔD between LS (8) and NLS (14) solutions and the number of Gauss-Newton iterations to convergence n_{GN} . For the latter, there is no implicit equation as the iteration counter is initialized at each coordinate optimization process.

$$\Delta D = \sqrt{(x_G - \hat{x})^2 + (y_G - \hat{y})^2 + (z_G - \hat{z})^2} \quad (22)$$

3) *Geometrical integrity of positioning*: In an indoor positioning system, Dilution Of Precision (DOP) indicates geometric location distribution. It contains the knowledge of positioning accuracy under specific base station network and scene characteristics [6]. Using the estimated coordinates of the tag, the DOP parameter indirectly shows the level of geometrical uncertainty in an area relative to the anchors. In this article, the position dilution of precision (PDOP) was used as it depends on x , y , and z coordinates. To calculate PDOP, the set of ranging equations (1) can be implemented with precalculated end coordinates from (14). By finding partial derivatives with respect to each coordinate similarly as was shown in (10), the result is formulated in matrix form as:

$$A_p = \begin{bmatrix} \frac{x_1 - \hat{x}}{d_1} & \frac{y_1 - \hat{y}}{d_1} & \frac{z_1 - \hat{z}}{d_1} & 1 \\ \frac{x_2 - \hat{x}}{d_2} & \frac{y_2 - \hat{y}}{d_2} & \frac{z_2 - \hat{z}}{d_2} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{x_N - \hat{x}}{d_N} & \frac{y_N - \hat{y}}{d_N} & \frac{z_N - \hat{z}}{d_N} & 1 \end{bmatrix}. \quad (23)$$

Next, the covariance matrix Q is calculated from the LS normal matrix:

$$Q = (A_p^T A_p)^{-1} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_y^2 & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_z^2 \end{bmatrix}. \quad (24)$$

Lastly, PDOP is calculated from trace of matrix Q as:

$$PDOP = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}, \quad (25)$$

III. DATA PROCESSING, MODEL TRAINING AND FILTERING

A. Data collection

UWB measurement data were collected at three different sites: Krah Pipes OÜ factory, Eliko office, and Auroom Kastre factory, which all contained a set network of UWB sensors. Data from the first two sites were used to cross-validate and train the ML model, while data from the third site were used for testing. Krah Pipes OÜ (Fig. 2) is a company that manufactures thermoplastic pipes and in terms of RF propagation, presents a complex environment with constantly moving objects [26]. The Eliko real-time locating system (RTLS) was installed inside the manufacturing facility by placing UWB anchors at fixed locations while the tag was sequentially placed at different locations on the factory floor during the measurements. Based on the Qorvo's DW1000 chip, the RTLS was set to operate on UWB channel 4 [27]. Eliko RTLS also uses Active-Passive TWR protocol with clock offset error mitigation [28]. Ground truth coordinates were measured in a local frame of reference with the Leica DISTO S910 measurement tool and assigned to 8 UWB anchors, as well as 30 different tag locations around the facility. The measurement tool has an accuracy of ± 1 mm [29].

The measurement tool was positioned on a mezzanine floor in order to have LOS with all measurement points. By using an update rate of 10 Hz, each location was measured for 30 seconds, resulting in approximately 300 ranging sequences per location. A similar measurement procedure was performed



Fig. 2. Industrial site at Krah Pipes OÜ, which manufactures thermo-plastic pipes in a complex industrial environment. The site contained a network of 8 UWB anchors (layout in Fig. 19).

at the Eliko office (30 measurement points) and Auroom Kastre factory (40 measurement points) using UWB positioning network of 17 and 15 anchors respectively. The office environment provided additional training data in terms of poor PDOP conditions i.e., measurements that were taken outside of the convex hull of the UWB anchor layout as can be seen in Fig. 18 (e.g., points 13, 14 and 15).

B. Data processing and model training

Raw ranging data, collected during the measurements, was assembled into training-validation (Krah factory and Eliko office) and testing (Auroom factory) datasets containing end coordinates, true distances (dependent) and features (independent) described in Section II-C. It should be noted that data from the Auroom factory were not used in training in order to have a stand-alone dataset to test the general model. The purpose of the model was to predict end-coordinate error or offset based on pre-calculated independent features. After data cleaning and shuffling, the datasets were changed into the appropriate format for cross-validation and training. Three ML methods were chosen: extreme gradient boosting, regression tree, and random forest. The idea was to compare the prediction performance of a simple ML method (i.e., a single regression tree) against more complex ones.

The aforementioned ML algorithms and datasets were used in the R Studio environment [30]. For each ML method, 10-fold data cross-validation was carried out to select hyper-parameters that provide the smallest prediction error against the validation set. Essentially, the training dataset was separated into 10 segments with 1 segment being the validation set. Such an approach helps to generalize the model and mitigate overfitting. Next, using chosen hyper-parameters in an initial model, combination of most important features were selected for the final model.

R Studio provides appropriate cross-validation *train* and *trainControl* functions through the *caret* library [31]. The main hyper-parameters used were: tree depth and number of

boosting iterations for XGBoost; tree depth and complexity parameter for regression tree and tree depth for random forest.

It should be noted that no prior feature selection before model cross-validation was done. Rather regression tree, random forest, and XGBoost libraries in R Studio already contain built-in functions to output features that contribute the most in making the prediction.

1) *Regression tree feature selection and training*: In contrast to using a decision tree for classification task, end-coordinate offset is considered as a continuous target variable that is predicted using a regression tree. It is generated using a set of training samples with the corresponding response variables. A trained tree structure is then used to predict the value of an unknown test sample. It consists of root, branches, nodes, and leaves. Each internal node represents a feature, branches represent the feature values and leaf nodes represent the outcome of prediction [32].

Cross-validation compared sets of training data using different regression tree depths in terms of prediction error. As can be seen in Fig. 3, a tree with a depth of 7 is sufficient for providing the least amount of error as choosing a deeper tree results in no further error mitigation. Additionally, the tree can be pruned or optimized using a complexity parameter (CP), which is the minimum improvement in the model needed at each node. CP is used to select the optimal size for the tree. As can be seen in Fig. 4, a tree size of 7 has a complexity parameter of 0.025, which outputs a tree shown in Fig. 5.

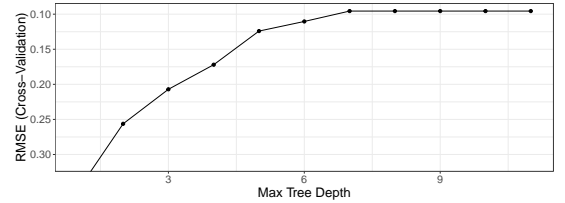


Fig. 3. Determining regression tree depth after 10-fold cross-validation. It can be seen that a tree depth of 7 is enough to provide the smallest root mean square error (RMSE).

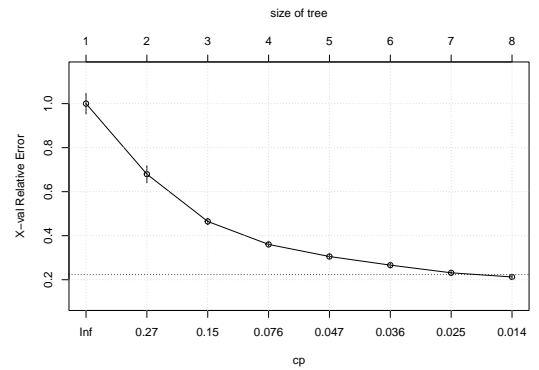


Fig. 4. After 10-fold cross-validation, a tree depth of 7 corresponds to a complexity parameter of 0.025. These hyperparameters were used to generate the final regression tree (shown in Fig. 5).

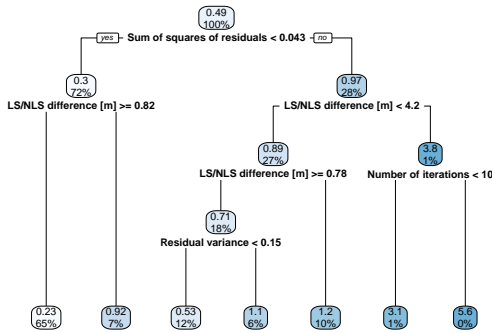


Fig. 5. Final regression tree to be used in ML prediction.

It should be noted that the *rpart* library also removes surrogate features i.e., features that provide the same goodness of split. Therefore, the final tree may have a different depth compared to cross-validated trees. Features were extracted by using a built-in *rpart.plot* function and feature importance was based on the goodness of split [17]. Features for the final model can be seen in Fig. 6.

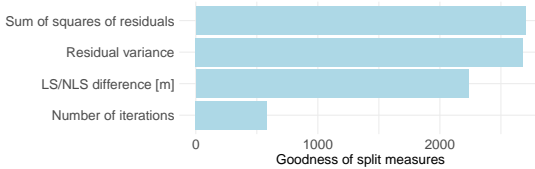


Fig. 6. Features used in the final regression tree model, which are ordered based on the goodness of split in a regression tree.

2) Random forest feature selection and training: In ensemble learning, bagging and boosting are two main approaches. Random forest can be viewed as an evolution of bagging methodology and can be used in classification and regression problems. It is defined as an ensemble of decision trees that implements randomness in the model-building process of each decision tree [33]. It can process high-dimensional data effectively, so it is different from neural networks (NN). In RF, each tree acts as an independent regression function, and regression trees are trained using different bootstrap samples of the training data. The average prediction of each individual tree is used as the final output [34].

Random forest training, validation and testing were done using the *ranger* package, which is a fast implementation of random forest suited for high-dimensional data [18]. Cross-validation on training data showed how different number of random forest predictors compare in terms of prediction error. As shown in Fig. 7, using 100 random trees with 8 predictors provides a sufficient amount of error as using more than eight might lead to model overfitting and results in no significant reduction in RMSE. Next, feature selection was done for the initial RF model, with 8 random predictors. By comparing different combinations of features, those with the least amount

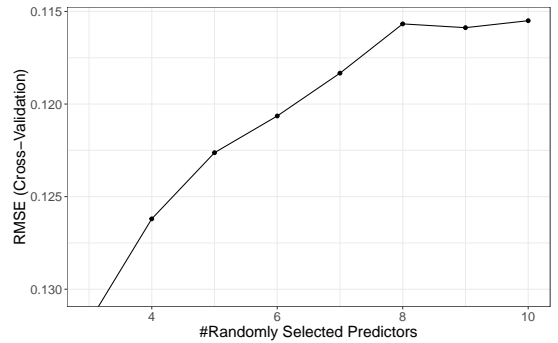


Fig. 7. Finding the optimal number of randomly selected RF predictors using cross-validation with 100 random trees. It can be seen that using more than 8 features results in no significant increase in cross-validation error.

of error in predicting validation set response values were selected. As can be seen in Fig. 8, choosing more than 7 features results in no significant decrease in prediction error. These features are presented in Fig. 9.

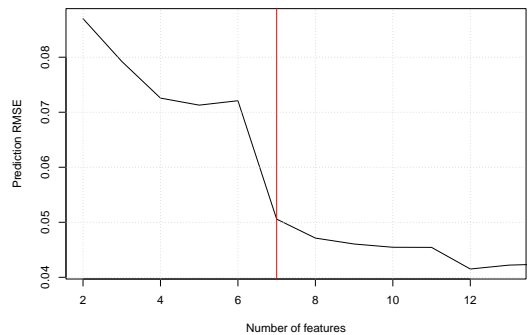


Fig. 8. Using more than seven most important features results in no significant RMSE decrease in random forest prediction and could overfit the model.

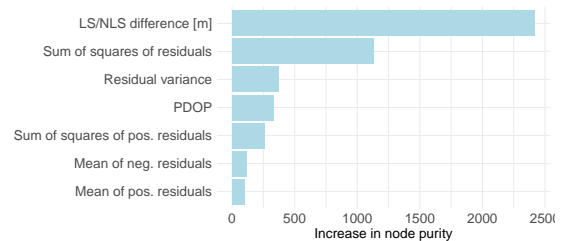


Fig. 9. Set of features in the final model that provide the biggest node purity increase in random forest prediction.

3) *XGBoost feature selection and training*: On the other hand, XGBoost represents the approach of boosted ensemble learning. It is a large-scale general-purpose gradient boosting library, which has been seen to dominate structured and tabular data sets on classification, regression, and predictive modeling problems [19], [35]. The algorithm creates a sequential ensemble of tree models, all of which work to improve each other. The final prediction results in a summation of the predictions of multiple regression trees. The XGBoost algorithm comprises a series of base classifiers such as decision tree, k-nearest neighbors, support vector machine, and logistic regression. These are linearly superimposed, so that they work together to optimize the algorithm [36].

Using cross-validation with *xgboost* library, different sets of XGBoost parameters were compared in terms of prediction error as shown in Fig. 10. It can be seen that a model with a tree depth of 5 and 150 boosting iterations is sufficient as choosing more than 150 iterations would present no significant increase in prediction performance. Additionally, feature selection was done using the initial model with aforementioned hyper-parameters. By comparing different combinations of features, those with the least amount of error in predicting validation set response values were selected. As can be seen in Fig. 11, more than 8 features provide only a marginal increase in predicting validation set response values. List of features used in the final XGBoost model is presented in Fig. 12.

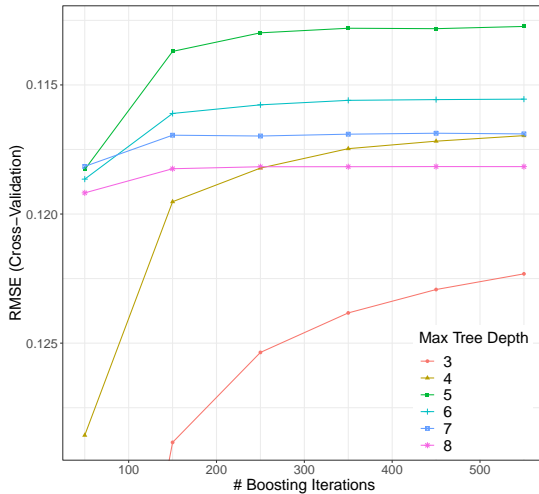


Fig. 10. Determining suitable XGBoost tree depth and number of boosting iterations using cross-validation. Tree depth 5 and 150 boosting iterations are chosen parameters for the model. Choosing a higher number of iterations results in no significant decrease in RMSE and might lead to overfitting.

C. Coordinate filtering

As a final step, the end coordinate is estimated using a Kalman filter. In the current context, the filter averages end coordinates, while considering the uncertainty of measurements (prediction) and previously filtered coordinates. While in a

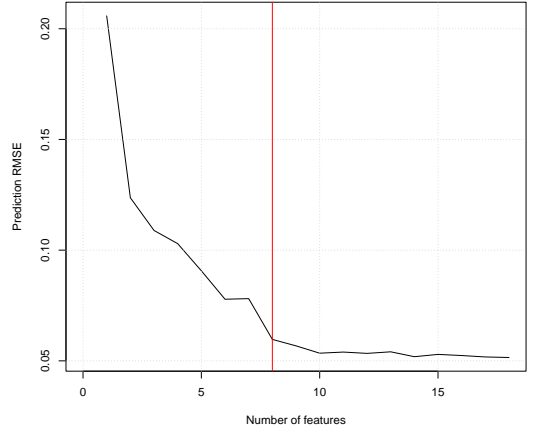


Fig. 11. Using more than 8 features has no significant impact on XGB prediction accuracy.

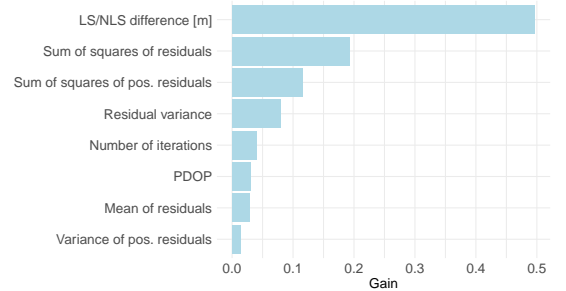


Fig. 12. Features used in the final XGBoost model.

traditional Kalman Filter (KF) the process and measurement noise have fixed values, then in real-life applications it can be seen that measurement uncertainty is a dynamic value, which in turn is affected by external factors such as NLOS. Therefore, it is preferable to know the measurement uncertainty at every ranging calculation in order to estimate whether the current coordinate can be trusted or not.

In this article, the ML model predicts end coordinate offset from true value, based on features used in the ML model. Since, the direction of the error with regards to x , y , and z axes is not known, this prediction can be considered as a measure of uncertainty in all three axes. By implementing the prediction as a dynamic measurement uncertainty in an Adaptive Kalman Filter (AKF), positioning accuracy can be improved further.

1) *KF and AKF filtering*: In this work, the main difference between a KF and AKF is in the application of the R matrix, which represents positioning measurement uncertainty. In KF, the diagonal elements of R in (26) were chosen as fixed values $\text{diag}(0.01, 0.01, 0.01)$ corresponding to the precision of the DW1000 device [37] with:

$$R_{KF} = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix} = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}. \quad (26)$$

However, AKF measurement uncertainty in (27) is updated at each iteration as the end coordinate is calculated and ML prediction is added to the variance of diagonal elements as

$$R_{AKF} = \begin{bmatrix} 0.01 + \hat{D}_{ML} & 0 & 0 \\ 0 & 0.01 + \hat{D}_{ML} & 0 \\ 0 & 0 & 0.01 + \hat{D}_{ML} \end{bmatrix}. \quad (27)$$

In essence, the ML prediction drives the filtering process by dynamically changing measurement uncertainty i.e., whether to trust measurement or process. In KF, EKF and AKF, the process noise matrix Q has constant values $\text{diag}(0.01, 0.01, 0.01)$. As shown at the beginning of Alg. 1, the state transition matrix A , state covariance P_0 and observation matrix H are initialized as 3-by-3 identity matrices. \hat{X}_0 represents the first converged NLS solution from (14), Z_k is the measurement vector and I is a 3-by-3 identity matrix.

2) EKF filtering: Lastly, ML-driven AKF is compared with the Extended Kalman Filter (EKF), which is capable of dealing with non-linear problems such as multilateration described in II-A. In contrast to KF and AKF, which predict and correct coordinates, EKF makes state corrections using residuals between measured distances Z_k and distances to the last estimated coordinates. In Alg. 1 state correction step $H_k \hat{X}_k^-$ is replaced with D_k^- where:

$$D_k^- = \begin{bmatrix} \sqrt{(x_k^- - x_1)^2 + (y_k^- - y_1)^2 + (z_k^- - z_1)^2} \\ \sqrt{(x_k^- - x_2)^2 + (y_k^- - y_2)^2 + (z_k^- - z_2)^2} \\ \vdots \\ \sqrt{(x_k^- - x_n)^2 + (y_k^- - y_n)^2 + (z_k^- - z_n)^2} \end{bmatrix} \quad (28)$$

with x_k^- , y_k^- and z_k^- representing coordinates from last iteration. Measurement vector Z_k represents current iteration distance equations with added measurement noise

$$Z_k = \begin{bmatrix} \sqrt{(x_k - x_1)^2 + (y_k - y_1)^2 + (z_k - z_1)^2 + v_1} \\ \sqrt{(x_k - x_2)^2 + (y_k - y_2)^2 + (z_k - z_2)^2 + v_2} \\ \vdots \\ \sqrt{(x_k - x_n)^2 + (y_k - y_n)^2 + (z_k - z_n)^2 + v_n} \end{bmatrix} \quad (29)$$

where v_k represents measurement noise vector, which has covariance matrix R_k as $\text{diag}(0.01, 0.01, 0.01)$. Process noise matrix Q is also set as $\text{diag}(0.01, 0.01, 0.01)$.

With EKF, the entire NLS approximation process discussed in II-B may be bypassed and do linearization through the observation matrix H , which is comprised of first-order partial derivatives as demonstrated by Kim, Farhad and Pyun [37]:

$$H_k = \begin{bmatrix} \frac{\partial d_1(x_k, y_k, z_k)}{\partial x_k} & \frac{\partial d_1(x_k, y_k, z_k)}{\partial y_k} & \frac{\partial d_1(x_k, y_k, z_k)}{\partial z_k} \\ \frac{\partial d_2(x_k, y_k, z_k)}{\partial x_k} & \frac{\partial d_2(x_k, y_k, z_k)}{\partial y_k} & \frac{\partial d_2(x_k, y_k, z_k)}{\partial z_k} \\ \vdots & \vdots & \vdots \\ \frac{\partial d_n(x_k, y_k, z_k)}{\partial x_k} & \frac{\partial d_n(x_k, y_k, z_k)}{\partial y_k} & \frac{\partial d_n(x_k, y_k, z_k)}{\partial z_k} \end{bmatrix}, \quad (30)$$

where derivatives correspond to

$$\frac{\partial d_i(x_k, y_k, z_k)}{\partial x_k} = \frac{x_k - x_i}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2 + (z_k - z_i)^2}} \quad (31)$$

$$\frac{\partial d_i(x_k, y_k, z_k)}{\partial y_k} = \frac{y_k - y_i}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2 + (z_k - z_i)^2}} \quad (32)$$

$$\frac{\partial d_i(x_k, y_k, z_k)}{\partial z_k} = \frac{z_k - z_i}{\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2 + (z_k - z_i)^2}}. \quad (33)$$

In the context of coordinate calculation, skipping the NLS coordinate calculations makes EKF computationally less demanding. On the other hand, a poor LS coordinate in the state vector can affect the filtering process and result in an inaccurate coordinate. Therefore, in this work for comparison purposes, EKF was provided with a converged NLS coordinate as the initial state vector.

Algorithm 1 Kalman Filter

Input: $\hat{X}_0, Z_k, P_0, Q, R$

Output: \hat{X}_k

Initialize A, P_0, H, I

Prediction step

for $k = 1, \dots, \infty$

1: State prediction $\hat{X}_k^- = A \hat{X}_{k-1}$

2: Covariance prediction $P_k^- = A P_{k-1} A^T + Q$

Correction step

3: Kalman gain $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$

4: State correction $\hat{X}_k = \hat{X}_k^- + K_k (Z_k - H_k \hat{X}_k^-)$

5: Covariance correction $P_k = P_k^- (I - K_k H_k)$

return \hat{X}_k, P_k

end for

IV. RESULTS

Test data was measured in an industrial site at Auroom Kastre factory, which manufactures sauna modules as shown on Fig. 13. The measurement setup was similar to the Eliko office and Krah Pipes factory with 40 different measurement points scattered over the factory area as can be seen in Fig. 17.

True coordinates were measured with the Disto S910 measurement device and ranging data was collected using UWB tag with a 10 Hz update rate. The goal was to test the performance of different end coordinate calculation methods, specifically comparing regular filtering methods to those augmented with ML prediction. Additionally, no data gathered



Fig. 13. Manufacturing area inside Auroom Kastre factory. Red circles highlight visible UWB anchors. Anchor layout can be seen on Fig. 17.

from the test site was included in ML model training to have unbiased verification of the model.

Considering true coordinates (x_T, y_T, z_T) , the following metrics were used to evaluate positioning accuracy and precision: mean location error (MLE), root mean square error (RMSE), distance root mean square error (DRMS), mean radial spherical error (MRSE) and maximum error [38], [39]:

1) 2D metrics:

$$MLE_{2D} = \frac{\sum_{i=1}^n \sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2}}{n}, \quad (34)$$

$$RMSE_{2D} = \sqrt{\frac{\sum_{i=1}^n [(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2]}{n}}, \quad (35)$$

$$DRMS = \sqrt{\sigma_x^2 + \sigma_y^2}, \quad (36)$$

$$MAX_{2D} = \max_{i \in n} (\sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2}). \quad (37)$$

2) 3D metrics:

$$MLE_{3D} = \frac{\sum_{i=1}^n \sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2 + (z_T - \hat{z}_i)^2}}{n}, \quad (38)$$

$$RMSE_{3D} = \sqrt{\frac{\sum_{i=1}^n [(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2 + (z_T - \hat{z}_i)^2]}{n}}, \quad (39)$$

$$MRSE = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}, \quad (40)$$

$$MAX_{3D} = \max_{i \in n} (\sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2 + (z_T - \hat{z}_i)^2}). \quad (41)$$

Overall statistics summarizing all 40 measurement points can be seen in Tables I and II.

In general, it was challenging to achieve good vertical precision and accuracy in most of the measurement locations. This can be attributed to UWB anchor layout geometry, with anchors located approximately on the same height level, resulting in a poor dilution of precision. Additional difficulties arose from occasional NLOS conditions between anchors and

the tag. However, as shown in Table II, the overall vertical position error of approximately 0.5 m is at a similar level as in a previously published work by Laadung *et al.* [40].

TABLE I
OVERALL METRICS FOR 2D POSITIONING.

	MLE 2D [m]	RMSE 2D [m]	DRMS [m]	Max. error 2D [m]
NLS	0.46	0.95	0.85	11.16
KF	0.43	0.72	0.57	7.01
AKF + XGB	0.28	0.29	0.11	0.62
AKF + RF	0.28	0.29	0.1	0.55
AKF + RT	0.27	0.28	0.11	0.63
EKF	0.62	0.96	0.78	6.28

TABLE II
OVERALL METRICS FOR 3D POSITIONING.

	MLE 3D [m]	RMSE 3D [m]	MRSE [m]	Max. error 3D [m]
NLS	0.8	1.36	1.17	14.04
KF	0.74	1.05	0.8	8.78
AKF + XGB	0.48	0.5	0.18	0.94
AKF + RF	0.48	0.5	0.18	0.9
AKF + RT	0.51	0.53	0.2	1.07
EKF	2.86	3.26	1.94	11.73

An example of superimposed end coordinate results can be seen in Fig. 14 and Fig. 15 along with respective performance metrics in Tables III and IV. The general location of the point can be seen on overall the map in Fig. 17. It can be seen both visually and statistically that EKF had the worst performance, especially in 3D positioning. With many visible outliers, EKF relies on coordinates calculated straight from noisy ranging data. Furthermore, EKF does not have any convergence process (i.e., Gauss-Newton iterations), thus relying only on the first calculated end coordinate solution. On the other hand, filtering with ML prediction outperforms non-ML approach in all metrics.

TABLE III
COMPARISON OF END COORDINATE PERFORMANCE METRICS IN 2D
POSITIONING FOR PT. 20.

	MLE 2D [m]	RMSE 2D [m]	DRMS [m]	Max. error 2D [m]
NLS	0.2	0.24	0.22	1.42
KF	0.16	0.19	0.15	0.85
AKF + XGB	0.14	0.14	0.07	0.24
AKF + RF	0.13	0.14	0.07	0.25
AKF + RT	0.15	0.15	0.06	0.24
EKF	0.25	0.52	0.51	3.22

TABLE IV
COMPARISON OF END COORDINATE PERFORMANCE METRICS IN 3D
POSITIONING FOR PT. 20.

	MLE 3D [m]	RMSE 3D [m]	MRSE [m]	Max. error 3D [m]
NLS	0.57	0.65	0.43	2.7
KF	0.53	0.58	0.31	1.9
AKF + XGB	0.41	0.42	0.14	0.67
AKF + RF	0.44	0.45	0.15	0.69
AKF + RT	0.46	0.47	0.12	0.68
EKF	0.65	0.87	0.74	5.74

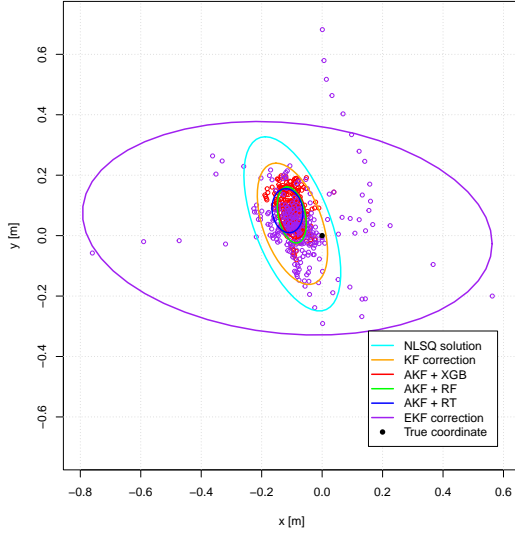


Fig. 14. Comparison of confidence ellipses for pt. 20 with respect to x and y axes. Each ellipse contains samples within one standard deviation (68% confidence). For the sake of clarity, only two point clouds are shown (EKF and AKF+XGB). It can be seen how prediction keeps the point cloud more tightly together, whereas EKF relies only on noisy ranging data which produce much more sparsely distributed samples. Due to outliers, the figure has been zoomed in on the largest ellipse.

Regarding three different ML algorithms it can be seen that even by applying a simple regression tree, the overall metrics are better compared to non-ML filtering. ML performance was summarized with the cumulative error distribution in Fig. 16 and metrics for model prediction performance in Table V. The latter includes commonly used regression performance indicators such as RMSE, mean square error (MSE), and mean absolute error (MAE) [41].

TABLE V
PERFORMANCE METRICS OF ML MODELS ON THE TEST SET.

	RMSE	MSE	MAE
XGBoost	1.28	1.64	0.36
Regression tree	1.37	1.87	0.46
Random forest	1.18	1.4	0.33

Lastly, filtering and ML methods were compared in terms of elapsed time with results shown in Table VI. Benchmarking was done in the R Studio environment using built-in ML libraries *xgboost*, *ranger*, *rpart*, and *microbenchmark*. The hardware specification of the computer was Intel(R) Core(TM) i5-7300U CPU @ 2.60 GHz with 16 GB RAM. It can be seen the amount of delay ML adds to the filtering scheme. Ordinary Kalman filter performs the fastest while EKF being 3.5 times slower. However, ML prediction adds computational delay, with XGBoost and regression tree being approximately 18 times slower than KF and random forest being the slowest.

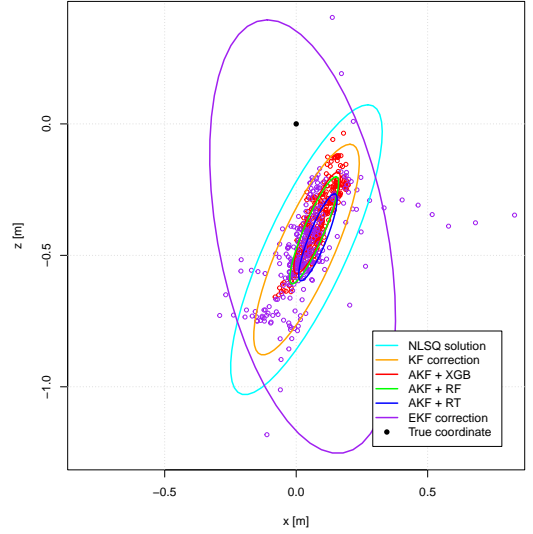


Fig. 15. Comparison of confidence ellipses for pt. 20 with respect to x and z axes. For the sake of clarity, only two point clouds are shown (EKF and AKF+XGB). Additionally, due to outliers, the figure has been zoomed in on the largest ellipse.

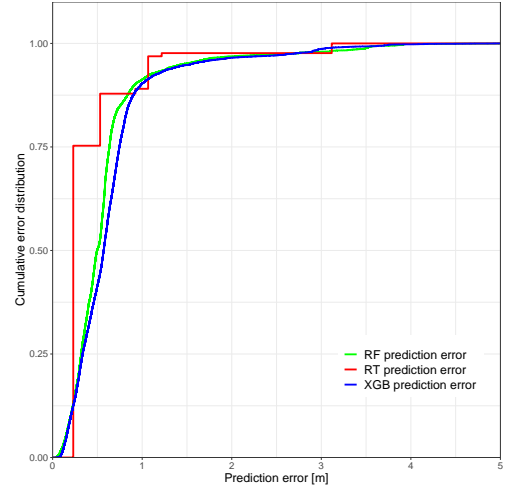


Fig. 16. Cumulative distribution of prediction errors. It can be seen that a regression tree provides more robust prediction levels according to leaf nodes from a single tree as shown in Fig. 5. RF and XGB predictions are smoother at the cost of more complex models.

Finally, the XGBoost model was also applied in the Eliko RTLS UWB positioning solution using the XGBoost C Package [42]. The system hardware consisted of Intel(R) Xeon(R) W-2123 CPU @ 3.60 GHz with 16 GB RAM. The prediction time delay was approximately 1 ms.

TABLE VI
SINGLE ITERATION TIME FOR FILTERING AND PREDICTION.

	Min. time [ms]	Mean time [ms]	Max. time [ms]	Ratio
KF	0.04	0.04	0.06	1
AKF + XGB	0.68	0.71	0.72	17.75
AKF + RF	14.77	15.49	16.74	387.25
AKF + RT	0.68	0.72	0.9	18
EKF	0.12	0.14	0.21	3.5

V. CONCLUSION AND DISCUSSION

In this article, it was investigated how different features of ranging residuals and coordinate calculation can be used in UWB-based positioning integrity estimation. These features were described through statistical metrics like those used in literature as well as several novel ones. The goal was to use different ML methods to select features with the biggest informational gain and based on these select features, predict end coordinate offset from true value. Lastly, this error was used as a measure of uncertainty in a coordinate filtering scheme and compared with non-ML-driven filters. It was shown that ML models provide significant improvement in terms of accuracy and precision in both 2D and 3D positioning. Overall statistics show that ML-driven filtering has approximately 0.1 m less MLE and 0.3 m smaller DRMS than compared to ordinary KF in 2D positioning. All of the tested methods were also compared in terms of processing time. ML-driven methods presented a significant delay when compared to ordinary coordinate filtering due to added model-based prediction. However, the processing time was adequate to be used in a high update rate (e.g., 10 Hz) positioning system. Additionally, it was seen how the regression tree algorithm has approximately the same amount of delay as a much more complex XGBoost, which consists of 150 consecutive boosting trees. Algorithm runtime may be related to the ML library's implementation, hardware specification, and efficiency of the code. Therefore, the actual implementation in a dedicated RTLS system might result in an even smaller processing delay.

ACKNOWLEDGMENT

The authors would like to thank Dr. Taavi Laadung and Aleksei Fjodorov for their valuable suggestions and feedback.

REFERENCES

- [1] S. Hayward, K. van Lopik, C. Hinde, and A. West, "A survey of indoor location technologies, techniques and applications in industry," *Internet of Things*, p. 100608, 2022.
- [2] X. Li and Y. Wang, "Research on a factor graph-based robust UWB positioning algorithm in NLOS environments," *Telecommunication Systems*, vol. 76, pp. 207–217, 2021.
- [3] M. A. Al-Ammar, S. Alhadhrami, A. Al-Salman, A. Alarifi, H. S. Al-Khalifa, A. Alnafessah, and M. Alsaleh, "Comparative survey of indoor positioning technologies, techniques, and algorithms," in *2014 International Conference on Cyberworlds*. IEEE, 2014, pp. 245–252.
- [4] M. Ridolfi, A. Kaya, R. Berkvens, M. Weyn, W. Joseph, and E. D. Poorter, "Self-calibration and collaborative localization for UWB positioning systems: A survey and future research directions," *ACM Computing Surveys (CSUR)*, vol. 54, no. 4, pp. 1–27, 2021.
- [5] B. J. Silva and G. P. Hancke, "Non-line-of-sight identification without channel statistics," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2020, pp. 4489–4493.
- [6] Y. Guo, W. Li, G. Yang, Z. Jiao, and J. Yan, "Combining dilution of precision and kalman filtering for UWB positioning in a narrow space," *Remote Sensing*, vol. 14, no. 21, p. 5409, 2022.
- [7] M. Wang, Z. Chen, Z. Zhou, J. Fu, and H. Qiu, "Analysis of the applicability of dilution of precision in the base station configuration optimization of ultrawideband indoor TDOA positioning system," *IEEE Access*, vol. 8, pp. 225 076–225 087, 2020.
- [8] A. Maali, H. Mimoun, G. Baudoin, and A. Ouldali, "A new low complexity NLOS identification approach based on UWB energy detection," in *2009 IEEE Radio and Wireless Symposium*. IEEE, 2009, pp. 675–678.
- [9] Z. Xiao, H. Wen, A. Markham, N. Trigoni, P. Blunsom, and J. Frolik, "Non-line-of-sight identification and mitigation using received signal strength," *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1689–1702, 2014.
- [10] S. Marano, W. M. Gifford, H. Wymeersch, and M. Z. Win, "NLOS identification and mitigation for localization based on UWB experimental data," *IEEE Journal on selected areas in communications*, vol. 28, no. 7, pp. 1026–1035, 2010.
- [11] J. Fan and A. S. Awan, "Non-line-of-sight identification based on unsupervised machine learning in ultra wideband systems," *IEEE Access*, vol. 7, pp. 32 464–32 471, 2019.
- [12] M. J. Bocus and R. J. Piechocki, "Passive unsupervised localization and tracking using a multi-static uwb radar network," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 01–06.
- [13] V. Barral, C. J. Escudero, and J. A. García-Naya, "NLOS classification based on RSS and ranging statistics obtained from low-cost UWB devices," in *2019 27th European Signal Processing Conference (EUSIPCO)*. IEEE, 2019, pp. 1–5.
- [14] V. Barral, C. J. Escudero, J. A. García-Naya, and P. Suárez-Casal, "Environmental cross-validation of NLOS machine learning classification/mitigation with low-cost UWB positioning systems," *Sensors*, vol. 19, no. 24, p. 5438, 2019.
- [15] D. Liu, M.-C. Lee, C.-M. Pun, and H. Liu, "Analysis of wireless localization in nonline-of-sight conditions," *IEEE transactions on vehicular technology*, vol. 62, no. 4, pp. 1484–1492, 2013.
- [16] Y.-T. Chan, W.-Y. Tsui, H.-C. So, and P.-c. Ching, "Time-of-arrival based localization under NLOS conditions," *IEEE Transactions on vehicular technology*, vol. 55, no. 1, pp. 17–24, 2006.
- [17] T. Therneau and B. Atkinson, *rpart: Recursive Partitioning and Regression Trees*, 2022, r package version 4.1.19. [Online]. Available: <https://CRAN.R-project.org/package=rpart>
- [18] M. N. Wright and A. Ziegler, "ranger: A fast implementation of random forests for high dimensional data in C++ and R," *Journal of Statistical Software*, vol. 77, no. 1, pp. 1–17, 2017.
- [19] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou, M. Li, J. Xie, M. Lin, Y. Geng, Y. Li, and J. Yuan, *xgboost: Extreme Gradient Boosting*, 2022, r package version 1.6.0.1. [Online]. Available: <https://CRAN.R-project.org/package=xgboost>
- [20] J. Guillory, D. Truong, and J.-P. Wallerand, "Multilateration with self-calibration: uncertainty assessment, experimental measurements and Monte-Carlo simulations," *Metrology*, vol. 2, no. 2, pp. 241–262, 2022.
- [21] P. Moravek, D. Komosny, M. Simek, and J. Muller, "Multilateration and flip ambiguity mitigation in ad-hoc networks," *Przeglad Elektrotechniczny*, vol. 2012, no. 05b, pp. 222–229, 2012.
- [22] K. Liu and Z. Li, "Adaptive kalman filtering for UWB positioning in following luggage," in *2019 34rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. IEEE, 2019, pp. 574–578.
- [23] I. Guvenc, S. Gezici, F. Watanabe, and H. Inamura, "Enhancements to linear least squares localization through reference selection and ML estimation," in *2008 IEEE Wireless Communications and Networking Conference*. IEEE, 2008, pp. 284–289.
- [24] I. Guvenc, C.-C. Chong, and F. Watanabe, "NLOS identification and mitigation for UWB localization systems," in *2007 IEEE Wireless Communications and Networking Conference*. IEEE, 2007, pp. 1571–1576.
- [25] W. Murphy and W. Hereman, "Determination of a position in three dimensions using trilateration and approximate distances," *Department of Mathematical and Computer Sciences, Colorado School of Mines, Golden, Colorado, MCS-95*, vol. 7, p. 19, 1995.
- [26] "Welcome to Krah Pipes - Krah Pipes — krah-pipes.ee," <http://www.krah-pipes.ee/eng/>, [Accessed 15-Nov-2023].
- [27] Decawave Ltd, "DW1000 Datasheet," <https://www.qorvo.com/products/d/da007946>, [Accessed 15-Nov-2023].

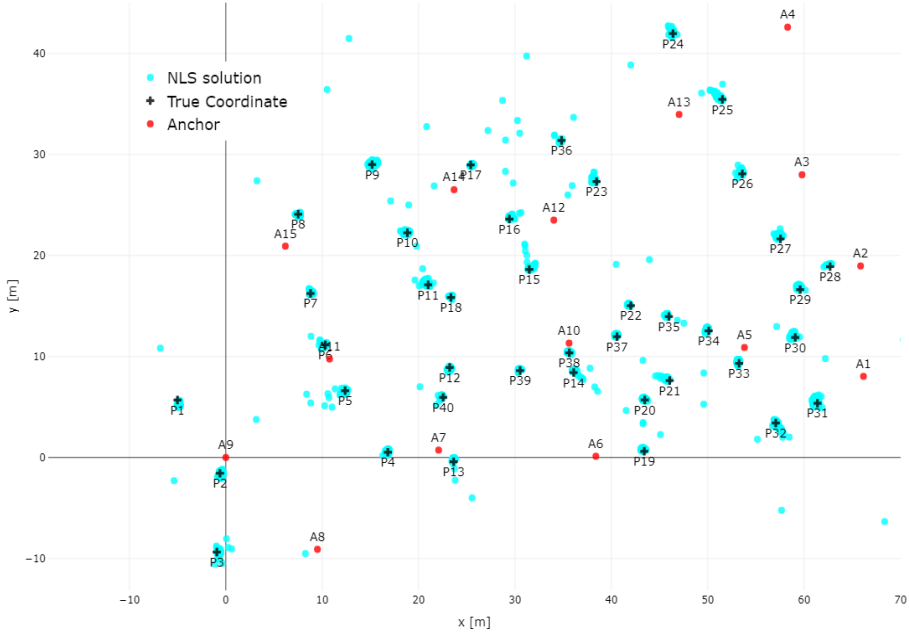


Fig. 17. Overview of measurement campaign in Auroom factory. Measurements were done at 40 separate points around the factory's indoor area.

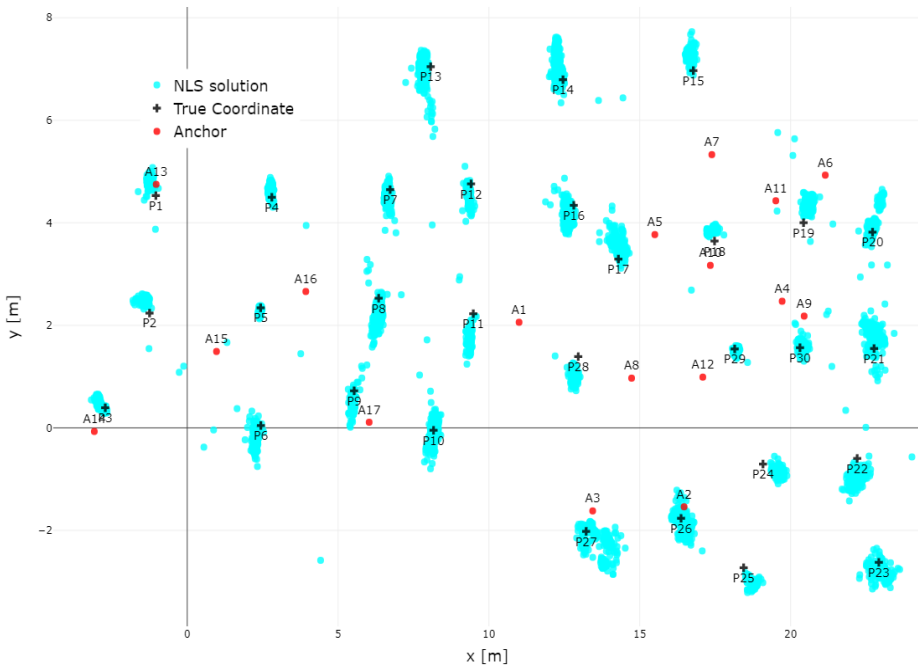


Fig. 18. Overview of measurement campaign in Eliko office. Several measurement points (e.g., 14, 15, 22, 23) are not surrounded by anchors and have no LOS with them.

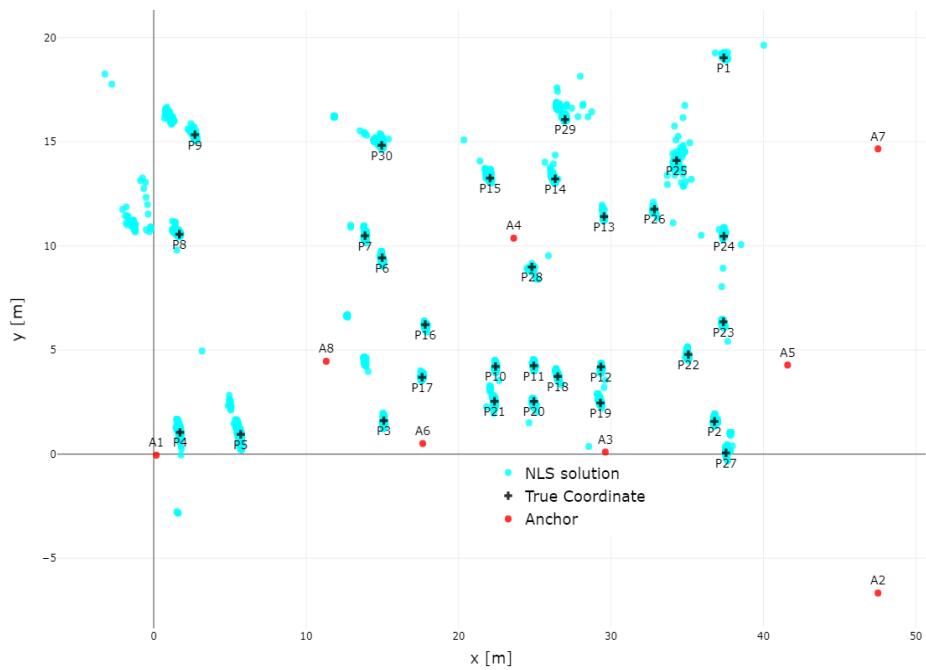


Fig. 19. Overview of measurement campaign in Krah Pipes factory.

- [28] I. Dotlic, A. Connell, and M. McLaughlin, "Ranging methods utilizing carrier frequency offset estimation," in *2018 15th Workshop on Positioning, Navigation and Communications (WPNC)*, 2018, pp. 1–6.
- [29] Leica Geosystems AG, "Leica DISTO S910," https://shop.leica-geosystems.com/sites/default/files/2019-04/leica_disto_s910_fol.828429.0115.en.pdf, [Accessed 03-Jan-2024].
- [30] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2021. [Online]. Available: <https://www.R-project.org/>
- [31] Kuhn and Max, "Building predictive models in r using the caret package," *Journal of Statistical Software*, vol. 28, no. 5, p. 1–26, 2008. [Online]. Available: <https://www.jstatsoft.org/index.php/jss/article/view/v028i05>
- [32] O. M. Badawy and M. A. B. Hasan, "Decision tree approach to estimate user location in WLAN based on location fingerprinting," in *2007 National Radio Science Conference*. IEEE, 2007, pp. 1–10.
- [33] C. C. Aggarwal *et al.*, *Data mining: the textbook*. Springer, 2015, vol. 1.
- [34] Y. Xiong, Y. Zhang, X. Guo, C. Wang, C. Shen, J. Li, J. Tang, and J. Liu, "Seamless global positioning system/inertial navigation system navigation method based on square-root cubature kalman filter and random forest regression," *Review of Scientific Instruments*, vol. 90, no. 1, 2019.
- [35] S. Ramraj, N. Uzir, R. Sunil, and S. Banerjee, "Experimenting XGBoost algorithm for prediction and classification of different datasets," *International Journal of Control Theory and Applications*, vol. 9, no. 40, pp. 651–662, 2016.
- [36] S. Li and X. Zhang, "Research on orthopedic auxiliary classification and prediction model based on XGBoost algorithm," *Neural Computing and Applications*, vol. 32, pp. 1971–1979, 2020.
- [37] D.-H. Kim, A. Farhad, and J.-Y. Pyun, "UWB positioning system based on LSTM classification with mitigated NLOS effects," *IEEE Internet of Things Journal*, vol. 10, no. 2, pp. 1822–1835, 2022.
- [38] B. Silva and G. P. Hancke, "Ranging error mitigation for through-the-wall non-line-of-sight conditions," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 11, pp. 6903–6911, 2020.
- [39] G. Laveti, G. S. Rao, K. J. Rani, A. Naline, and A. M. Babu, "GPS receiver SPS accuracy assessment using LS and LQ estimators for precise navigation," in *2014 Annual IEEE India Conference (INDICON)*. IEEE, 2014, pp. 1–5.
- [40] T. Laadung, S. Ulp, A. Fjodorov, M. M. Alam, and Y. Le Moullec, "Adaptive extended kalman filter position estimation based on ultra-wideband active-passive ranging protocol," *IEEE Access*, 2023.
- [41] J. S. Kushwah, A. Kumar, S. Patel, R. Soni, A. Gawande, and S. Gupta, "Comparative study of regressor and classifier with decision tree using modern tools," *Materials Today: Proceedings*, vol. 56, pp. 3571–3576, 2022.
- [42] xgboost developers, "XGBoost C Package," <https://xgboost.readthedocs.io/en/stable/c.html>, 2022, [Accessed 15-Nov-2023].



MIHKEL TOMMINGAS received his higher education at Tartu Aviation College in 2008 and M.Sc. degree in telecommunication from Tallinn University of Technology in 2017 where he is currently pursuing the Ph.D. degree in information and communication technology. He has worked in both the military and civilian sectors. 2008 - 2015 as a Communications and IT officer in the Estonian Air Force. 2017 - 2021 as a Radio Access Network (RAN) measurement specialist at Elisa Corporation. Since 2021, he has worked as a Researcher at OÜ Eliko Tehnoloogia Arenduskeskus. His current research focuses on the improvement of indoor and multi-network positioning algorithms.



MUHAMMAD MAHTAB ALAM (Senior Member, IEEE) received the M.Sc. degree in electrical engineering from Aalborg University, Denmark, in 2007, and the Ph.D. degree in signal processing and telecommunication from the INRIA Research Center, University of Rennes 1, France, in 2013. From 2014 to 2016, he was Post-Doctoral Research at the Qatar Mobility Innovation Center, Qatar. In 2016, he joined as the European Research Area Chair and as an Associate Professor with the Thomas Johann

Seebeck Department of Electronics, Tallinn University of Technology, where he was elected as a Professor in 2018 and Tenured Full Professor in 2021. Since 2019, he has been the Communication Systems Research Group Leader. He has over 15 years of combined academic and industrial multinational experiences while working in Denmark, Belgium, France, Qatar, and Estonia. He has several leading roles as PI in multimillion Euros international projects funded by European Commission (Horizon Europe LATEST-5GS, 5G-TIMBER, H2020 5GROUTES, NATOSPS (G5482), Estonian Research Council (PRG424), Telia Industrial Grant etc. He is an author and co-author of more than 100 research publications. He is actively supervising a number of Ph.D. and Postdoc Researchers. He is also a contributor in two standardization bodies (ETSI SmartBAN, IEEE-GreenICT-EECH), including "Rapporteur" of work item: DTR/ SmartBAN 0014. His research focuses on the fields of wireless communications-connectivity, mobile positioning, 5G/6G services and applications.



IVO MÜRSEPP Education: 1998 – 2002 Tallinn University of Technology (TUT), BSc in telecommunication cum laude; 2002-2004 TUT, MSc in telecommunication; 2004-2013 TUT, PhD in telecommunication.

Professional career: 2002-2015 Teaching assistant in TUT institute of radio- and communication technology; 2015 - 2018 Associate Professor in TUT institute of radio- and communication technology; 2015 – 2020 specialist of IT systems in cyber command of Estonian defence forces;

since 2018 senior lecturer in TUT Thomas Johann Seebeck institute of Electronics.

Main research areas: mobile communications, mobile positioning, indoor positioning.



SANDER ULP received the M.Sc. degree in telecommunication and the Ph.D. degree in information and communication technology from the Tallinn University of Technology, in 2013 and 2019, respectively. In 2018, he was a Researcher with OÜ Eliko Tehnoloogia Arenduskeskus. Since 2019, he has held the position of Chief Technology Officer (CTO) at this technology competence center, which is dedicated to pioneering advancements in indoor positioning research and technologies. His current research

interests include distributed estimation, learning and adaptation over networks, digital signal processing, localization technologies and indoor positioning.

Appendix 3

III

M. Tommingas, T. Laadung, S. Varbla, I. Mürsepp, and M. Mahtab Alam, "UWB and GNSS Sensor Fusion Using ML-Based Positioning Uncertainty Estimation," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 2177–2189, 2025

UWB and GNSS Sensor Fusion Using ML-Based Positioning Uncertainty Estimation

MIHKEL TOMMINGAS¹, TAAVI LAADUNG², SANDER VARBLA¹, IVO MÜÜRSEPP¹,
AND MUHAMMAD MAHTAB ALAM¹ (Senior Member, IEEE)

¹Thomas Johann Seebeck Department of Electronics, Tallinn University of Technology, 19086 Tallinn, Estonia

²ÕÜ Eliko Tehnoloogia Arenduskeskus, 12918 Tallinn, Estonia

CORRESPONDING AUTHOR: M. TOMMINGAS (e-mail: mihkel.tommingas@taltech.ee).

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Program under Grant 101058505 '5G-TIMBER'; in part by the project "Increasing the knowledge intensity of Ida-Viru entrepreneurship" co-funded by the European Union as well as by NATO-SPS G7699 PROTECT; and in part by the project of "Development of an industrial digital control system based on precise positioning technology ELIKO TAK and Atemix Automatika" under Grant 2014-2020.4.02.21-0311. The work of Sander Varbla was supported by the Tallinn University of Technology under Grant GFEASV23.

ABSTRACT This article presents a novel machine learning (ML) augmented sensor fusion scheme for seamless indoor-outdoor positioning using ultra-wideband (UWB) and global navigation satellite system (GNSS) sensors. Dilution of precision (DOP) is a common metric that describes the level of geometrical uncertainty and is often applied in sensor fusion schemes. However, it does not fully reflect other factors that may influence positioning integrity, such as signal quality, pseudorange error, or the number of servicing nodes. An incorrect estimation of sensor uncertainty may significantly affect the precision and robustness of individual sensors and their fused positioning solution, especially in indoor-outdoor transition zones, where positioning is most challenging. Therefore, this article proposes a sensor fusion scheme augmented with two distinct extreme gradient boosted (XGBoost) ML models for estimating UWB and GNSS positioning uncertainties. Trained using real-life datasets, these models have the advantage of considering an ensemble of features rather than a single parameter to estimate the uncertainty of the current coordinate. In contrast to sensor fusion solutions, which implement only highly accurate RTK fixed solution mode, the GNSS model can operate with different correction qualities as well. The proposed scheme was applied on a moving testbed with UWB and GNSS sensors while relying on the ML models to enhance the coordinate filtering. The results show that the ML-based approach can improve seamless transition between indoor and outdoor areas with almost no sensor dropouts with a mean positioning error of 0.16 m and a maximum error of approximately 0.5 m.

INDEX TERMS Coordinate uncertainty estimation and filtering, indoor and outdoor navigation, machine learning, seamless UWB and GNSS positioning, sensor fusion.

I. INTRODUCTION

USING positioning sensors to achieve an accurate and precise end coordinate of an object indoors, outdoors, or between those areas is a challenging task. The performance of these systems is highly dependent on the operational environment. For example, the global navigation satellite system (GNSS) performs well in outdoor open-sky areas with unobstructed reception from servicing satellites. On the other hand, the signal from satellites is usually severely degraded indoors or in urban areas, making GNSS positioning challenging [1], [2], [3]. Similarly, considering

an ultra-wideband (UWB) based positioning system, it is also intended to work within an area serviced by a network of UWB anchors [4]. UWB systems typically employ radio-frequency (RF) signals in a frequency range between 3.1 GHz to 10.6 GHz and a bandwidth greater than 500 MHz [5], [6]. The latter attribute makes the UWB signal more robust in the presence of multipath effects and less susceptible to interference when compared to other RF-signal-based positioning systems [4], [7], [8]. Usually established indoors, the UWB-based system provides an alternative positioning solution in GNSS-denied environments.

For a positioning solution that is capable of locating an object both indoors and outdoors with a seamless transition from one environment to another, the system would need to incorporate information from both indoor and outdoor positioning sensors. Based on numerous works presented in the literature, it can be seen that achieving a reliable transition between the two environments is still a challenge. The problem arises when moving positioning sensors from one operational environment to the other (e.g., indoors to outdoors). As the signal quality of one positioning sensor degrades, the error of the sensor coordinates increases. On the other hand, when transitioning into an area dominated by signals of the other sensor, the positioning uncertainty of this positioning system decreases with an increase in signal quality. While seamless indoor-outdoor positioning accuracy and precision may be enhanced by implementing additional sensors (e.g., inertial measurement unit (IMU) or wheel sensor), the end coordinate still depends on the integrity of each individual positioning sensor. Therefore, the current article seeks to enhance the seamless coordinate accuracy and precision using only information from UWB and GNSS sensors.

In a seamless positioning system, one of the main challenges is to determine the uncertainty of the end coordinate in the transition zone. While both indoor and outdoor sensors are operating at the edge of their operational area, it is still necessary to estimate their uncertainty. An appropriate integrity parameter in a coordinate filtering scheme (e.g., Kalman filter (KF)) may significantly improve seamless transition from one operational area to the next. Coordinate solutions from different sensors would be given weights based on their positioning integrity, thereby increasing bias toward the solution with less uncertainty.

In the literature, there are different approaches to determining this uncertainty. For example, Zhang et al., proposed to use the circular error probable (CEP) metric for the indoor UWB network, while positioning error from global positioning system (GPS) was described as the difference between current GPS measurement and the estimate of the integrated magnetic, angular rate, and gravity (MARG)/GPS solution [9]. They reported an average positioning accuracy of approximately 3 m. As CEP is primarily used to evaluate the precision of stationary points, it would be difficult to use this metric for a moving object. Furthermore, as pointed out by Lv, Wang, Jin, and Shen, this method needs to know the prior error distribution of the UWB system in the deployment area. Additionally, the positioning systems are switched based on values presented by GNSS horizontal dilution of precision (HDOP) and UWB CEP threshold, which may not guarantee smooth and stable trajectories in the transition area [10]. The literature shows that dilution of precision (DOP) is often used as a parameter in determining the uncertainty of positioning. Using the estimated position, the DOP value indirectly shows the level of geometrical uncertainty in an area relative to servicing nodes (e.g., GNSS satellites or UWB anchors) [11]. Considering a coordinate

filtering scheme with two sensors, the one with a larger DOP value would be treated as more imprecise compared to another sensor [12]. Lv et al. proposed to use HDOP in their seamless positioning solution that consisted of GNSS real-time kinematics (RTK), UWB, IMU and wheel sensor [10]. They achieved 8 cm static positioning accuracy and a stable transition between indoor and outdoor areas. Yao et al. used HDOP for UWB and carrier-to-noise ratio for GNSS as weights in a tightly coupled GNSS/UWB/inertial navigation system (INS) positioning scheme with federal filtering, achieving approximately 10 cm horizontal accuracy in the transition area [13]. Zhu et al. developed an integrated positioning strategy using GNSS, UWB, dead reckoning (DR), and visual map matching (VMM) with HDOP-based uncertainty weights for GNSS and UWB, achieving approximately 0.8 m overall horizontal accuracy [14].

However, it can be argued that DOP alone does not provide enough information about positioning uncertainty. For example, the accuracy of the GNSS positioning depends on three types of errors. Firstly, there are receiver instrument errors, which may influence the performance of positioning precision. Next, there are errors caused by the signal propagation path. These include ionospheric and tropospheric delays as well as multipath effects. Finally, there are errors caused by the space segment, including satellite ephemeris, clock drift and position errors related to the geometry of constellations. Only the receiver's position relative to orbiting satellites can be observed through dilution of precision values [15], [16]. It can be seen that DOP describes only a small part of the entire ensemble of error sources. Therefore, sensor fusion solutions, incorporating only a single parameter of error (e.g., HDOP), may lack sufficient data to fully describe positioning errors. Moreover, the data interface of the GNSS receiver may output many other parameters such as the number of servicing satellites, age of correction, deviation of position error, correction quality, etc., which may also be considered.

The sources of error for UWB positioning systems may be analyzed similarly. Although not as susceptible to propagation path effects, the UWB positioning quality is affected but not limited by factors such as the number of servicing anchors, their vicinity to the tag, impairments caused by non-line of sight (NLOS), and suitable anchor layout geometry [7], [11], [17].

Geofencing is another method that could be used to switch between indoor and outdoor positioning systems. For example, Wang et al. proposed a tightly coupled GNSS/UWB/INS/Map integration scheme for autonomous vehicles [18]. Indoors, the system compares object position relative to the map and makes observations of UWB signal occlusions (blockage between the UWB anchors and the tag) to reduce NLOS errors, thereby improving seamless transition between indoor and outdoor areas. Di Pietra, Dabov and Piras suggested to use geofencing as a trigger to switch positioning sensors between indoor and outdoor areas in a pedestrian navigation scheme with GNSS, UWB,

and INS sensors [19]. While geofencing has advantages in discarding erroneous indoor or outdoor positioning data, it can be argued that several disadvantages exist. Firstly, while in a transition area, the end coordinates of an indoor or outdoor positioning sensor might be cut off too soon, as reception of these sensors usually extends a significant distance into their adjacent operational area. Secondly, geofencing cannot describe uncertainty of the positioning coordinate. Lastly, this method requires prior knowledge of the operational area, usually in the form of a predefined map or track making the solution area-specific and cumbersome to implement.

In contrast to the aforementioned methods, this paper presents an alternative approach in estimating the positioning uncertainty using machine learning (ML). One of the main advantages of ML-based approaches is the ability to make decisions effectively using observed data without explicit mathematical formulation [20]. Compared to traditional statistical methods, ML techniques enable the identification of complex dependencies in data that may not be apparent through exploratory data analysis. The main contributions of this paper are as follows:

- This article proposes a machine learning (ML) augmented complementary sensor fusion solution for GNSS and UWB positioning systems. The extreme gradient boosted (XGBoost) models incorporate a combination of features that could provide a more comprehensive estimation of coordinate uncertainty compared to single parameter solutions. These features include ones used in the literature as well as parameters that, to the best of author's knowledge, have not been applied before. For example, when DOP is a parameter commonly used as a sole indicator of positioning uncertainty, then in this article it is just one of many features used in both the GNSS and UWB ML models. The models' output provides an estimate of the sensor's true coordinate offset and is used as measurement uncertainty within the coordinate filtering process.
- The GNSS ML model has been trained using data with different correction qualities: differential GPS (DGPS), floating-point RTK, and RTK fix. Such an approach contrasts with other works that only consider highly accurate RTK fix data or propose tightly-coupled fusion where the RTK floating-point ambiguities are solved within a dedicated algorithm [10], [21], [22], [23].
- The GNSS and UWB ML models were trained on data collected in real-life industrial and commercial buildings. For unbiased results, the obtained models were tested in a separate environment. Therefore, it may be suggested that this solution does not require prior knowledge of the deployment area.

The solution in this article can be considered as semi-loosely coupled GNSS and UWB fusion. UWB coordinates and features are extracted from raw UWB ranging information making it a tightly coupled part of the solution.

On the other hand, GNSS sensor features and coordinates are extracted from the readily available National Marine Electronics Association (NMEA) messages with no need for feature and coordinate calculations. Therefore, the GNSS sensor is loosely coupled, in contrast to tightly coupled solutions, which rely on extracting pseudorange data from the GNSS receiver. Moreover, the ML augmented sensor fusion could be potentially used with off-the-shelf RTK devices, provided that these output appropriate features for the ML model.

The UWB ML model was trained based on data gathered at three different indoor environments, which contained a UWB anchor network. The GNSS ML model was trained based on data collected at a single location, which included outdoor, near-building, and indoor areas. Both models were trained and tested using the XGBoost ML library [24]. The two models were then used to estimate the offset from the true coordinate for the respective positioning sensor. These estimates along with GNSS and UWB coordinates were then combined using measurement fusion. Finally, the fused values were then applied in a coordinate filtering scheme of an Adaptive Kalman Filter (AKF). In essence, the merit of using a ML model is to incorporate more than one feature to enhance the estimation of uncertainty of the respective sensor, thereby improving end coordinate accuracy and precision in a multi-sensor positioning solution.

The paper is organized as follows: Section II describes end coordinate estimation for UWB and GNSS localization as well as features used in both models. Section III gives an overview of data collection, ML model training, validation, and testing. Section IV describes the application of the ML uncertainty estimate in sensor fusion and coordinate filtering. Section V describes testing equipment, area, and test results. The article is concluded in Section VI.

II. COORDINATE ESTIMATION AND FEATURES

This article proposes a UWB and GNSS sensor fusion algorithm that employs ML-based positioning uncertainty estimations. As stated in the introduction, accurate coordinate uncertainty estimation has a central role in achieving an accurate fused coordinate, especially in situations where the integrity of one or several sensors is compromised. The general flowchart is shown in Fig. 1 and comprises of multiple steps. The GNSS outputs geodetic coordinates as well as GNSS RTK-specific features for the ML model. The global coordinates are transformed into a local frame of reference using the East-North-Up (ENU) method.

On the other hand, the UWB sensor outputs only ranging information, which is used to calculate UWB tag coordinates using multilateration with a non-linear least squares (NLS) approach. The features of the UWB ML model are calculated based on the NLS solution and initial ranging information. As each ML model produces a separate estimate for uncertainty, these estimates are merged with covariance fusion techniques. Local coordinates would be fused and weighted based on ML predictions. Lastly, fusion output is

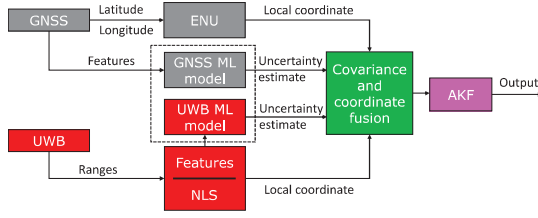


FIGURE 1. Flowchart of UWB and GNSS ML augmented sensor fusion. The dashed box represents the author's contribution in developing two distinct ML models for respective sensor uncertainty estimation. Sensor coordinates and their dynamically changing position uncertainties are fused and filtered to produce the end coordinate at the output.

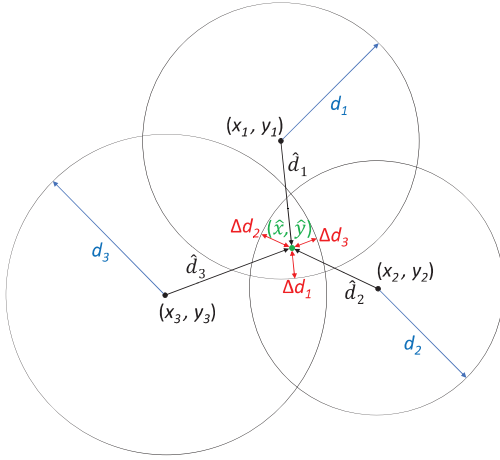


FIGURE 2. Example of a 2D trilateration scheme with inaccurate range measurements. The difference between the distance to the estimated position \hat{d}_i and the actual measured range d_i results in a residual Δd_i that can be used in estimating positioning uncertainty [27].

filtered with an adaptive Kalman filter (AKF) to produce the final coordinate estimate.

A. UWB LOCAL COORDINATE CALCULATION

Estimating the position of the UWB tag, regarding surrounding anchors, represents a problem of multilateration. Fig. 2 shows an object on coordinates (\hat{x}, \hat{y}) in the vicinity of three anchors denoted with (x_i, y_i) . Distance measurements d_i from each individual anchor can then be used to estimate the tag's position. Usually, to obtain a solution in 2D space, at least three-, and in 3D space four anchors are required [25]. Additionally, the geometry of the anchor layout may influence positioning results. For example, anchors positioned in a straight line may produce flip ambiguity with possible solutions on either side of the line [26]. Without any measurement errors, $d_i = \hat{d}_i$ and the least squares (LS) method provides a solution at the intersection of the three circles [28]. However, real-life positioning is usually impacted by ranging noise and NLOS, thereby requiring the application of optimization techniques [8], [29]. In this

article, the 3D end coordinate of the tag is first estimated with LS and then further optimized with a non-linear least squares (NLS) approach by minimizing the objective function:

$$\hat{x}, \hat{y}, \hat{z} = \underset{x, y, z}{\operatorname{argmin}} \sum_{i=1}^N \left((x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2 - d_i^2 \right)^2 \quad (1)$$

where x , y and z are the coordinates that provide the smallest error. Detailed LS and NLS calculation steps are explained in different works by authors such as Guvenc and Guillory [25], [30].

The difference between a measured distance d_i from the respective anchor, and distance \hat{d}_i from the estimated coordinate $(\hat{x}, \hat{y}, \hat{z})$, is represented as a ranging residual Δd_i as:

$$\Delta d_i = d_i - \hat{d}_i. \quad (2)$$

In this work, it is suggested that ranging residuals and their statistical values are used in the UWB ML model to indirectly describe UWB positioning integrity.

B. FEATURES OF UWB ML MODEL

The UWB ML model estimates coordinate uncertainty based on statistics of ranging residuals, coordinate optimization techniques, and geometrical integrity. The ensemble of features used by the model is following: LS/NLS difference, sum of squares of ranging residuals, sum of squares of positive ranging residuals, residual variance, number of NLS iterations, position DOP (PDOP), mean of residuals, and variance of positive residuals. Feature calculation equations are given in the author's article about ML-based UWB positioning integrity estimation [27].

C. GNSS LOCAL COORDINATE CALCULATION

East-North-Up transformation is a method of projecting geodetic coordinates onto a local flat tangent plane. For example, the geodetic coordinates extracted from the GNSS receiver can be transformed onto Cartesian plane, which is more intuitive, and allows to make distance calculation using Euclidean geometry. This projection is practical for establishing a unified indoor-outdoor reference frame in small areas. To define the projection, it is needed to specify the point of tangency (origin). That location becomes the center of projection and is usually the center of the project site [31].

The transformation is twofold. Firstly, the geodetic latitude ϕ , longitude λ , and height h values are converted into Earth-centered, Earth-fixed (ECEF) coordinate system [32]:

$$N(\phi) = \frac{a^2}{\sqrt{a^2 \cos^2 \phi + b^2 \sin^2 \phi}}, \quad (3)$$

$$X = (N(\phi) + h) \cos \phi \cos \lambda, \quad (4)$$

$$Y = (N(\phi) + h) \cos \phi \sin \lambda, \quad (5)$$

$$Z = \left(\frac{b^2}{a^2} N(\phi) + h \right) \sin \phi, \quad (6)$$

TABLE 1. Features from NMEA messages [33].

Message	Field	Description	Symbol	Example
\$GPGGA	7	GPS quality indicator	x	4
\$GPGGA	8	Number of satellites in use	xx	11
\$GPGGA	9	Horizontal dilution of precision	x.x	1.1
\$GPGGA	14	Age of correction data (in seconds)	xx	8
\$GPGST	3	RMS value of the standard deviation of the pseudorange measurements	x.x	2.7
\$GPGST	7	Standard deviation of latitude error (m)	x.x	1.2
\$GPGST	8	Standard deviation of longitude error (m)	x.x	3.2
\$GPGST	9	Standard deviation of altitude error (m)	x.x	4.5

where a and b are Earth's equatorial and polar radii respectively and X, Y, Z represent ECEF coordinates. This transformation would be done using geodetic coordinates of both the point of interest (e.g., GNSS rover) and point of origin (e.g., GNSS base station), denoted by X_p, Y_p, Z_p and X_o, Y_o, Z_o respectively. Next, these values are shifted with regards to the point of origin. The vector values (E, N and U) from the origin to the point of interest are calculated as:

$$\mathbf{R}_r = \begin{bmatrix} -\sin \lambda_o & \cos \lambda_o & 0 \\ -\sin \phi_o \cos \lambda_o & \sin \phi_o \sin \lambda_o & \cos \phi_o \\ \cos \phi_o \cos \lambda_o & \cos \phi_o \sin \lambda_o & \sin \phi_o \end{bmatrix}, \quad (7)$$

$$\mathbf{R}_s = \begin{bmatrix} X_p - X_o \\ Y_p - Y_o \\ Z_p - Z_o \end{bmatrix}, \quad (8)$$

$$\mathbf{R}_r \times \mathbf{R}_s = \begin{bmatrix} E \\ N \\ U \end{bmatrix}, \quad (9)$$

where \mathbf{R}_r and \mathbf{R}_s mark the rotation and shifting matrices respectively. ENU coordinates for the point of origin are (0, 0, 0) and the length between the origin and point of interest can be calculated with the Euclidean distance formula. ENU method satisfies the need to map the coordinates in a small area on the Earth as it conforms so nearly to a plane that distortion on such a system is negligible [31].

D. GNSS FEATURES

The GNSS receiver used in this research outputs data with the following NMEA 0183 message headers: *\$GPGGA*, *\$GPGST*, *\$GPZDA*, *\$GPRMC* and *\$GPVTG*. Table 1 presents the fields considered as most relevant in describing positioning quality and therefore applied in GNSS ML model training. As mentioned in the introduction, the GNSS features can be extracted from the serial interface of the receiver and require no additional calculations. The GNSS device used in this research operates with three correction

states, which indicator can be extracted at *\$GPGGA* field 7. These corrections include DGPS (value 2), floating-point RTK (value 5) and fixed RTK (value 4).

DGPS is an extension of GPS technology, based on satellite and terrestrial communication. A station with a known location is considered as a source of reference, which communicates additional data to the receiver to reduce localization error. The idea lies in determining errors related to pseudorange observables and is calculated by comparing the value from the GNSS receiver and the value computed using the coordinates of the satellites and the reference station [34]. In urban areas, the DGPS method may provide a positioning accuracy lower than 10 m [16].

GNSS RTK is a precise positioning method capable of delivering cm-level accuracy. By leveraging simultaneous carrier-phase measurements from both the GNSS base station and the receiver, it implements differencing techniques to eliminate signal phase biases, clock offsets as well as tropospheric and ionospheric errors. A key challenge in RTK is resolving the integer ambiguity, which represents the exact number of wavelengths between a satellite and the receiver. Before the ambiguities are resolved, the GNSS receiver uses floating-point estimates for the ambiguity. Once a valid integer solution is determined, the receiver transitions to RTK fixed ambiguity solution mode, resulting in a significantly improved positioning accuracy [35], [36].

The *\$GPGST* log contains pseudorange measurement noise statistics that are translated into the position domain to give statistical measures of the quality of the position solution. This log reflects the accuracy of the solution type used in the *\$GPGGA* message, except for the RMS field, which does not represent carrier-phase-based positions but the accuracy of the pseudorange position [33].

III. DATA COLLECTION, ML MODEL TRAINING AND TESTING

A. UWB MEASUREMENT DATA COLLECTION, PROCESSING AND MODEL TRAINING

The UWB ML model was trained on data collected at two different sites which contained an indoor UWB network: OÜ Krah Pipes factory and OÜ Eliko Tehnoloogia Arenduskeskus office. Based on the Qorvo's DW1000 chip, the Eliko real-time locating system (RTLS) was set to operate on UWB channel 4 [37]. It also uses active-passive two-way ranging (AP-TWR) protocol with clock offset error mitigation [38]. The goal was to collect UWB ranging data at different locations, calculate UWB model features, and assign separately measured ground truth offset as the ML model's response variable. The ground truth was measured in a local frame of reference with the Leica DISTO S910 laser distance measurement tool, which assigned 3D coordinates to UWB anchors as well as the tag. The measurement tool has a typical accuracy of ± 1 mm [39].

The model was tested in a separate area at Auroom Kastre factory (Fig. 3), which contained a network of 15

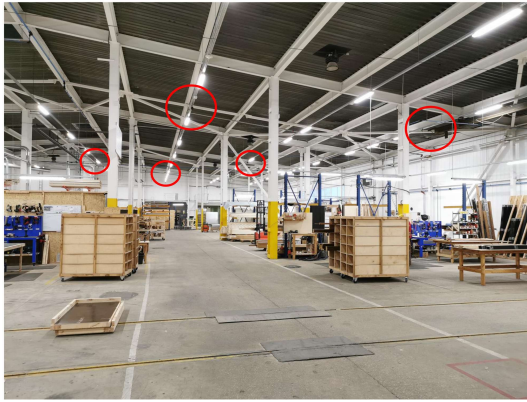


FIGURE 3. Industrial site at Auroom Kastre factory. Five of a total of 15 UWB anchors are highlighted with red circles.

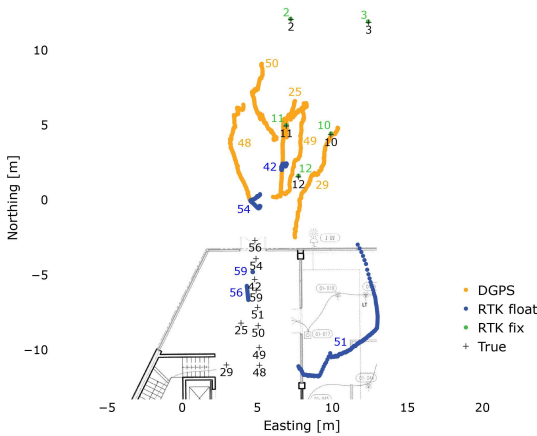


FIGURE 4. Layout of the GNSS measurement campaign at the Eliko office building. Static measurements were taken indoors, near-building, and in outdoor areas. For clarity, only 15 measurement points out of a total 60 measurements are shown in the figure. The blue and orange traces mark the highly inaccurate and imprecise DGPS and RTK float solutions taken indoors. Measurements that were taken closer to the building door, were also more accurate and precise, while points with RTK fix solution had the best precision and accuracy.

UWB anchors. Ranging and ground truth data were collected at 40 different points around the factory. Detailed maps of UWB measurement campaigns and ML model training steps can be seen in the author's publication concerning UWB coordinate corrections with ML techniques [27]. In the current article, the XGBoost ML model was used to estimate the uncertainty of UWB positioning using features mentioned in Section II-B.

B. GNSS MEASUREMENT DATA COLLECTION

GNSS measurement data were collected with the Fieldbee L2 RTK receiver at an office building site, as seen in Fig. 4 [40]. Before measurement collection, the RTK base

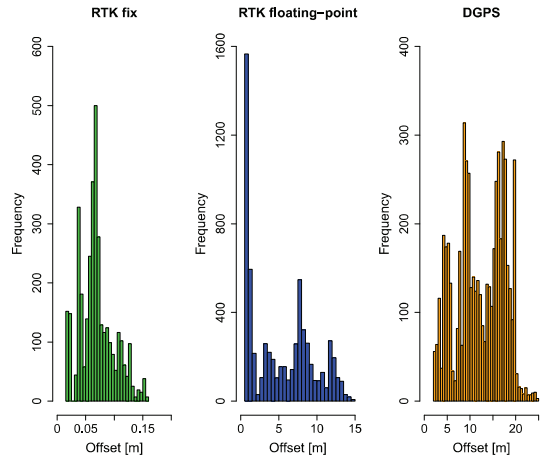


FIGURE 5. Histograms for all three GNSS correction qualities taken during the measurement campaign.

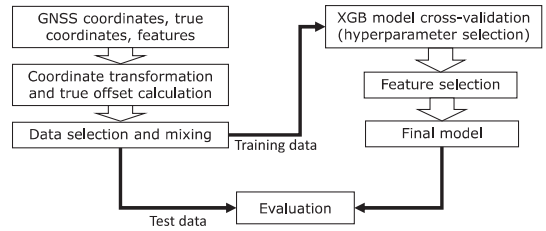


FIGURE 6. GNSS data processing flowchart.

station was set up near the test site and assigned with pre-measured geodetic coordinates. Ground truth coordinates were measured in a local frame of reference with the Leica DISTO S910 measurement tool, which had line-of-sight (LOS) with all the measurement points. The goal was to take static measurements in different environments: in obstructed, semi-obstructed, and open-sky areas. In summary, GNSS measurement data were recorded for 30 seconds at 60 different measurement points with a 10 Hz update rate. Data distribution regarding the three correction qualities is shown as histograms in Fig. 5. As expected, when GNSS RTK solves the integer ambiguities (mentioned in Section II-D), it has the best accuracy compared to other correction methods. The floating-point RTK had the largest accuracy range from approximately 1 m to 15 m. During the measurements, it was seen that indoors, at the furthest distance from the door, the main correction method was DGPS, which produced coordinates with varying offsets ranging from approximately 2 m to 25 m.

C. GNSS DATA PROCESSING AND ML MODEL TRAINING

GNSS data were processed similarly to UWB measurements and the general flowchart can be seen in Fig. 6. Collected data were extracted based on features shown in Table 1.

Also, using measurements from the Leica DISTO S910 tool, the coordinate offset was calculated and added as a response variable to the GNSS features. The purpose of the ML model was to estimate end-coordinate error based on corresponding data from GNSS features. To ensure unbiased training data, the measurements and respective response variables were sampled and divided into three equal-sized subsets, corresponding to the three correction methods: DGPS, RTK floating-point, and RTK fix. Lastly, the selected dataset was mixed and divided into segments for training and testing with a partition of 80 and 20 percent respectively.

XGBoost was the chosen ML library for model training and testing due to its proven effectiveness in previous UWB positioning applications. Its ability to deliver both accurate and computationally efficient estimations is essential for high-update-rate positioning systems [27], [41]. The algorithm employs a sequential approach, constructing a series of gradient-boosted trees. Each subsequent tree is trained to correct the errors of its predecessor, and the final prediction is a summation of all individual tree predictions. Furthermore, in recent years XGBoost has become a popular decision tree-based ensemble ML technique that has been dominating applied ML for structured or tabular data and has been successfully used for regression or classification by researchers since its release [42]. Supervised XGBoost learning was done in the RStudio environment using imported *xgboost* library [24], [43].

After data partitioning, 80 percent of the data was used for 10-fold cross-validation to select suitable hyperparameter values for the initial model. RStudio provides appropriate cross-validation *train*, *xgbTree* and *trainControl* functions with the *caret* library [44]. The training dataset, which consisted of collected features and their response variables, was separated into 10 segments with 1 segment being the validation set. This approach helps to choose more generalized hyperparameter values [45]. The main hyperparameters for XGBoost were tree depth and number of boosting iterations [46]. Using cross-validation with the *xgboost* library, different XGBoost hyperparameter values were compared in terms of prediction root mean square error (RMSE) as shown in Fig. 7. Hyperparameter values were limited, which can help avoid overfitting and an overly complex model [47]. Also, it can be seen that a model with a tree depth of 7 and 100 boosting iterations is sufficiently accurate, as choosing more than 100 iterations would present no significant increase in prediction performance.

The hyperparameters, training dataset, and response variables were then used to build the initial model with the *xgboost* function from the ML library. It should be noted that no prior feature selection was done before model cross-validation. Alternatively, XGBoost library in RStudio contains a built-in function *xgb.importance* to output features that provide the biggest informational gain in making the prediction. After establishing an initial model, inherent features were ranked in descending order of their informational

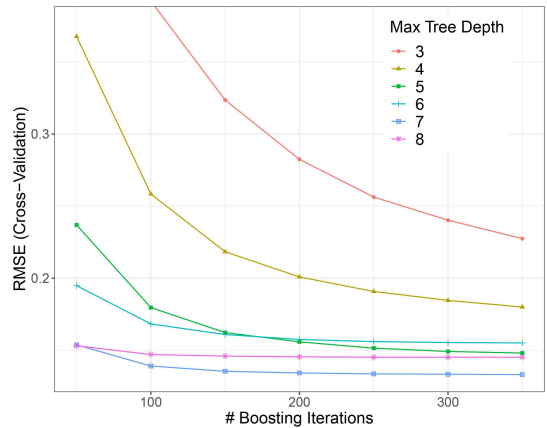


FIGURE 7. Prediction RMSE varies with different hyperparameter values, Tree depth and the number of boosting iterations were limited to 7 and 100 respectively as these values provide sufficient prediction accuracy and help avoid overfitting and an overly complex model.

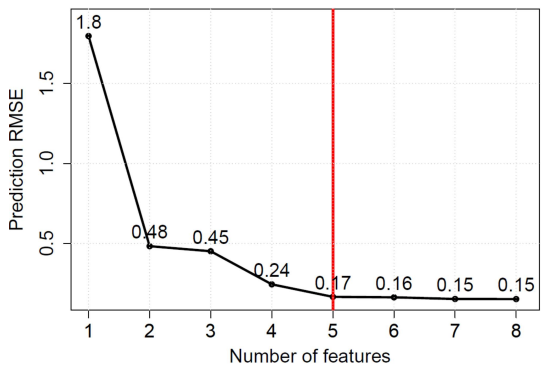


FIGURE 8. The prediction RMSE as a function of the number of features, Using more than 5 features has no significant impact on prediction accuracy and may lead to overfitting of the model.

gain. Next, by selecting a sequential combination of features, prediction RMSE was observed to select the number of features that provide a sufficiently accurate estimation. As shown in Fig. 8, more than 5 features provide no significant increase in predicting test set response values. In contrast, choosing more features may lead to overfitting and an overly complex model [48]. The final selected features and their informational gain are shown in Fig. 9.

D. GNSS ML MODEL TESTING

The final GNSS ML model was tested on a dataset, which was not used in the training process as shown in Fig. 6. Because of data mixing, the test set also contained samples with different correction qualities. As can be seen in Fig. 10, the ML model predicts coordinate offset with high accuracy. The performance was evaluated with common regression metrics: RMSE, mean square error (MSE), and

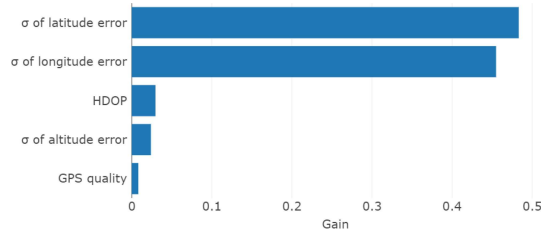


FIGURE 9. 5 features that provide the biggest informational gain in the XGBoost model. The gain quantifies how much a feature contributes in improving the models prediction.

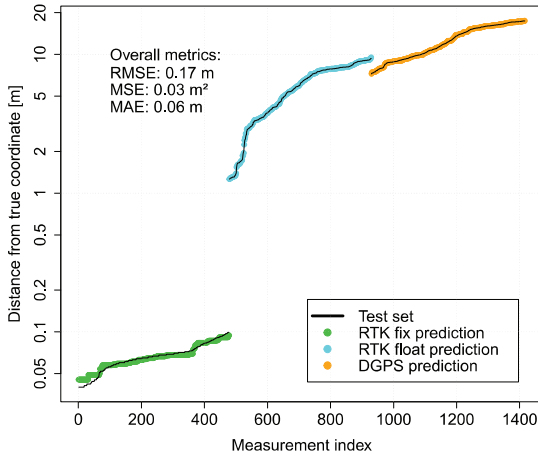


FIGURE 10. Prediction error of GNSS ML model for different correction qualities. The vertical axis is presented in the logarithmic scale.

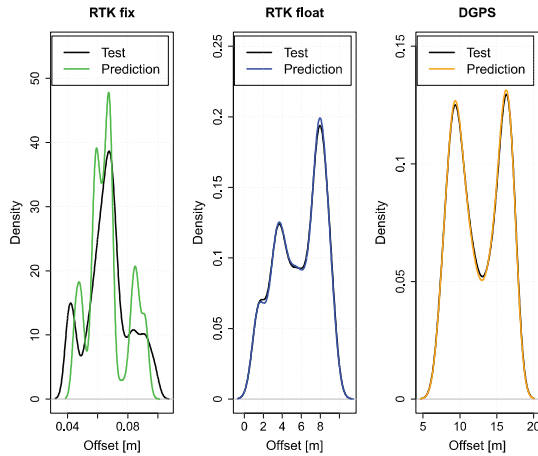


FIGURE 11. Sample density comparison of ML prediction and test set values for different correction qualities.

mean absolute error (MAE). Sample distributions of the test set and corresponding predictions can also be seen in Fig. 11.

IV. COORDINATE FUSION AND FILTERING

A. COORDINATE AND COVARIANCE FUSION

Measurement fusion is one of the most common data fusion methods in a system with complementary sensors [49]. Considering the positioning sensors used in this article, measurement fusion is done for UWB and GNSS coordinates and their covariances or uncertainty measures. Firstly, using the predicted estimate \hat{y}_U of the UWB model and \hat{y}_G from the GNSS ML model, their respective covariance matrices are established as:

$$\mathbf{R}_U = \begin{bmatrix} \hat{y}_U^2 & 0 \\ 0 & \hat{y}_U^2 \end{bmatrix}, \quad (10)$$

$$\mathbf{R}_G = \begin{bmatrix} \hat{y}_G^2 & 0 \\ 0 & \hat{y}_G^2 \end{bmatrix}. \quad (11)$$

Next, assuming measurements with normally distributed probability density functions (PDF) a joint PDF \mathbf{R}_F is calculated as:

$$\mathbf{R}_F = \left(\mathbf{R}_G^{-1} + \mathbf{R}_U^{-1} \right)^{-1}. \quad (12)$$

Lastly, estimated coordinates $\hat{\mathbf{Z}}_U$ from UWB and $\hat{\mathbf{Z}}_G$ from GNSS, their respective covariances, and fused covariance are used to produce fused coordinates as:

$$\hat{\mathbf{Z}}_k = \mathbf{R}_F \left((\mathbf{R}_G^{-1} \hat{\mathbf{Z}}_G) + (\mathbf{R}_U^{-1} \hat{\mathbf{Z}}_U) \right). \quad (13)$$

B. ADAPTIVE COORDINATE FILTERING

In this work, an adaptive Kalman filter (AKF) is used to filter end coordinate estimates with fused covariances. Considering that UWB and GNSS positioning is done in different operational environments, with dynamically changing obstructions and multipath effects for both sensors, it is required to estimate the positioning uncertainty at each coordinate update. Furthermore, GNSS positioning uncertainty is dependent on the correction quality, which also changes dynamically as was shown in Fig. 4. In essence, the ML predictions for both sensors drive the filtering process by dynamically changing measurement uncertainty, i.e., whether to trust measurement or process. In the current filtering scheme, the state transition matrix \mathbf{A} for position, velocity, and acceleration was established as:

$$\mathbf{A} = \begin{bmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} & 0 & 0 & 0 \\ 0 & 1 & \Delta t & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (14)$$

where Δt is measurement period of 0.1 s. And process noise matrix \mathbf{Q} as:

Algorithm 1 Adaptive Kalman Filter

Input: $\hat{\mathbf{X}}_0, \hat{\mathbf{Z}}_k, \mathbf{P}_0, \mathbf{Q}, \mathbf{R}_{F_k}$
Output: $\hat{\mathbf{X}}_k$
Initialize $\mathbf{A}, \mathbf{P}_0, \mathbf{H}, \mathbf{I}$
Prediction step
for $k = 1, \dots, \infty$
1: State prediction $\hat{\mathbf{X}}_k^- = \mathbf{A}\hat{\mathbf{X}}_{k-1}$
2: Covariance prediction $\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q}$
Correction step
3: Kalman gain $\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_{F_k})^{-1}$
4: State correction $\hat{\mathbf{X}}_k = \hat{\mathbf{X}}_k^- + \mathbf{K}_k (\hat{\mathbf{Z}}_k - \mathbf{H}_k \hat{\mathbf{X}}_k^-)$
5: Covariance correction $\mathbf{P}_k = \mathbf{P}_k^- (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)$
return $\hat{\mathbf{X}}_k, \mathbf{P}_k$
end for

$$\mathbf{Q} = \begin{bmatrix} \frac{\Delta t^4}{4} & \frac{\Delta t^3}{2} & \frac{\Delta t^2}{2} & 0 & 0 & 0 \\ \frac{\Delta t^3}{2} & \Delta t^2 & \Delta t & 0 & 0 & 0 \\ \frac{\Delta t^2}{2} & \Delta t & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\Delta t^4}{4} & \frac{\Delta t^3}{2} & \frac{\Delta t^2}{2} \\ 0 & 0 & 0 & \frac{\Delta t^3}{2} & \Delta t^2 & \Delta t \\ 0 & 0 & 0 & \frac{\Delta t^2}{2} & \Delta t & 1 \end{bmatrix} \sigma_a^2, \quad (15)$$

where σ_a is random acceleration standard deviation with a heuristically chosen value of 10^{-4} m/s². As the filtering is done only for the x and y coordinates, the observation matrix is set up as:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \quad (16)$$

The order of steps inside the AKF are shown in Alg. 1. The initialization coordinates $\hat{\mathbf{X}}_0$ are extracted from the sensor, which has the lower uncertainty estimate based on the respective ML model. \mathbf{P}_0 represents the initial state covariance, which was set as $\mathbf{I} \cdot 100$, with \mathbf{I} being a 6-by-6 identity matrix.

V. RESULTS

A. TEST SETUP

Tests were conducted at Tallinn University of Technology using an indoor garage with a 6-anchor UWB Eliko RTLS system. In the outdoor area a Fieldbee L2 GNSS RTK base station and a Trimble S6 total station were installed using pre-measured reference points [40], [50]. Geodetic coordinates were transformed into the local frame of reference using the ENU method as explained in Section II-C, with GNSS RTK base station at local coordinates (0, 0). The UWB tag, GNSS receiver, and reflection prism of the total station were fixed to a trolley, which also included a computer for collecting the positioning data as seen in Fig. 12. The aim was to move the test trolley from indoors to the outdoor area and return indoors while simultaneously collecting positioning data from GNSS receiver, UWB, and total station devices.

The initial total station setup was established using the resection method with three reference points [51]. These were acquired using a Trimble R12 GNSS receiver in RTK

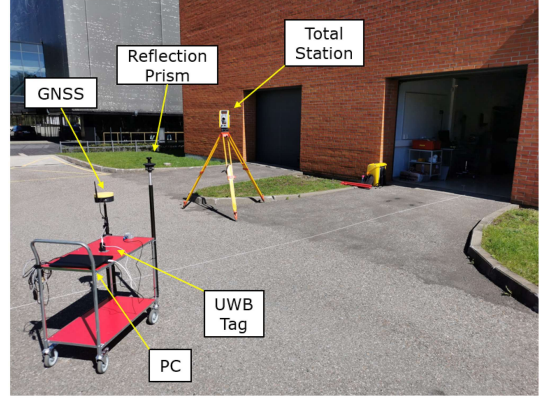


FIGURE 12. Test setup at the campus area of Tallinn University of Technology.

TABLE 2. Positioning systems used in the test.

	Eliko RTLS UWB	Fieldbee L2 GNSS RTK	Trimble S6
Accuracy	0.2 m	0.01 m + 1 ppm CEP (RTK fix)	4 mm + 2 ppm and 2" angular
Sample rate	10 Hz	10 Hz	2.5 Hz
Method	AP-TWR	Real-time kinematic	LOS with reflection prism
Coordinate system	Local	Global	Global

mode with three initializations (60 epochs per measurement). After establishing the initial base station, the three reference points were remeasured using the total station for consistent coordinates, and the reference network was further densified. The improved network formed the basis for further validation measurements. By comparing the initial Trimble R12 and total station measurements, an approximate absolute accuracy of 10 mm could be assumed for validation surveys.

As can be seen in Table 2, the total station has a sample rate of 2.5 Hz. In order to calculate correct performance metrics, sample rates of all systems must match at 10 Hz. Therefore, extra markers were added to the total station points through interpolation. Additionally, the UWB tag and GNSS receiver were positioned with an offset regarding the reflection prism. Therefore, their output coordinates were rotated and shifted to match the location of the prism. Since IMU data was not used during the test, the direction of the trolley was calculated in post-processing using interpolated points of the total station. During the test, the collected data on the two computers were included with a Unix timestamp, and the system clocks were synchronized against a time-server at nettime.pool.ntp.org.

B. INTERPRETATION OF RESULTS

An example of a test track can be seen in Fig. 13. Indoors there is poor GNSS satellite reception and only DGPS

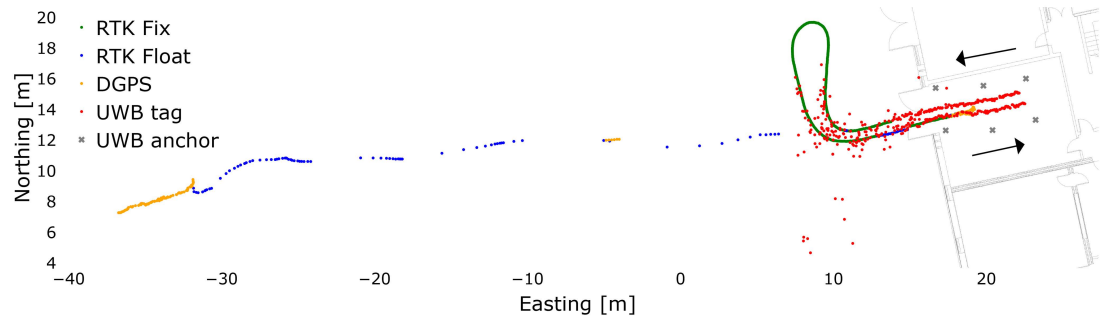


FIGURE 13. Coordinates from GNSS and UWB sensors taken along the indoor-outdoor-indoor movement path with arrows showing the movement direction. Traversing from the building, DGPS and RTK-float solutions are highly inaccurate, presenting a coordinate offset approximately 60 m from the starting point. Returning indoors, GNSS receiver fluctuates between different coordinate correction modes, while retaining a stable trajectory.

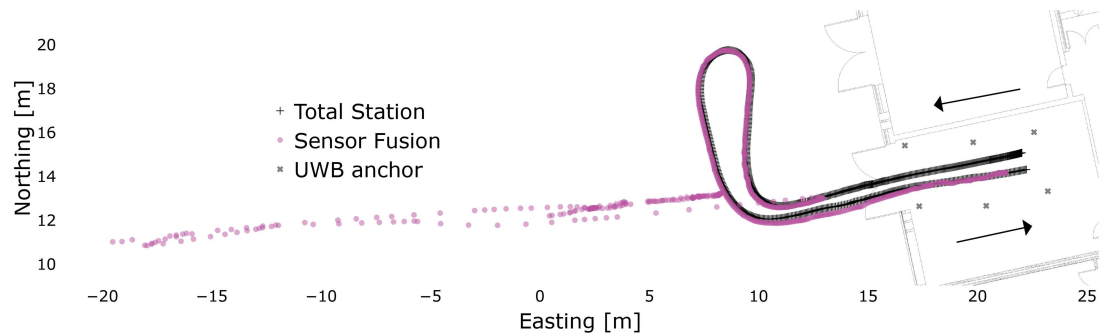


FIGURE 14. GNSS and UWB sensor fusion using HDOP-based uncertainty with AKF for benchmarking with all available GNSS corrections (DGPS, RTK float, and RTK fix).

corrections are used. This results in highly inaccurate GNSS solutions up to 60 m away. As the trolley moves out of the garage, the receiver changes into RTK floating-point mode and receives RTK fix in a few meters from the building. When the trolley approaches the building, the GNSS fluctuates between RTK fix and float modes as the satellite reception degrades. However, GNSS maintains an acceptable heading also with RTK float, which does not deviate much from the heading with the RTK fix. This test shows that RTK float, while inaccurate indoors, may contribute accurate positioning information when returning from the RTK fix state. UWB positioning worked as expected, providing accurate coordinates indoors, with small dispersion. However, the positioning quality deteriorated rapidly when moving away from the anchors.

Measurement fusion was done only with GNSS and UWB coordinates using two different methods for estimating uncertainty. The idea was to compare ML-based and HDOP-based estimation. As stated in the introduction, dilution of precision is a common measure of uncertainty. However, as it reflects only the level of geometrical uncertainty, this article proposed to use a ML model with an ensemble of features to establish a better estimate for uncertainty. By using data gathered from the test track in Fig. 13,

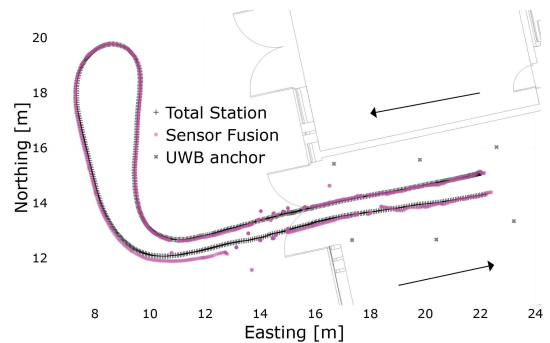


FIGURE 15. GNSS RTK and UWB sensor fusion using HDOP-based uncertainty estimation with AKF for benchmarking with RTK fix only.

GNSS and UWB measurements were fused and filtered with two different approaches for measurement uncertainties. Fig. 15 illustrates measurement fusion with HDOP-based uncertainties using UWB and GNSS with RTK fix, similar to works by Lv, Yao and Zhu [10], [13], [14]. HDOP parameter was extracted from GNSS *\$GPGGA* field 14 and UWB HDOP was calculated using pseudoranges, anchor positions

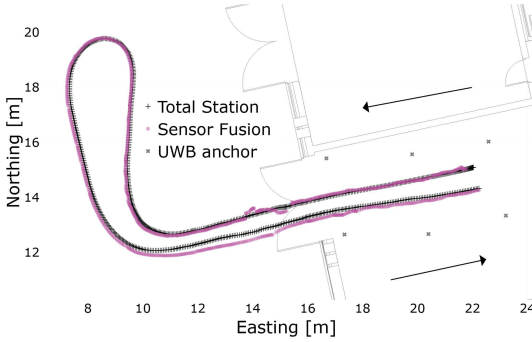


FIGURE 16. The proposed GNSS and UWB sensor fusion with ML-based uncertainty estimation with AKF using all available corrections.

and tag's estimated position. At each measurement, the \mathbf{R}_F matrix in the AKF is updated with HDOP values to give less weight to measurements with bigger geometrical uncertainty. As shown in Fig. 15 this approach has disadvantages. Firstly, as only GNSS with RTK fix is used, this fusion leads to sensor dropouts as seen near the garage door. The track has gaps when there is no UWB solution and no GNSS RTK fix. Positioning is restored when the RTK fix is achieved or when there is a UWB coordinate available. Furthermore, when all GNSS correction modes are incorporated in the fusion scheme, then HDOP does not produce a valid uncertainty estimate for RTK float and DGPS modes. As shown in Fig. 5 and Fig. 14, the coordinate offset may vary significantly, and the DOP parameter may produce a false estimate for the uncertainty. Moreover, as weighting is performed for both GNSS and UWB coordinates, it can be seen how stable indoor UWB coordinates in Fig. 13 cannot mitigate fused coordinate error due to highly inaccurate GNSS coordinates and its uncertainty estimation in Fig. 14.

On the other hand, ML-based uncertainty estimation can be applied with all GNSS corrections as the model is trained in different correction modes and conditions similar to the test area as was shown in Fig. 4. As DGPS and RTK float present solutions in a wide range, the ML model considers this information when making the uncertainty estimation. For example, at the beginning of the test, the ML model assigns a significant offset to DGPS and RTK float as the features in the model reflect information of a highly inaccurate GNSS solution. However, these points are still considered in the coordinate filtering process and are not discarded. Also, when returning from the RTK fix state, the ML model assigns lower weights to the uncertainty as it considers a more accurate RTK float solution. The result of using the ML augmented sensor fusion scheme is illustrated in Fig. 16.

C. PERFORMANCE EVALUATION

Table 3 shows overall sensor fusion results with different approaches. Following metrics were used to evaluate positioning accuracy and precision: mean location error (MLE),

TABLE 3. Comparison of different sensor fusion schemes.

	MLE [m]	RMSE [m]	Maximum error [m]
Proposed ML-based fusion with all corrections	0.16	0.18	0.49
Fusion with RTK fix and HDOP	0.14	0.19	1.29
Fusion with all corrections and HDOP	4.56	9.64	35.32

RMSE, and maximum error [52], [53]:

$$MLE = \frac{1}{n} \sum_{i=1}^n \sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2}, \quad (17)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n [(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2]}, \quad (18)$$

$$MAX = \max_{i \in n} \left(\sqrt{(x_T - \hat{x}_i)^2 + (y_T - \hat{y}_i)^2} \right). \quad (19)$$

where (x_T, y_T) are the ground truth coordinates and (\hat{x}_i, \hat{y}_i) mark the estimated fusion coordinates. In Table 3 it can be seen how the HDOP-based approach with RTK fix has an MLE and RMSE at a similar level to ML-based estimation. However, the former method suffers from sensor dropouts in the transition area, resulting in a significant maximum error. Additionally, using HDOP with all available corrections (DGPS, RTK float, and RTK fix), results in an incorrect uncertainty estimation leading to a highly inaccurate fused coordinate.

VI. CONCLUSION

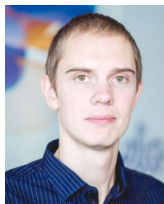
This article investigated how ML-based positioning uncertainty estimation can be used to improve indoor and outdoor positioning with sensor fusion. A key challenge for any similar solution is to estimate the uncertainty of each positioning system (GNSS or UWB) which contributes positioning data with a certain degree of integrity. While many sensor fusion solutions employ a single uncertainty parameter (e.g., DOP), this article considers a combination of features in a ML model, which helps to improve the overall uncertainty estimation. Also, this article considers the use of GNSS correction data with different degrees of quality - RTK fix, RTK floating-point, and DGPS. While these modes produce coordinates with varying accuracy and precision, this data can still be used with ML-based uncertainty estimates. By leveraging different features from GNSS NMEA messages, it was shown that the ML model can identify accurate or inaccurate GNSS corrections and apply estimated weights in a coordinate filtering scheme of an AKF. The results show that ML-based filtering outperforms the DOP-based approach. Overall, ML-based fusion had an MLE of 0.16 m, RMSE of 0.18 m, and maximum error of 0.49 m. Whereas, the DOP-based approach with RTK fix had respective values of 0.14 m, 0.19 m, and 1.29 m.

When DOP is used with all available corrections, these values are significantly higher, as DOP produces inaccurate uncertainty estimates for the GNSS in indoor and transition areas. In essence, DOP-based uncertainty estimation is not a comprehensive parameter since it only describes geometrical uncertainty. Furthermore, when applying DOP with UWB and RTK fixed mode only, there are large positioning gaps in the transition area caused by missing positioning data. On the other hand, the proposed ML model was trained in all available correction modes and different operational areas: indoors, transition areas, and outdoors. This approach makes UWB and GNSS sensor fusion with ML-based uncertainty estimation more stable and may further be improved with additional training data and complementary sensors (e.g., IMU, wheel sensor).

REFERENCES

- [1] N. El-Sheimy and Y. Li, "Indoor navigation: State of the art and future trends," *Satell. Navig.*, vol. 2, no. 1, p. 7, 2021.
- [2] A. Sesysuk, S. Ioannou, and M. Raspopoulos, "A survey of 3D indoor localization systems and technologies," *Sensors*, vol. 22, no. 23, p. 9380, 2022.
- [3] M. Elsanhoury et al., "Precision positioning for smart logistics using ultra-wideband technology-based indoor navigation: A review," *IEEE Access*, vol. 10, pp. 44413–44445, 2022.
- [4] G. Retscher, "Indoor navigation user requirements, state-of-the-art and developments for smartphone localization," *Geomatics*, vol. 3, no. 1, pp. 1–46, 2022.
- [5] D. Coppens, A. Shahid, S. Lemey, B. Van Herbruggen, C. Marshall, and E. De Poorter, "An overview of UWB standards and organizations (IEEE 802.15.4, FiRa, apple): Interoperability aspects and future research directions," *IEEE Access*, vol. 10, pp. 70219–70241, 2022.
- [6] L. Ferrigno, G. Miele, F. Milano, V. Pingerna, G. Cerro, and M. Laracca, "A UWB-based localization system: analysis of the effect of anchor positions and robustness enhancement in indoor environments," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (I2MTC)*, 2021, pp. 1–6.
- [7] S. Hayward, K. van Lopik, C. Hinde, and A. West, "A survey of indoor location technologies, techniques and applications in industry," *Internet Things*, vol. 20, Nov. 2022, Art. no. 100608.
- [8] X. Li and Y. Wang, "Research on a factor graph-based robust UWB positioning algorithm in NLOS environments," *Telecommun. Syst.*, vol. 76, pp. 207–217, Feb. 2021.
- [9] K. Zhang, C. Shen, Q. Zhou, H. Wang, Q. Gao, and Y. Chen, "A combined GPS UWB and MARG localization algorithm for indoor and outdoor mixed scenario," *Clust. Comput.*, vol. 22, pp. 5965–5974, May 2019.
- [10] P. Lv, M. Wang, B. Jin, and Y. Shen, "Seamless indoor and outdoor positioning of vehicles based on RTK/UWB," in *Proc. 6th CAA Int. Conf. Veh. Control Intell. (CVCI)*, 2022, pp. 1–8.
- [11] Y. Guo, W. Li, G. Yang, Z. Jiao, and J. Yan, "Combining dilution of precision and Kalman filtering for UWB positioning in a narrow space," *Remote Sens.*, vol. 14, no. 21, p. 5409, 2022.
- [12] A. Liu, J. Wang, S. Lin, and X. Kong, "A dynamic UKF-based UWB/wheel odometry tightly coupled approach for indoor positioning," *Electronics*, vol. 13, no. 8, p. 1518, 2024.
- [13] L. Yao et al., "GNSS/UWB/INS indoor and outdoor seamless positioning algorithm based on federal filtering," *Meas. Sci. Technol.*, vol. 35, no. 1, 2023, Art. no. 15135.
- [14] B. Zhu, X. Tao, J. Zhao, M. Ke, H. Wang, and W. Deng, "An integrated GNSS/UWB/DR/VMM positioning strategy for intelligent vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 10842–10853, Oct. 2020.
- [15] M. Specht, "Experimental studies on the relationship between HDOP and position error in the GPS system," *Metrol. Meas. Syst.*, vol. 29, no. 1, pp. 17–36, 2022.
- [16] R. Zekavat and R. M. Buehrer, *Handbook of Position Location: Theory, Practice, and Advances*. Hoboken, NJ, USA: Wiley, 2019.
- [17] M. Wang, Z. Chen, Z. Zhou, J. Fu, and H. Qiu, "Analysis of the applicability of dilution of precision in the base station configuration optimization of ultrawideband indoor TDOA positioning system," *IEEE Access*, vol. 8, pp. 225076–225087, 2020.
- [18] C. Wang, A. Xu, X. Sui, Y. Hao, Z. Shi, and Z. Chen, "A seamless navigation system and applications for autonomous vehicles using a tightly coupled gnss/uwb/ins/map integration scheme," *Remote Sens.*, vol. 14, no. 1, p. 27, 2021.
- [19] V. Di Pietra, P. Dabove, and M. Piras, "Loosely coupled GNSS and UWB with INS integration for indoor/outdoor pedestrian navigation," *Sensors*, vol. 20, no. 21, p. 6292, 2020.
- [20] A. Nessa, B. Adhikari, F. Hussain, and X. N. Fernando, "A survey of machine learning for indoor positioning," *IEEE Access*, vol. 8, pp. 214945–214965, 2020.
- [21] W. Jiang, Z. Cao, B. Cai, B. Li, and J. Wang, "Indoor and outdoor seamless positioning method using UWB enhanced multi-sensor tightly-coupled integration," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 10633–10645, Oct. 2021.
- [22] X. Li et al., "An indoor and outdoor seamless positioning system for low-cost UGV using PPP/INS/UWB tightly coupled integration," *IEEE Sensors J.*, vol. 23, no. 20, pp. 24895–24906, Oct. 2023.
- [23] T. Liu, B. Li, G. Chen, L. Yang, J. Qiao, and W. Chen, "Tightly coupled integration of GNSS/UWB/VIO for reliable and seamless positioning," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 2, pp. 2116–2128, Feb. 2024.
- [24] T. Chen et al., "XGBoost: Extreme gradient boosting: R package version 1.6.0.1." 2022. [Online]. Available: <https://CRAN.R-project.org/package=xgboost>
- [25] J. Guillory, D. Truong, and J.-P. Wallerand, "Multilateration with self-calibration: uncertainty assessment, experimental measurements and Monte-Carlo simulations," *Metrology*, vol. 2, no. 2, pp. 241–262, 2022.
- [26] P. Moravek, D. Komosny, M. Simek, and J. Muller, "Multilateration and flip ambiguity mitigation in ad-hoc networks," *Przeglad Elektrotechniczny*, vol. 88, no. 05b, pp. 222–229, 2012.
- [27] M. Tommingas, M. M. Alam, I. Mürsepp, and S. Ulp, "UWB positioning integrity estimation using ranging residuals and ML augmented filtering," *IEEE J. Indoor Seamless Position. Navig.*, vol. 2, pp. 205–218, Jun. 2024.
- [28] K. Liu and Z. Li, "Adaptive Kalman filtering for UWB positioning in following luggage," in *Proc. 34rd Youth Acad. Annu. Conf. Chin. Assoc. Autom. (YAC)*, 2019, pp. 574–578.
- [29] B. J. Silva and G. P. Hancke, "Non-line-of-sight identification without channel statistics," in *Proc. 46th Annu. Conf. IEEE Ind. Electron. Soc.*, 2020, pp. 4489–4493.
- [30] I. Guvenc, C.-C. Chong, and F. Watanabe, "NLOS identification and mitigation for UWB localization systems," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2007, pp. 1571–1576.
- [31] J. V. Sickle, *GPS for Land Surveyors*. Abingdon, U.K.: Taylor Francis Group, 2008.
- [32] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *GPS—Theory and practice*. New York, NY USA: Springer, 1997.
- [33] "GNSS logs—NovAtel documentation portal." Accessed: Sep. 19, 2024. https://docs.novatel.com/OEM7/Content/Logs/Core_Logs.htm?tocpath=Commands
- [34] A. Stateczny et al., "Study on the positioning accuracy of GNSS/INS systems supported by DGPS and RTK receivers for hydrographic surveys," *Energies*, vol. 14, no. 21, p. 7413, 2021.
- [35] E. Fredeluces, T. Ozeki, N. Kubo, and A. El-Mowafy, "Modified RTK-GNSS for challenging environments," *Sensors*, vol. 24, no. 9, p. 2712, 2024.
- [36] H. Ma, Q. Zhao, S. Verhagen, D. Psychas, and X. Liu, "Assessing the performance of multi-GNSS PPP-RTK in the local area," *Remote Sens.*, vol. 12, no. 20, p. 3343, 2020.
- [37] (Decawave Ltd., Dublin, Ireland). *DW1000 Datasheet*. Accessed: Nov. 15, 2023. <https://www.qorvo.com/products/d/da007946>
- [38] I. Dotlic, A. Connell, and M. McLaughlin, "Ranging methods utilizing carrier frequency offset estimation," in *Proc. 15th Workshop Position. Navig. Commun. (WPNC)*, 2018, pp. 1–6.

- [39] (Leica Geosyst. AG, St. Gallen, Switzerland). *Leica DISTO S910*. Accessed: Jan. 3, 2024. https://shop.leica-geosystems.com/sites/default/files/2019-04/leica_disto_s910_fol_828429_0115_en.pdf
- [40] (FieldBee eFarmer, Amstelveen, The Netherlands). *Fieldbee I2 GNSS RTK Base Station*. Accessed: Oct. 15, 2024. <https://www.fieldbee.com/wp-content/uploads/2020/09/FieldBee-L2-base-station-datasheet.pdf>
- [41] Y. Liu, L. Liu, L. Yang, L. Hao, and Y. Bao, "Measuring distance using ultra-wideband radio technology enhanced by extreme gradient boosting decision tree (XGBoost)," *Autom. Constr.*, vol. 126, Jun. 2021, Art. no. 103678.
- [42] E. K. Sahin, "Assessing the predictive capability of ensemble tree methods for landslide susceptibility mapping using XGBoost, gradient boosting machine, and random forest," *SN Appl. Sci.*, vol. 2, no. 7, p. 1308, 2020.
- [43] (R Found. Stat. Comput., Vienna, Austria). *R: A Language and Environment for Statistical Computing*. 2021. [Online]. Available: <https://www.R-project.org/>
- [44] M. Kuhn, "Building predictive models in R using the caret package," *J. Stat. Softw.*, vol. 28, no. 5, pp. 1–26, 2008. [Online]. Available: <https://www.jstatsoft.org/index.php/jss/article/view/v028i05>
- [45] H. M. Aydin, M. A. Ali, and E. G. Soyak, "The analysis of feature selection with machine learning for indoor positioning," in *Proc. 29th Signal Process. Commun. Appl. Conf. (SIU)*, 2021, pp. 1–4.
- [46] S. Demir and E. K. Sahin, "An investigation of feature selection methods for soil liquefaction prediction based on tree-based ensemble algorithms using Adaboost, gradient boosting, and XGBoost," *Neural Comput. Appl.*, vol. 35, no. 4, pp. 3173–3190, 2023.
- [47] R. G. Leiva, A. F. Anta, V. Mancuso, and P. Casari, "A novel hyperparameter-free approach to decision tree construction that avoids overfitting by design," *IEEE Access*, vol. 7, pp. 99978–99987, 2019.
- [48] J. Reunanen, "Overfitting in feature selection: Pitfalls and solutions," Ph.D. dissertation, Dept. Inf. Comput. Sci., Aalto Univ., Espoo, Finland, 2012.
- [49] A. Becker, *Kalman Filter From the Ground Up*. 2023. [Online]. Available: <https://www.kalmanfilter.net/book.html>
- [50] (Trimble, Westminster, CO, USA). *Trimble S6 Datasheet*. Accessed: Sep. 24, 2024. <https://www.kbse.com.pk/pdf/trimble-s6.pdf>
- [51] (Trimble, Westminster, CO, USA). *Resection Computations in Trimble Access*. Accessed: Oct. 4, 2024. <https://help.trimblegeospatial.com/TrimbleAccess/latest/en/PDFs/Access-Resection-Computations.pdf>
- [52] G. Laveti, G. S. Rao, K. J. Rani, A. Nalinee, and A. M. Babu, "GPS receiver SPS accuracy assessment using LS and LQ estimators for precise navigation," in *Proc. Annu. IEEE India Conf. (INDICON)*, 2014, pp. 1–5.
- [53] B. Silva and G. P. Hancke, "Ranging error mitigation for through-the-wall non-line-of-sight conditions," *IEEE Trans. Ind. Informat.*, vol. 16, no. 11, pp. 6903–6911, Nov. 2020.



MIKKEL TOMMINGAS received the higher education from Tartu Aviation College in 2008 and the M.Sc. degree in telecommunication from the Tallinn University of Technology in 2017 where he is currently pursuing the Ph.D. degree in information and communication technology. He has worked in both the military and civilian sectors. From 2008 to 2015, he was a Communications and IT officer with Estonian Air Force. From 2017 to 2021, he was a Radio Access Network Measurement Specialist with Elisa Corporation. From 2021 TO 2024, he has been worked as a Researcher with OÜ Eliko Tehnoloogia Arenduskeskus and since 2024 as an Early Stage Researcher at Tallinn University of Technology. His current research focuses on the improvement of indoor and multi-network positioning algorithms.



TAAVI LAADUNG was born in Tallinn, Estonia, in 1990. He received the B.Sc., M.Sc., and Ph.D. degrees in telecommunication from the Tallinn University of Technology, Estonia, in 2013, 2016, and 2024, respectively, where he served as a Practical Work Supervisor from 2015 to 2016. From 2017 to 2019, he worked as a Communication Systems R&D Specialist with Estonian Defence Forces. Since 2019, he has been a Researcher with Eliko Tehnoloogia Arenduskeskus OÜ, Tallinn. His research interests focus on enhancing algorithms and methods for wireless indoor tracking, positioning, and object localization systems, with an emphasis on UWB ranging protocols, particularly the active-passive two-way ranging protocol.



SANDER VARBLA received the Ph.D. degree in geodesy from the Tallinn University of Technology, Estonia, in 2023, where he is currently a Researcher with the Department of Civil Engineering and Architecture. His research interests include physical geodesy, remote sensing, and engineering surveying.



IVO MÜRSEPP received the B.Sc. (cum laude), M.Sc., and Ph.D. degrees in telecommunication from the Tallinn University of Technology (TUT) from 1998 to 2002, from 2002 to 2004, and from 2004 to 2013, respectively.

From 2002 to 2015, he was a Teaching Assistant with the Institute of Radio and Communication Technology, TUT, where he was an Associate Professor 2015 to 2018. From 2015 to 2020, he was a specialist of IT systems in cyber command with Estonian Defence Forces. Since 2018, he

has been a Senior Lecturer with the Thomas Johann Seebeck Institute of Electronics, TUT. His main research areas: mobile communications, mobile positioning, and indoor positioning.



MUHAMMAD MAHTAB ALAM (Senior Member, IEEE) received the M.Sc. degree in electrical engineering from Aalborg University, Denmark, in 2007, and the Ph.D. degree in signal processing and telecommunication from the INRIA Research Center, University of Rennes I, France, in 2013. From 2014 to 2016, he was Postdoctoral Research with the Qatar Mobility Innovation Center, Qatar. In 2016, he joined as the European Research Area Chair and as an Associate Professor with the Thomas Johann Seebeck Department of

Electronics, Tallinn University of Technology, where he was elected as a Professor in 2018 and a Tenured Full Professor in 2021. Since 2019, he has been the Communication Systems Research Group Leader. He has over 15 years of combined academic and industrial multinational experiences while working in Denmark, Belgium, France, Qatar, and Estonia. He has several leading roles as PI in multimillion Euros international projects funded by European Commission (Horizon Europe LATEST-5GS, 5G-TIMBER, H2020 5GROUTES, NATOSPS (G5482), Estonian Research Council (PRG424), and Telia Industrial Grant. He is an author and co-author of more than 100 research publications. He is actively supervising a number of Ph.D. and Postdoc Researchers. He is also a contributor in two standardization bodies (ETSI SmartBAN and IEEE-GreenICT-EECH), including "Rapporteur" of work item: DTR/ SmartBAN 0014. His research focuses on the fields of wireless communications—connectivity, mobile positioning, and 5G/6G services and applications.

Curriculum Vitae

Name: Mihkel Tommingas
Date of birth: 10.10.1984
E-mail address: mihkel.tommingas@taltech.ee
ORCID: 0000-0002-7104-4073
Current address: Tallinn

Education:

2021 - Tallinn University of Technology, Telecommunication (PhD programme)
2015 - 2017 Tallinn University of Technology, Telecommunication (Master's programme)
2008 - 2009 Speciality Officer Course
2004 - 2008 Tartu Aviation College, Speciality in Air Traffic Communications Technology
2003 - 2004 Estonian Army service
1991 - 2003 Tartu Raatuse Gymnasium

Qualifications:

2020 - 2021 Mpirical courses in communication technologies (2G/3G/4G/5G)
2013 MS Visio 2010 course
2013 Training course for the Digital Recording & Replay System (Ultra Electronics, Audiosoft)
2013 Rohde & Schwarz VCS-4G training
2012 Basics of fiber optics and fiber optic cable assembly and welding course
2012 Keymile UMUX transmission system course
2011 Rohde & Schwarz radio control & monitoring system course
2010 Aviation radio communications course
2010 Rohde & Schwarz I-Level maintenance course for XT4410A/M and series 4200 radios

Work experience:

2024 - Tallinn University of Technology, Early Stage Researcher
2021 - 2023 Eliko Location technologies, Researcher
2017 - 2021 Elisa Corporation, Communication specialist in Radio Access Network measurements and indoor solutions
Responsibilities: Specializing in measuring outdoor and indoor radio coverage. Especially in accepting of new GSM, UMTS and LTE radio sites. Planning of new indoor solutions to improve mobile radio quality. Also dealing with clients in locations with poor radio reception and finding optimal solutions for the client and the company.
2014 - 2015 Estonian Air Force, Air Force Staff, Staff Officer of Communications Department
2011 - 2014 Estonian Air Force, Air Surveillance Wing, Chief of Communications and IT Department
2009 - 2011 Estonian Air Force, Air Surveillance Wing, Communication officer

2008 - 2009 Estonian Air Force, Air Surveillance Wing, Communication technician Responsibilities: Management of the Communications and IT Department. Dealing with on-going projects. Managing routine system maintenance procedures and repairing system-specific equipment. Delegating everyday work to subordinates. Acting as a consultant and cooperating with other structures.

Languages:

Estonian	Native
English	Very good
German	Good
Russian	Average

IT skills: Windows OS, MS Office, audio-video processing, R, Matlab, Python

Defences and supervisions:

2017, Distinguishing of Anthropogenic and Natural Hydroacoustic Noise, Tallinn University of Technology, MSc, supervisor Julia Berdnikova, Mirko Mustonen, Tallinn University of Technology, School of Information Technologies, Thomas Johann Seebeck Department of Electronics

Swaminathan Ramachandran, MSc, 2022, supervisor Mihkel Tommingas, Heigo Mölder, Indirect Tracking of Objects with GNSS, Tallinn University of Technology, School of Information Technologies, Thomas Johann Seebeck Department of Electronics

Scientific work:

M. Tommingas, S. Ulp, M. M. Alam, I. Mürsepp, and T. Laadung, "Estimating UWB Positioning Integrity Based on Ranging Residuals," in *2023 24th International Conference on Applied Electromagnetics and Communications (ICECOM)*, pp. 1–5, IEEE, 2023

M. Tommingas, M. M. Alam, I. Mürsepp, and S. Ulp, "UWB Positioning Integrity Estimation Using Ranging Residuals and ML Augmented Filtering," *IEEE Journal of Indoor and Seamless Positioning and Navigation*, vol. 2, pp. 205–218, 2024

M. Tommingas, T. Laadung, S. Varbla, I. Mürsepp, and M. Mahtab Alam, "UWB and GNSS Sensor Fusion Using ML-Based Positioning Uncertainty Estimation," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 2177–2189, 2025

Mustonen, M., Klauson, A., Berdnikova, J., Tommingas, M. (2017, December). "Assessment of the proportion of anthropogenic underwater noise levels in passive acoustic monitoring". In *Proceedings of Meetings on Acoustics* (Vol. 31, No. 1). AIP Publishing. (co-author)

Berdnikova, J., Klauson, A., Mustonen, M., Tommingas, M. (2017). "Underwater ship noise pattern detection and identification". *The Journal of the Acoustical Society of America*, 142(4), 2686–2686. (co-author)

Elulookirjeldus

Nimi: Mihkel Tommingas
Sünniaeg: 10.10.1984
E-mail: mihkel.tommingas@taltech.ee
ORCID: 0000-0002-7104-4073
Elukoht: Tallinn

Haridus:

2021 - Tallinna Tehnikaülikool, telekommunikatsioon (doktorantuur)
2015 - 2017 Tallinna Tehnikaülikool, telekommunikatsioon (magistrantuur)
2008 - 2009 Eriala-ohvitseri kursus
2004 - 2008 Tartu Lennuakadeemia, side-ja navigatsioonisüsteemide käitamine
2003 - 2004 Ajateenistus
1991 - 2003 Tartu Raatuse Gümnaasium

Kvalifikatsioonid:

2020 - 2021 Mpirical telekommunikatsiooni kursused (2G/3G/4G/5G)
2013 MS Visio 2010 kursus
2013 Digitaalsete salvestusseadmete käitamise kursus (Ultra Electronics, Audiosoft)
2013 Rohde & Schwarz VCS-4G kursus
2012 Fiiberoptika seadmete ja käitamise baaskursus
2012 Keymile UMUX transmissioonisüsteemide käitamine
2011 Rohde & Schwarz raadiote kontrolli ja monitooringu kursus
2010 Lennunduse raadiote käitamise kursus
2010 Rohde & Schwarz I-taseme hoolduse kursus

Töökogemus:

2024 - Tallinna Tehnikaülikool, nooremteadur (doktorant)
2021 - 2023 Eliko Tehnoloogia Arenduskeskus OÜ, teadur
Vastutus: Metoodiliste ja eksperimentaalsete tööde läbiviimine vastavalt projektiplaanile. Tulemuste analüüs ja esitamine projektijuhile ning programmijuhile. Koostöö arendusinseneridega. Töö tulemuste tutvustamine ettevõtte sees ning firmavälistel üritustel.
2017 - 2021 Elisa Eesti AS, siselahenduste ja mõõtmiste spetsialist
Vastutus: Väli- ning siselevi raadiomõõtmiste teostamine. Tugijaamade raadioparameetrite kontroll (GSM/UMTS/LTE). Uute siselahenduste planeerimine raadiolevi parendamiseks. Kliendisuhetus uute objektide planeerimisel.
2014 - 2015 Eesti Õhuvägi, Õhuväe staap, staabiohvitser
2011 - 2014 Eesti Õhuvägi, Õhuväe staap, side- ja IT lüli ülem
2009 - 2011 Eesti Õhuvägi, Õhuväe staap, sideohvitser

2008 - 2009 Eesti Õhuvägi, Õhuväe staap, sidetehnik
Vastutus: Side- ja IT lüli tegevuste planeerimine ja teostamine. Uute projektide planeerimine ja teostus. Sideseadmete hoolduse ja protseduuride haldamine. Meeskonna juhtimine ja tegevuste koordineerimine teiste osakondadega.

Keeled:

Eesti	Emakeel
Inglise keel	Väga hea
Saksa keel	Hea
Vene keel	Keskmine

IT oskused: Windows OS, MS Office, audio-video töötlus, R, Matlab, Python

Publikatsioonid:

Loetelu toodud ingliskeelse elulookirjelduse juures.

Kaitstud lõputööd ja juhendamised:

Mihkel Tommingas, magistrikraad, 2017, (juh) Julia Berdnikova; Mirko Mustonen, Tehislike ning looduslike komponentide leidmine hüdroakustilises müras, Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Thomas Johann Seebecki elektroonikainstituut

Swaminathan Ramachandran, magistrikraad, 2022, (juh) Mihkel Tommingas; Heigo Mölder, Indirect tracking of objects with GNSS (Objektide kaudne jälgimine GNSS-iga), Tallinna Tehnikaülikool, Infotehnoloogia teaduskond, Thomas Johann Seebecki elektroonikainstituut

ISSN 2585-6901 (PDF)
ISBN 978-9916-80-308-0 (PDF)