

Self-Learning Detection and Mitigation of Non-Line-of-Sight Measurements in Ultra-Wideband Localization

Laura Flueraș^{*†}, Elena Simona Lohan[†] and Dragoș Niculescu^{*}

^{*} Computer Science Department, University Politehnica of Bucharest, Romania

[†] Electrical Engineering Unit, Tampere University, Finland

Abstract—Non-line-of-sight (NLOS) propagation is one of the main error sources in indoor localization, so a large body of work has been dedicated to identifying and mitigating NLOS errors. The most accurate NLOS detection methods often rely on large training data sets that are time-consuming to obtain and depend on the environment and hardware. We propose a method for detecting NLOS distance measurements without manually collected training data and knowledge of channel statistics. Instead, the algorithm generates LOS/NLOS labels for sets of distance measurements between fixed sensors and the mobile target based on distance residuals. The residual-based detection has 70–80% accuracy but has high complexity and cannot be used with high confidence on all measurements. Therefore, we use the predicted labels and the channel impulse responses of the measurements to train a classifier that achieves over 90% accuracy and can be used on all measurements, with low complexity. After we train the classifier during an initial phase that captures specifics of the devices and of the environment, we can skip the residual-based detection and use only the trained model to classify all measurements. We also propose an NLOS mitigation method that reduces, on average, the mean and standard deviation of the localization error by 2.2 and 5.8 times, respectively.

Index Terms—Non-line of sight (NLOS), localization, positioning, ultra-wideband (UWB), machine learning

I. INTRODUCTION

Indoor localization has garnered attention in recent years for its useful applications such as navigating in public spaces, offering customized location-based services and interactions with the environment, or controlling and monitoring industrial robots and indoor drones. One popular localization method uses several sensors (called anchors) with fixed and known positions that communicate with a mobile target (called tag). The method estimates the location of the tag based on time or distance measurements between each anchor and the tag. Non-line-of-sight (NLOS) propagation, in which an object or a person obstructs the direct path between two devices, affects most localization methods. In this case, the observed time of

flight (TOF) or distance between two devices is larger than without the obstacle, which causes a localization error.

Detecting and correcting NLOS distance errors has been widely studied in the literature [1], [2], [3], [4], [5], [6]. The methods with the highest detection accuracy rely on the statistics of the channel impulse response (CIR) of the signal [2], [3], [4]. However, they require extensive measurement campaigns to learn the statistics of LOS and NLOS measurements. Such measurements are time-consuming, require some expertise, depend on the environment and on the used hardware, and need to be repeated frequently in order to capture the environment dynamics. Therefore, collecting training data before every deployment and maintaining databases up-to-date are demanding tasks, which are usually infeasible in practice.

In this work, we propose an NLOS detection and mitigation method that does *not* require manually-acquired training data nor channel statistics. Fig. 1 shows the main steps of our approach. When a tag is first deployed in an area, it starts the initial phase of the algorithm, in which it collects distance measurements and CIRs from all the anchors in the area. In this first step, the measurements are labeled as LOS, NLOS, or ambiguous using *residual analysis*. If there are more anchors than the minimum necessary for 2D or 3D localization, we can compute locations using each subset of anchors. Since NLOS anchors¹ introduce higher location errors, they also have higher distance residuals, defined as the difference between the measured distance and the Euclidean distance between the anchor and the estimated position of the tag. The method labels anchors as LOS/NLOS if their average residuals can be grouped in two one-dimensional clusters (intervals). The labeling step has an acceptable accuracy (70–80%) but it cannot be used on all sets of measurements, since for small NLOS errors the separation between LOS and NLOS residuals is not clear. Moreover, the residual-based labeling step requires computing the location using all anchor combinations, which scales with $\mathcal{O}(2^N)$ for N anchors.

Therefore, we introduce the second step, in which we train a Random Forest (RF) classifier using the labels predicted in the first step and the CIR features of the measurements. The model can recover the correct class boundary even with noisy labels, reaching a higher classification accuracy (>90%)

This work was supported by funding from European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska Curie grant agreement No. 813278 (A-WEAR: A network for dynamic wearable applications with privacy constraints, <http://www.a-wear.eu/>). The work was also partly supported by a grant from the Romanian National Authority for Scientific Research and Innovation, UEFISCDI project PN-III-P3-3.6-H2020-2020-0124.

978-1-6654-0402-0/21/\$31.00 © 2021 IEEE

¹“NLOS anchor” is shorthand for “anchor in NLOS with the tag.”

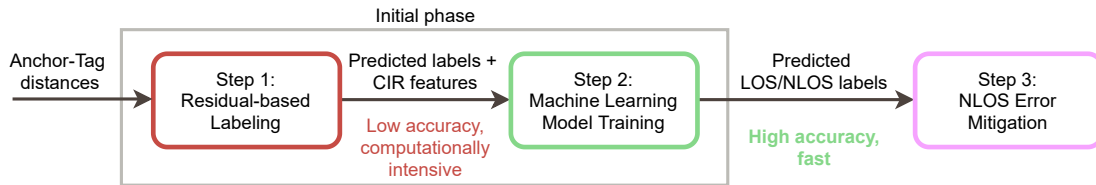


Fig. 1. Flowchart of the proposed method for the detection and mitigation of NLOS distance measurements.

than the residual-based labeling. After the model is trained, we can directly classify all distance measurements as LOS/NLOS using the RF and skip the residual-based labeling. Classifying samples using RF has a constant complexity and can be used in all localization instances.

We further propose a location-correction method based on identified LOS/NLOS measurements which does not discard NLOS measurements. We evaluate the accuracy and localization error of the proposed method on a database of measurements acquired with UWB devices, which therefore resembles a real localization setup.

II. RELATED WORK

NLOS identification methods based on channel statistics have been well studied in literature [2], [3], [4] and they achieve very high classification accuracy ($>90\%$). However, they need extensive measurement campaigns to collect training data, which are rarely feasible in practice.

NLOS errors can be directly mitigated using semi-definite programming [5], [7] without assuming any measurement statistics. However, these methods are usually more computationally expensive than plain localization algorithms [5]. Other methods that do not require error statistics use additional hardware, such as inertial measurement units [8]. In [9], the authors proposed a NLOS mitigation technique for dense NLOS environments that does not need training but it assumes the measurement variance to be known. The error was corrected using two extended Kalman filters (EKFs) alternatively depending on the LOS/NLOS condition.

In [10], the authors propose an unsupervised NLOS identification method. The biggest difference from our approach is that the method in [10] can classify data only in bulk (so not online) because it needs a collection of data points to obtain the distribution of features with Gaussian mixture models.

In [11] and [12], the authors proposed NLOS identification methods based on pre-trained models. In [11], a convolutional neural network (CNN) is trained in one environment and updated with data from a new environment. The method is validated in two similar office environments, so it is not clear how well the model can be transferred between two very different environments, e.g., a mall and an office. This is also an open research question for our method. However, we do not rely on a pre-trained model but can train it online. In [12], a pre-trained model is improved by retraining using unlabeled samples. This approach can be used to improve the accuracy of our model (discussed in Section VI).

The closest work to ours is [6], where the authors used anchor residuals instead of CIR features to train a classifier. The simulated error for NLOS measurements was sampled from a uniform distribution between 0.75–3.5 m, so NLOS measurements were easily distinguishable from LOS ones, which had zero-centered normally-distributed errors. However, in our measurement campaigns (described in Section IV), we found that typical NLOS errors with UWB devices are spread between a smaller range of 0.25–0.8 m, so it was harder to accurately identify NLOS measurements using only anchor residuals. Therefore, we propose a classification in two steps: first using anchor residuals, then using CIR features.

Another work that applied residual analysis to identify NLOS errors is [13]. The authors used residual analysis to identify NLOS errors, a voting algorithm to correct these errors, and a fuzzy C-means algorithm to classify NLOS errors into “hard” and “soft” NLOS. A Kalman filter (KF) and an unscented Kalman filter (UKF) filtered the two types of NLOS errors and corrected the location estimates. The authors in [13] focused more on NLOS error mitigation than on NLOS identification and they did not mention the accuracy of the classification method alone. Compared to [13], we also provide a NLOS detection method which can be useful in detecting obstacles, creating building maps, or estimating crowd densities. We show that we can also reduce localization errors with our NLOS detection method.

III. NON-LINE-OF-SIGHT DETECTION

In this section, we present the basics of anchor-based localization (Section III-A) and the main steps of the unsupervised NLOS detection method, also highlighted in Fig. 1. First, we use residual analysis to obtain LOS/NLOS labels using only distance measurements (Section III-B). By repeating this step for multiple locations, we create a database of CIR features and their predicted labels. These are given as training data to an RF classifier (Section III-C). Once trained, the classifier can directly classify subsequent measurements.

A. Localization

Range-based localization estimates the coordinates \hat{x} of a target (also called a tag) using the distance measurements between the tag and N anchors with known locations x_{A_i} , where A_i is the i^{th} anchor, for $i = 1, \dots, N$. When the direct path between two devices is unobstructed, also known as line-of-sight (LOS) propagation, the distance between two devices can be recovered as $d = c \cdot \text{TOF}$, where c is the speed of light. If an object or person blocks the direct path, the

signal usually travels at a lower speed than through the air. This causes a larger TOF than without the obstacle and the estimated distance is larger than in reality. Obstacles can also completely block the direct signal; in such cases, reflections which travel longer paths than the direct signal can arrive at the receiver and also cause time delays. The last two scenarios are known as non-line of sight (NLOS) and cause distance and localization errors in UWB localization.

The anchor–tag distances can be written as:

$$d_i = \|\mathbf{x}_{A_i} - \mathbf{x}\| + v_i, \quad i = 1, \dots, N \quad (1)$$

where \mathbf{x} is the true location of the tag, $\|\cdot\|$ is the Euclidean norm, and v_i is a noise term. In vector form, this becomes:

$$\mathbf{y} = h(\mathbf{x}) + \mathbf{v}, \quad (2)$$

where \mathbf{y} is a vector which contains all distance measurements $d_i, i = 1, \dots, N$, \mathbf{v} is the error vector, and h is a vector-valued measurement function.

The location can be found using the least squares method:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{y} - h(\mathbf{x})\|. \quad (3)$$

We chose the regularized Gauss-Newton multilateration algorithm from [14], because it has low computational complexity and localization errors comparable to closed-form solutions. The algorithm needs an initial location, which can be obtained with a closed-form solution. For each iteration k , the algorithm computes the Jacobian matrix:

$$J_k(\mathbf{x}) = \begin{bmatrix} \frac{\mathbf{p}_1 - \mathbf{x}}{\|\mathbf{p}_1 - \mathbf{x}\|}, \dots, \frac{\mathbf{p}_N - \mathbf{x}}{\|\mathbf{p}_N - \mathbf{x}\|} \end{bmatrix}^T. \quad (4)$$

The solution at each iteration is $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta\mathbf{x}_k$ and $\Delta\mathbf{x}_k$ is the least-squares solution to:

$$-(\Sigma^{-\frac{1}{2}} J_k + c \mathbf{I}_N) \Delta\mathbf{x}_k = (\Sigma^{-\frac{1}{2}} (h(\mathbf{x}_k) - \mathbf{d}) + c(\mathbf{x}_k - \mathbf{x}_r)), \quad (5)$$

where \mathbf{I}_N is the unitary matrix of size $N \times N$, \mathbf{x}_r is a regularization location and c is a regularization coefficient equal to the inverse of the standard deviation of a distribution centered at \mathbf{x}_r . The algorithm stops when the norm of the incremental location is smaller than the tolerance δ or when it reaches the maximum number of iterations k_{max} .

B. Unsupervised Labeling with Anchor Residuals

If the distance measurements d_i are noiseless, i.e. $v_i = 0$ in Eq. (1), then in the 2D case the tag is found at the intersection between circles centered at the anchors, with a radius equal to d_i , for $i = 1, \dots, N$ (see Fig. 2a). If the distance measurements are noisy, the circles do not intersect in a single point anymore and the tag's location is (ideally) found inside the intersection area of the circles, as shown in Fig. 2b. The *residual* of anchor A_i is defined as [1]:

$$r_i = d_i - d'_i, \quad (6)$$

where d_i is the measured distance between anchor A_i and the tag and d'_i is the distance between the anchor and the estimated location $\hat{\mathbf{x}}$ of the tag. Intuitively, an anchor's residual is likely

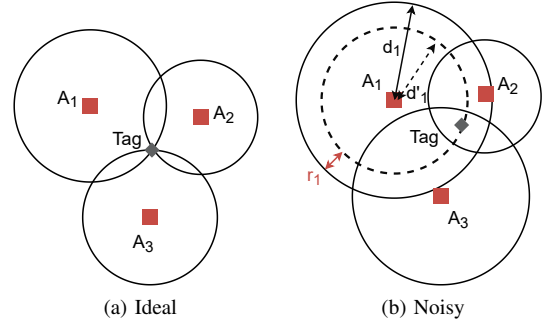


Fig. 2. In the ideal (2D) case, the tag is found at the intersection point between the circles centered at the anchors' locations with the radius equal to each anchor–tag distance. When the measured distances are noisy, the circles do not intersect in a single point anymore. An anchor's residual r_i is the difference between the measured distance d_i and the distance between the estimated location and the anchor's location d'_i .

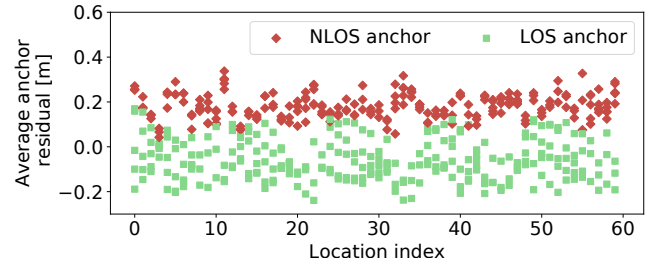


Fig. 3. Average anchor residuals, computed over all subsets in which an anchor is used. The residuals of LOS and NLOS anchors are linearly separable in many cases.

to be higher when the anchor is in NLOS with the tag, since this causes a higher distance measurement error [1].

For M -dimensional localization, at least $M + 1$ anchors are needed to solve the system of equations from Eq. (2). If N anchors with $N > M + 1$ are available, we can form subsets of K anchors, where $M + 1 \leq K \leq N$, and compute the residual of a subset S as the mean of the squared residuals of all anchors:

$$R_S = \frac{1}{|S|} \sum_{i=1}^{|S|} r_{i,S}^2, \quad (7)$$

where $|\cdot|$ denotes the cardinality of a set and $r_{i,S}$ is the residual of the i^{th} anchor in this subset. In [1], the final location estimate is obtained as the weighted average of the intermediate locations obtained in all subsets, where the weight is the inverse of a subset's residual.

Starting from the insight that anchor residuals are higher when anchors are in NLOS with the tag, we propose using the average residual of an anchor, computed over all subsets in which it is used, to identify NLOS anchors:

$$R_i = \frac{1}{|\{S | A_i \in S\}|} \sum_{\{S | A_i \in S\}} r_{i,S}. \quad (8)$$

Because NLOS anchors have higher distance measurement errors, their average residuals are also usually higher than those of LOS anchors. In many cases, residuals coming from LOS and NLOS anchors form two 1D clusters (or intervals)

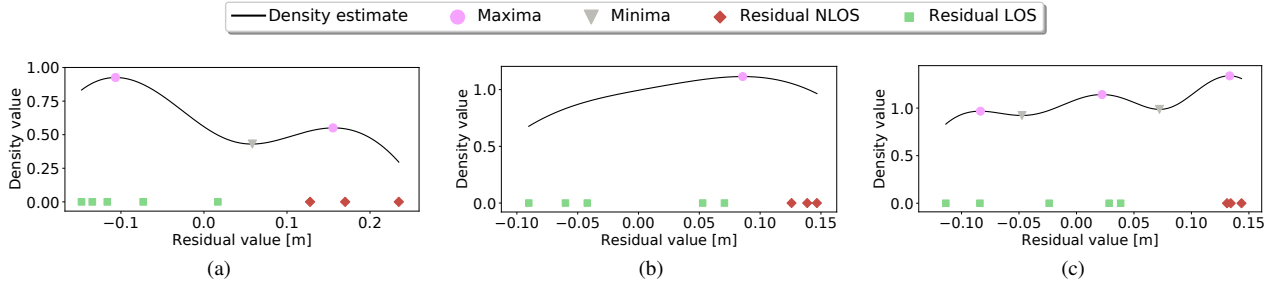


Fig. 4. Examples of kernel density estimation (KDE) applied on anchor residuals for the shaping parameter $h = 0.08$.

which can be separated. Fig. 3 shows such an example, for a simulation of eight anchors when the tag and some anchors are separated by a wall. The wall introduces a lognormally-distributed error with a median of 24 cm and a standard deviation of 1.8 m (the parameters were obtained from a measurement campaign [15]). In many cases, the average residuals of LOS and NLOS anchors can be clearly delimited.

To find the threshold that separates LOS from NLOS residuals, we use kernel density estimation (KDE) to get the distribution of the average anchor residuals (for a given location). The kernel density estimator of a series of independent and identically distributed samples $\{R_1, \dots, R_N\}$ is:

$$\hat{f}_h(R) = \frac{1}{Nh} \sum_{i=1}^N K\left(\frac{R - R_i}{h}\right), \quad (9)$$

where K is a non-negative function called the kernel (we used a Gaussian kernel) and $h > 0$ is a smoothing parameter. Fig. 4 shows three examples of KDE applied on anchor residuals, for $h = 0.08$. When LOS and NLOS anchor residuals form two different clusters, the distribution has two maxima and a single minimum, like in Fig. 4a. In this case, we label anchors whose residual is higher than the minimum as NLOS and the rest as LOS. There can also be ambiguous cases. In Fig. 4b, the residuals are uniformly spread over the interval and the distribution has only one maximum. In Fig. 4c, on the other hand, there are more than one local minima found.

The parameter h determines the smoothness of the fitted distribution. If h is too small, the estimated distribution will contain spurious data artifacts, similar to the distribution in Fig. 4c. If h is too high, the estimator cannot capture the underlying data structure, leading to an estimate similar to Fig. 4b. Therefore, too large or too small of a smoothing parameter leads to ambiguous cases in which the data cannot be labeled. Because we aim for a high LOS/NLOS detection accuracy, we label only unambiguous cases in which the distribution has exactly one minimum. In the current implementation, we use the ambiguous cases only for validation but in the future we could use them to retrain the RF model and increase its accuracy (see the discussion in Section VI). In Section III-B, we will discuss the choice of h which maximizes the labeling accuracy and the percentage of instances classified.

Because not all anchor residuals can be unambiguously split into two intervals, we cannot apply the residual-based labeling on all measurements. Also, we need to compute locations

using all anchor subsets, which scales with 2^N . Therefore, we use the labels provided by the residual method to train an ML model that can classify *all* distance measurements.

C. Model Training

The identification and mitigation of the LOS/NLOS condition using features based on the CIR of the signal has been studied in [2], [3], [4]. Supervised classification methods have very high accuracy ($> 90\%$) but need training data, i.e., a database of distance measurements labeled as LOS/NLOS and their CIRs. We replace the manual labeling with the unsupervised labeling based on anchor residuals, as described in Section III-B. Because our training set has label noise (LOS measurements labeled as NLOS and vice-versa), we want to train a machine learning model robust to label noise. We chose a Random Forest (RF) classifier, since it is an ensemble machine learning (ML) algorithm that performs well with noisy labels [16]. The RF is a collection of decision trees which outputs the class predicted by most of the individual decision trees through bootstrap aggregating.

The purpose of this paper is not to find the best ML algorithm for the task, but to demonstrate the general idea, that we can detect NLOS measurements without training data using residual-based labeling. We leave as future work an exhaustive search through more ML models suitable for data sets with noisy labels that further increase the NLOS detection accuracy.

We train the model using CIR features known to characterize well LOS/NLOS conditions [2], [3]: the energy of the received signal, the maximum amplitude of the signal, the mean excess delay, the RMS delay spread, the kurtosis, and the difference between the TOA and the time at which the signal has the maximum amplitude ($\Delta T(\text{TOA}, \text{Max})$).

IV. EVALUATION SETUP

We simulate a localization scenario based on a database of real UWB measurements to evaluate the feasibility and performance of the proposed method for NLOS error detection and correction. Fig. 5 describes the simulation flow. We start from a setup with an area of 9×20 m and $N \in \{5, 6, 7, 8, 9\}$ anchors distributed approximately uniformly on the perimeter of the area. When N is odd, one anchor is in the center of the area, while the others are on the perimeter. We consider a grid of ≈ 1700 true locations of the tag spaced 20 cm apart within the area encompassed by the anchors. In a real deployment

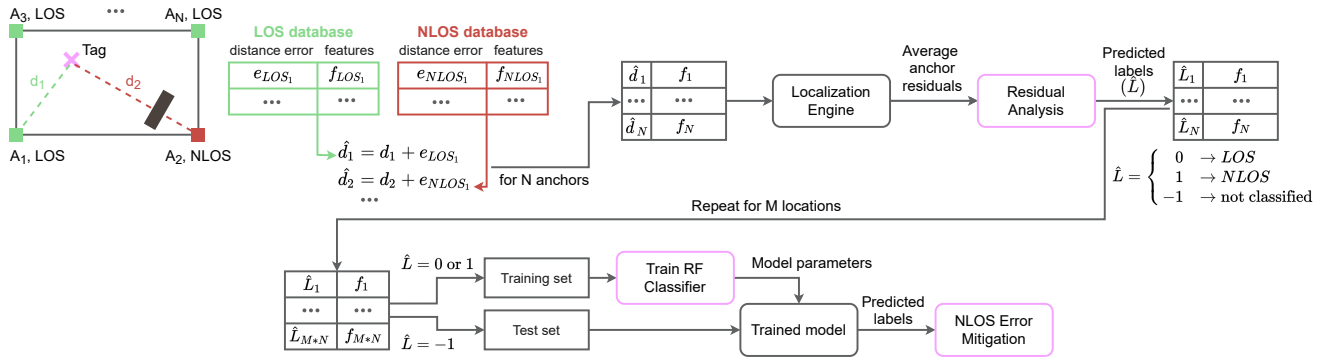


Fig. 5. Flowchart of the simulation framework for NLOS detection and mitigation.

with a reasonable location update period of 100 ms, 1700 locations could be obtained in under 3 min. For each true location of the tag, we choose at random $\lfloor N * Q \rfloor$ anchors to be in NLOS with the tag, where $Q \in \{0, 0.3, 0.5\}$.

We simulate the distance measurements between each anchor and the tag by adding to the true distance a distance error selected from a measurement database, based on whether the anchor is in LOS or NLOS with the tag. We also store the CIR corresponding to the selected measurement. Although we could simulate the distance errors for each obstacle based on proposed models [15], it is harder to simulate the CIR for a particular type of obstruction. Therefore, for a realistic setup, we preferred selecting the distance error and CIR of a real measurement from a database. We use a database of distance measurements and CIRs acquired with UWB devices developed by 3db Access, which was partly described in [15]. The measurements were acquired in LOS, in NLOS with human body shadowing, or in NLOS with concrete wall shadowing at various indoor locations. The NLOS database aggregates the measurements with both types of obstructions. TABLE I shows the number of measurements acquired in each scenario and the range of covered distances.

We feed the N distances to a localization engine, which computes the 2D location and average anchor residuals over all anchor subsets. For localization, we used the Gauss-Newton multilateration algorithm strengthened with a regularization term. We initialized the algorithm with $\delta = 1$ mm, $k_{max} = 10$ iterations, $\mathbf{x}_r =$ the median of the anchors' locations, and $c = 10^{-1}$ (corresponding to a standard deviation of 10 m around \mathbf{x}_r , suitable for our setup).

The residual analysis block receives as input the average anchor residuals and predicts the labels of each anchor-tag measurement. The label can be either LOS, NLOS, or ambiguous (in case the density of the anchor residuals does not have exactly one minimum). We repeat the procedure for M locations and build a database of $M * N$ predicted labels and the corresponding CIR features.

We split the database into a training set, which contains the measurements predicted as LOS/NLOS, and a test set, which contains the ambiguous measurements. We use the training set to train a Random Forest classifier, which learns

TABLE I
DESCRIPTION OF THE DATABASE WITH DISTANCE MEASUREMENTS.

Scenario	Distances	Number of measurements
LOS	1–8 m	16 814
NLOS with concrete wall	1–10.5 m	13 770
NLOS with human body	1–10 m	8462

the LOS/NLOS CIR features based on the labels predicted by the residual analysis. Once the model is trained, we can use it to directly classify all measurements, without going through the residual analysis procedure. Finally, we mitigate NLOS measurements and reduce the localization error.

V. EVALUATION

We now evaluate the performance of the residual-based labeling (Section V-A), of the trained RF classifier (Section V-B), and of the NLOS mitigation method (Section V-C).

A. Residual-based LOS/NLOS Labeling

The first step is the unsupervised labeling using anchor residuals, described in Section III-B. We can alternatively formulate the labeling as a detection problem, where the detected event is a NLOS measurement. To evaluate the performance of the labeling method, we use the balanced accuracy, which is the arithmetic mean of the true positive and true negative rates (TPR and TNR, respectively):

$$\text{Balanced accuracy} = (\text{TPR} + \text{TNR}) / 2, \quad \text{where} \quad (10)$$

$$\text{TPR} = \text{TP} / (\text{TP} + \text{FN}), \quad \text{and} \quad (11)$$

$$\text{TNR} = \text{TN} / (\text{TN} + \text{FP}). \quad (12)$$

TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives.

Because we label only cases where the density estimate of the anchor residuals has exactly one minimum, we are also interested in the percentage of instances labeled (denoted by P_C), defined as the number of locations for which the algorithm provides a label for all anchors. This ratio and

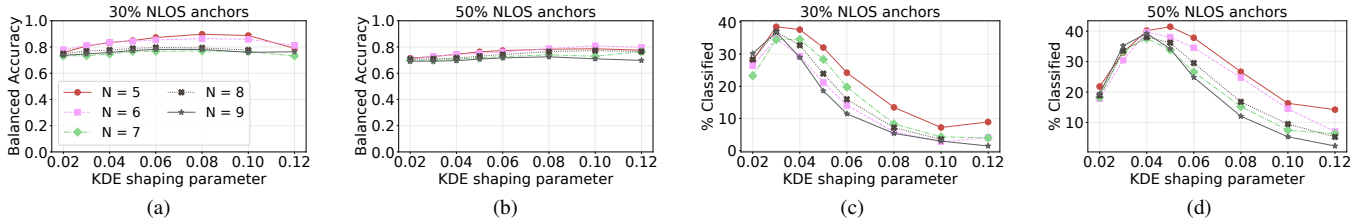


Fig. 6. The balanced accuracy and percentage of classified instances of the residual-based labeling as a function of the KDE shaping parameter for 30 % and 50 % NLOS anchors out of $N \in \{5, 6, 7, 8, 9\}$ anchors.

the accuracy depend on the KDE shaping parameter (h). If h is too small or too large, the estimation is oversmoothed or undersmoothed, respectively, resulting in few classified instances (because the distribution has either too many or no minima at all). Therefore, we want to find the KDE shaping parameter which maximizes the accuracy and P_C .

Fig. 6 shows the balanced accuracy and percentage of classified instances against h for 30 % and 50 % NLOS anchors. We omitted the case when all anchors are in LOS with the tag because less than 1 % instances are classified in this case. This is desirable for the RF classifier because it keeps the number of LOS and NLOS measurements balanced.

Fig. 6a and 6b show that h changes the balanced accuracy with at most 15 % for $Q = 30\%$ NLOS anchors and at most 10 % for $Q = 50\%$ NLOS anchors. On the other hand, Fig. 6c and 6d show that h has a marked impact on the percentage of classified instances. P_C is the highest for $h = [0.03, 0.06]$ (depending on the number of anchors) and decreases for values outside the interval. Because, for different values of h , the change in accuracy is small but the change in P_C is large, we want a performance score based on these two metrics which increases the impact of P_C . Therefore, we chose as the aggregated performance score the harmonic mean between the balanced accuracy and P_C .

In practice, we usually have mixed NLOS conditions which can change over time. Therefore, we extend the performance score to be the harmonic mean of the balanced accuracy and percentage of classified instances for both $Q = 30\%$ and 50% and choose the KDE shaping parameter which maximizes this score. The optimum shaping parameter is $h = 0.04$ for $N = 5$ to 8 anchors and $h = 0.03$ for $N = 9$ anchors. TABLE II shows the LOS, NLOS, and balanced accuracy and the percentage of classified instances for the optimum h .

The classification accuracy is higher for 30 % NLOS anchors than for 50 %. When more anchors are in NLOS with the tag, the location estimate is more skewed and the residuals of all anchors (not only of NLOS anchors) are larger. In this case, LOS and NLOS anchor residuals are harder to distinguish.

We note that the accuracy slightly decreases for more anchors. This is because, with more anchors (out of which only a few are in NLOS), it is harder to find a value of h low enough to delimit the few NLOS anchors from the LOS ones but high enough to avoid an oversmoothed distribution that leads to more than two intervals.

TABLE II
PERFORMANCE OF UNSUPERVISED RESIDUAL-BASED LABELING FOR THE OPTIMUM h .

N	Q [%]	Accuracy [%]			Classified instances [%]
		LOS	NLOS	Balanced	
5	30	68.2	98.7	83.4	37.6
	50	66.8	82.1	74.4	40.2
6	30	68.3	98.9	83.6	29.3
	50	72.0	76.5	74.2	39.6
7	30	64.7	84.5	74.6	34.6
	50	62.4	79.5	70.9	37.6
8	30	74.4	81.1	77.7	32.8
	50	69.9	73.3	71.6	38.1
9	30	70.2	79.2	74.7	36.8
	50	66.0	72.5	69.2	35.2

B. Supervised Classification

We now evaluate the accuracy of the RF classification applied on measurements labeled with residual analysis. The training set consists of labeled measurements and their features. We aggregated the measurements for $Q = 30\%$ and 50 % NLOS anchors, since in practice we can have mixed NLOS conditions. We train a classifier for each number of anchors. The test set contains all measurements which were not labeled in the previous step, i.e., where the density of anchor residuals did not have exactly one minimum. The training set has approximately 6,500–10,000 samples and an almost equal number of LOS and NLOS samples. We use stratified K-fold cross-validation with $K = 4$ folds to identify the best model's parameters from a specified subset.

Fig. 7 shows the LOS, NLOS, and balanced accuracy of the model. The NLOS detection accuracy slightly exceeds 90 % in all cases, while the LOS accuracy exceeds 95 %. Compared with only the residual-based labeling, we gain 10–20 % accuracy. It is perhaps surprising that the RF classification accuracy exceeds 90 % even when the labeling accuracy can be as low as 63 %. This is because noisy labels resemble outliers or anomalies and ML models can usually recover to a certain extent the correct class boundaries [16].

C. NLOS Mitigation

We now devise a strategy for handling NLOS measurements in order to improve the localization accuracy. We present the

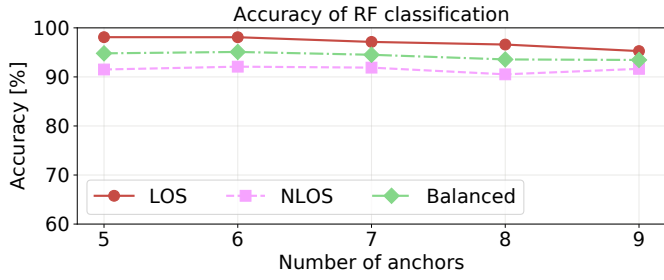


Fig. 7. The accuracy of the trained random forest (RF) classifier using data labeled with the residual method.

Algorithm 1 NLOS Mitigation

```

1:  $\mathbf{x}_{A_i}$  = Location of anchor  $A_i$ ,  $i = 1, \dots, N$ 
2:  $d_i$  = Distance from anchor  $A_i$  to the tag
3:  $S_{LOS} = \{A_i | A_i \text{ is in LOS with the tag}\}$ 
4:  $D$  = Number of dimensions
5: if ( $|S_{LOS}| \geq D + 1$ ) and ( $S_{LOS}$  is not degenerate) then
6:    $\hat{\mathbf{x}} = \text{compute\_location}(\{d_i | A_i \in S_{LOS}\})$ 
7:   for  $i = 1, \dots, N$  do ▷ Distance correction
8:     if  $A_i$  in NLOS then
9:        $r_i = \|\mathbf{x}_{A_i} - \hat{\mathbf{x}}\|$  ▷ Anchor residual
10:       $d_i \leftarrow d_i - r_i$  ▷ Corrected distance
11:   return  $\hat{\mathbf{x}} \leftarrow \text{compute\_location}(\{d_i, i = 1, \dots, N\})$ 
12: else
13:    $\mathcal{S} =$  The set of all anchor subsets using all LOS
14:   anchors and all combinations of NLOS anchors
15:   for  $S_k$  in  $\mathcal{S}$  do
16:      $\hat{\mathbf{x}}_k = \text{compute\_location}(\{d_i | A_i \in S_k\})$ 
17:      $R_k = \frac{1}{|S_k|} \sum_{\{i | A_i \in S_k\}} r_i^2$  ▷ Subset residual
18:   return  $\hat{\mathbf{x}} = \left( \sum_{k=1}^{|\mathcal{S}|} \hat{\mathbf{x}}_k R_k^{-1} \right) / \left( \sum_{k=1}^{|\mathcal{S}|} R_k^{-1} \right)$ 

```

NLOS mitigation procedure as a pseudocode in Algorithm 1 and describe it in the following.

If there are enough LOS anchors to compute one location (at least $D + 1$ anchors for D dimensions) and the set of anchors is not degenerate (i.e., the anchors are not collinear), we can use only the LOS anchors to compute the location. However, if there are few LOS anchors and their placement is not ideal (for instance, the tag falls outside the convex hull of the anchors), the location estimated using only the LOS anchors can sometimes have large errors. Therefore, we noticed that we obtain better location estimates if we correct NLOS measurements and use them for localization. For correction, we first estimate the intermediate location using only the LOS anchors. We compute the residuals of the NLOS anchors based on the intermediate location. Then, we subtract the residuals from the measured distances of NLOS anchors. We estimate the final location using the distance measurements of LOS anchors *and* the corrected distances of NLOS anchors.

If the set of LOS anchors is degenerate or there are not enough LOS anchors to compute the location, we must use some NLOS anchors to compute the tag's location. Because we cannot correct the NLOS measurements as in the previous

TABLE III
LOCALIZATION ERROR WITH NLOS MITIGATION

N	Method	Q = 30 % NLOS		Q = 50 % NLOS	
		Mean [m]	Std. dev. [m]	Mean [m]	Std. dev. [m]
5	No mitigation	0.14	0.27	0.20	0.39
	Proposed	0.07	0.06	0.11	0.14
6	No mitigation	0.11	0.08	0.19	0.31
	Proposed	0.06	0.04	0.10	0.17
7	No mitigation	0.14	0.30	0.21	0.64
	Proposed	0.07	0.05	0.08	0.07
8	No mitigation	0.14	0.36	0.22	0.67
	Proposed	0.06	0.05	0.08	0.06
9	No mitigation	0.13	0.35	0.20	0.64
	Proposed	0.05	0.04	0.07	0.14

case, we generate all subsets S_k containing all LOS anchors and all combinations of NLOS anchors such that $|S_k| \geq D + 1$. For each subset, we compute the intermediate location and the subset's residual using Eq. 7. The final location is the weighted linear combination of all intermediate locations, similar to the method proposed in [1], except that we do not use all possible subsets, thus reducing the complexity of the algorithm.

We compute the localization error as the Euclidean distance between the estimated location $\hat{\mathbf{x}}$ and the true location \mathbf{x} :

$$e = \|\hat{\mathbf{x}} - \mathbf{x}\|. \quad (13)$$

We evaluate the localization error on the test data set from Section III-C. For each set of anchor-tag measurements, we predict the LOS/NLOS condition using the trained model and then apply Algorithm 1 to mitigate the NLOS errors. TABLE III compares the localization errors obtained with the proposed mitigation algorithm (denoted by "proposed") with those obtained when using all anchor-tag distances, without mitigation. The localization errors of the proposed method have 1.8–2.8× smaller mean and 1.8–11.6× smaller standard deviation after NLOS mitigation. On average, the algorithm reduces the mean and standard deviation by 2.2 and 5.8 times, respectively. Therefore, the proposed method can successfully mitigate localization errors caused by NLOS propagation.

VI. DISCUSSION AND FUTURE WORK

One remaining question is whether it is worth deploying more anchors than the minimum necessary (e.g., three anchors for 2D localization). In buildings with rooms separated by thick walls, if we want to provide high-accuracy location services everywhere, we have to deploy at least three anchors in each room. However, the tag can still be in the range of anchors in other rooms, so it has to decide which anchors are in LOS. Even within the same room, some anchors might be shadowed by surrounding objects, so it is worth having extra anchors. In TABLE III, we see that when we correct NLOS errors, the localization error decreases for more anchors. Therefore, more anchors than the minimum are often needed in practical deployments.

So far, we have not discussed which entity should train the classifier: the localization engine (LE) or the tag. There are arguments for both sides. On the one hand, NLOS errors depend on the environment in which the LE operates. For instance, in an industrial setting with metallic objects, NLOS errors might be larger than in an office. Therefore, the LE could collect anchor–tag distances from tags operating in an area and train a model which can then classify all measurements at the LE or the tag, depending on which entity computes the location. On the other hand, the learned CIR features can also depend on the hardware of the end device. For instance, the CIR can have different shapes depending on the type of UWB device [15]. Therefore, models trained on features from one hardware model might not generalize well to others. One future research direction is to evaluate how well a model trained on one type of hardware or at one location generalizes to other device models and environments. If CIR features are indeed model- or environment-specific, one option is to train an initial model and periodically update it with new data from different devices and locations. This is also beneficial if the training is performed by the tag, since it requires less data storage and the model can be updated online. Since the residual labeling step outputs labels only for a part of the input measurements, the accuracy of the model could be improved using semi-supervised learning methods. For instance, the model can be retrained using its most confident predictions [17], which can also speed up the training process.

The localization error can also be reduced by applying Chen’s residual weighting algorithm [1] or variations of it, without going through the labeling and training process. However, computing the location using all anchor combinations has a complexity of $\mathcal{O}(2^N)$ for N anchors. In our case, the complexity is high only during the short training phase. After this, samples can be classified with the RF with a constant complexity of $\mathcal{O}(kp)$, where k is the number of decision trees and p is the maximum depth of one tree. In practice, the execution time of the proposed NLOS mitigation method (excluding the training phase) was faster than the residual weighting algorithm for $N \geq 8$ anchors.

NLOS identification has interesting applications beyond reducing localization errors, especially when it can be done without supervision, as in our proposal. Our method can potentially be used to build maps of a building by aggregating the locations at which the tag is consistently in NLOS with certain anchors. Crowd density estimation is another interesting possible application. Since human body shadowing introduces large distance errors, any increase in the number of detected NLOS measurements could suggest that a room gets more populated. Note that this method does not require all users to be connected to the localization network. Finally, the trained model can be applied on individual distance measurements, so it can be useful in peer-to-peer proximity applications (e.g. contact tracing, object finding).

VII. CONCLUSIONS

We proposed a method for detecting NLOS measurements in localization systems without manually-acquired training data or knowledge of channel statistics. The method predicts the LOS/NLOS labels using the measured distances between each anchor and the tag. We use the predicted labels and the CIR features of the measurements to train a classifier, which has over 90 % classification accuracy. We also proposed a NLOS mitigation technique which reduces, on average, the mean and spread of the localization error by at least $2\times$.

REFERENCES

- [1] P.-C. Chen, “A non-line-of-sight error mitigation algorithm in location estimation,” in *WCNC. 1999 IEEE Wireless Communications and Networking Conference*, vol. 1, pp. 316–320, IEEE, 1999.
- [2] 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.
- [3] H. Wymeersch, S. Marano, 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.
- [4] J. Schroeder, S. Galler, K. Kyamakya, and K. Jobmann, “NLOS detection algorithms for ultra-wideband localization,” in *2007 4th Workshop on Positioning, Navigation and Communication*, pp. 159–166, IEEE, 2007.
- [5] R. M. Vaghefi, J. Schloemann, and R. M. Buehrer, “NLOS mitigation in TOA-based localization using semidefinite programming,” in *2013 10th Workshop on Positioning, Navigation and Communication (WPNC)*, pp. 1–6, IEEE, 2013.
- [6] 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.
- [7] G. Wang, W. Zhu, and N. Ansari, “Robust TDOA-based localization for IoT via joint source position and NLOS error estimation,” *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8529–8541, 2019.
- [8] S. Li, M. Hedley, I. B. Collings, and D. Humphrey, “Joint trajectory and ranging offset estimation for accurate tracking in NLOS environments,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 1, pp. 3–14, 2019.
- [9] U. Hammes and A. M. Zoubir, “Robust MT tracking based on M-estimation and interacting multiple model algorithm,” *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3398–3409, 2011.
- [10] 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.
- [11] J. Park, S. Nam, H. Choi, Y. Ko, and Y.-B. Ko, “Improving deep learning-based uwb los/nlos identification with transfer learning: An empirical approach,” *Electronics*, vol. 9, no. 10, p. 1714, 2020.
- [12] 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*, 2020.
- [13] L. Cheng, Y. Li, Y. Wang, Y. Bi, L. Feng, and M. Xue, “A triple-filter NLOS localization algorithm based on fuzzy c-means for wireless sensor networks,” *Sensors*, vol. 19, no. 5, p. 1215, 2019.
- [14] N. Sirola, “Closed-form algorithms in mobile positioning: Myths and misconceptions,” in *2010 7th Workshop on Positioning, Navigation and Communication*, pp. 38–44, IEEE, 2010.
- [15] L. Flueteru, S. Wehrli, M. Magno, E. S. Lohan, and D. Niculescu, “High-accuracy ranging and localization with ultra-wideband communications for energy-constrained devices,” *IEEE Internet of Things Journal*, pp. 1–1, 2021.
- [16] B. Fréney and M. Verleysen, “Classification in the presence of label noise: a survey,” *IEEE transactions on neural networks and learning systems*, vol. 25, no. 5, pp. 845–869, 2013.
- [17] I. Triguero, S. García, and F. Herrera, “Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study,” *Knowledge and Information systems*, vol. 42, no. 2, pp. 245–284, 2015.