Received July 20, 2018; | Revised March 3, 2019; | Accepted April 12, 2019
DOI: xxx/xxxx

ARTICLE TYPE

A Decision-Tree based NLOS Detection Method for the UWB Indoor Location Tracking Accuracy Improvement

Ardiansyah Musa[1] | Gde Dharma Nugraha[1] | Hyojeong Han[1] | Deokjai Choi*[1] | Seongho Seo[2] | Juseok Kim[2]

[1]Advanced Network Lab, School of Electronics and Computer Engineering, Chonnam National University, Gwangju, South Korea
[2]Force Training System (FTS), Gwangju, South Korea

Correspondence
*Deokjai Choi. Email: dchoi@jnu.ac.kr

Summary

Among existing wireless technologies, ultra-wide-band (UWB) is the most promising solution for indoor location tracking. UWB has a great multipath fading immunity, however great multipath resolvability alone does not eliminate the effect of non-line-of-sight (NLOS) and multipath propagation. NLOS and multipath propagation in indoor environments can easily produce meters of UWB ranging error. This condition gives an enormous impact on the accuracy of indoor location tracking data. To address this problem, we propose an NLOS detection method using recursive decision-tree learning. Using the UWB channel quality indicators information, we develop our model with the Gini index and altered-priors splitting criteria. We then validate the constructed model using the 10-fold cross-validation method. Our experiment shows that the constructed model has correctly detected 90% of both line-of-sight (LOS) and NLOS cases on the seven different indoor environments. The result of this work can be used for the UWB indoor location tracking accuracy improvement.

KEYWORDS:
Decision-Tree Learning, Indoor Location Tracking, NLOS Detection, UWB

1 | INTRODUCTION

At present, location tracking data play an essential role in the internet of things (IoT) services[1]. In an indoor environment, this data can be provided by the use of the indoor location tracking solutions which commonly based on the wireless communication[2]. The most popular wireless technologies are the radio-frequency identification (RFID), the wireless fidelity (Wi-Fi), the Bluetooth low energy (BLE) and the ultra-wideband (UWB). Among these wireless technologies, UWB is the most promising solution for potential accuracy, ease of installation, and functional range[3]. UWB has an absolute bandwidth larger than 500 MHz and the transmission range greater than 100 meters.

In more detail, there are different methods for implementing indoor location tracking using wireless schemes, but they effectively devolve into three categories[4]. The first category based on the radio signal strength indicator commonly referred to as the RSSI-based scheme. The RSSI-based scheme is highly appreciated because of its minimal complexity. Thus, some RSSI-based application has been developed to track the location of an object in the indoor area[5,6]. However, this scheme required many reference points to find out the best possible match from a group of possible locations. The second category is based on the direction of the signal, which estimates the angle between the arrival of the signal to the array of reference antennas. This category is commonly known as Angle of Arrival (AOA) based scheme. The accuracy of this scheme can be high (within a few degrees). However, it required a significant hardware cost. The last category based on the measurement of time of flight (ToF), where the time it takes the radio signal to travel between transmitter and receiver is measured. This method is also known as Time of Arrival (TOA) based scheme. For UWB ranging, due to their nature of the inverse relationship of time and frequency, the lifetime of UWB signals is very short. Consequently, the TOA-based scheme is the most suitable method.

Currently, there are several types of ToF measurement: one-way ranging, two-way ranging, and private. Based on the IEEE 802.15.4a standard[7], the two-way-ranging (TWR) is the mandatory type for the IoT services. TWR involves a two-way exchange of messages between the tracking object and each reference point. In the TWR mechanism, at first, the reference points transmits a message to the tracking object and records the departure time. When the tracking object receives the message from the reference points, the tracking object will send a reply message. The reference points then calculate ToF between their message departure time and the reply arrival time. Using ToF information, the reference points then estimate the distance between them-self and the tracking object. Based on this distance information the indoor location tracking system can determine the tracking target location using a localization method.

The common localization method is based on the trilateration algorithm[8,9,10,11,12,13,14]. This algorithm requires three or more reference points to determine the location of an object. In[15] we proposed another localization method, a dulateration-based indoor location tracking (Dilacak) algorithm, which used to track a moving object using only two reference points. This algorithm is used to estimate the continuous location of the target, by combining the knowledge of ToF with the information of the initial/previous target position. This method was effective to decrease the amount of packet collision and the overall power consumption used by the system. However, as illustrated in Figure 1 , the existence of the obstacle such as a concrete pillar between a reference point caused the estimated location sometimes become unacceptable.

Actually, UWB has a great multipath fading immunity compared to other communication technologies. However, great multipath resolvability alone does not eliminate the effect of non-line-of-sight (NLOS) and multipath propagation[16]. When a reference point is experiencing an NLOS and multipath propagation condition, the signals need to travel around the obstacle. The ToF information will be longer which can easily produce meters of UWB ranging error. This condition gives an enormous impact to the location tracking accuracy. Moreover, in the case of high noise values, the wireless first path signal can be buried in the noise. Hence, the need for NLOS detection is indispensable.

By using detailed information about UWB channel quality indicators, an intelligent NLOS detection method can be developed based on a machine learning approach. In this paper, we propose an intelligent NLOS detection method using decision-tree learning. The result of this work can be used for indoor location tracking accuracy improvement. The rest of this paper is organized as follows: Section 2 discusses the sources of UWB ranging error and the channel quality indicators. Section 3 describes our proposed NLOS detection method. Section 4 presents our experimental results and the validation analysis in detail. Finally, section 5 concludes the paper.

2 | UWB RANGING ERRORS AND CHANNEL QUALITY INDICATORS

2.1 | The Sources of TOF Measurement Error

In the UWB ranging, the source of TOF measurement error commonly come from the internal (e_{LOS}) which includes all typical sources of error (i.e., finite bandwidth, printed circuit boards losses, clock drift, thermal noise). The crystal oscillator used in both the tracking target and the reference points sometimes is not working precisely. A small positive or negative offset may occur in the ToF measurement[17]. This small offset can cause a significant impact because one nanosecond of ToF error can lead to an approximate error of 30 cm in distance estimation.

To minimize the mentioned inaccuracy due to all typical sources of error, several TWR methods available in the literature[17,18,19,20,21,22]. The existing solutions include the single-sided TWR (SS-TWR), the symmetric double-sided TWR (SDS-TWR), and asymmetric double-sided TWR (ADS-TWR). For example in SDS-TWR protocol, each of the UWB nodes has an estimate of the round trip time T_r and turnaround time T_{ta} as depicted in equation (1)

$$TOF = \frac{(T_r^{ref_i} - T_{ta}^{ref_i}) + (T_r^{target} - T_{ta}^{target})}{4} \tag{1}$$

Where, $T_r^{ref_i}$ and $T_{ta}^{ref_i}$ are the round trip time and turnaround time of the reference point i, T_r^{target} and T_{ta}^{target} are the round trip time and turnaround time of the locating target.

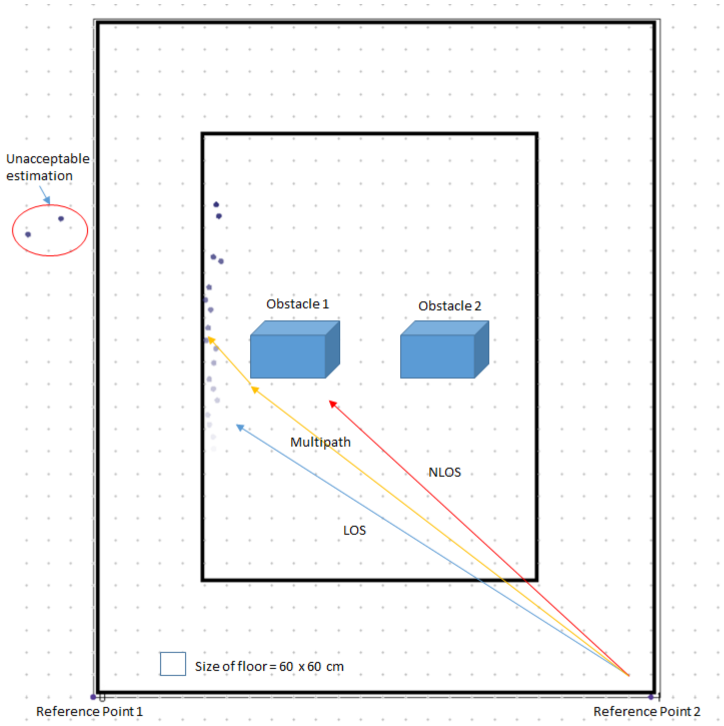


FIGURE 1 UWB Ranging in NLOS Environment.

To observe the error, we can define the frequency offsets of each reference point and the locating target as e_{ref} and e_{target}, as depicted in equation (2) and (3)

$$e_{ref} = \frac{RF_{ref} - NF_{ref}}{NF_{ref}} \tag{2}$$

$$e_{target} = \frac{RF_{target} - NF_{ref}}{NF_{target}} \tag{3}$$

where, RF_{ref} and RF_{target} are the real frequency of the reference and target node, NF_{ref} and NF_{target} are the nominal frequency of the reference and target node. The estimated ToF by the SDS-TWR protocol is depicted in equation (4) and (5)

$$TOF = \frac{(T_r^{ref_i} - T_{ta}^{ref_i})(1 + e_{ref_i}) - (T_r^{target} - T_{ta}^{target})(1 + e_{target})}{4} \tag{4}$$

However, in the NLOS and multipath propagation condition, the ToF measurement error is also affected by external error sources (e_{NLOS}). These external error sources can be categorized as follows[22].

- Propagation Time Delay (PTD). PTD occurs in two cases: 1) When the direct signal path is obstructed/blocked completely, 2) when the signal has to transverse through different materials.
- Preamble Accumulation Time Delay (PATD). PATD is influenced by the presence of multi-path and the quick frame arrival time because of relatively short distance measurements.

The estimated distance r_i regarding the e_{LOS} and e_{NLOS} is depicted in equation (5)

$$r_i = c \times ToF = \begin{cases} r_i + e_{LOS}, \text{if LOS} \\ r_i + e_{LOS} + b, \text{if NLOS} \end{cases} \tag{5}$$

Where, c is the speed of light = $3x10^8$ m/s, ToF is the reported time of flight from UWB ranging process, e_{LOS} is the ranging error in the LOS scenario (typical source of error), and e_{NLOS} is the result of the e_{LOS} and the random bias b caused by PATD and PTD. These errors either need to be eliminated or controlled to achieve an efficient and accurate ranging solution for indoor location tracking. Hence, we need an NLOS detection method which can be used later to mitigate and refine the distance estimation.

TABLE 1 UWB Channel Quality Indicators

Pulse Repetition Frequency (R_f)
Channel Number (C_n)
Transmission Preamble Length (P_l)
Standard Deviation of Noise (N_s)
Maximal Data Rate (B_m)
Total Channel Impulse Response Power (CIR_p)
Index of Detected First Path (F_i)
First Path Amplitude part 1 (F_1)
First Path Amplitude part 2 (F_2)
First Path Amplitude part 3 (F_3)
Length of Received Frame (F_l)
Maximum Noise Value (N_m)
Time-of-Flight Report (ToF)
Received RX Preamble Symbols (RX)

2.2 | UWB Channel Quality Indicators

Ideally, there should be no relationship between the reported timestamp of a received signal and the received signal level (RSL) at the reference points. However, in practice, a bias which varies with RSL can be observed in the reported timestamp compared with the correct value, and this leads to a bias in the calculated ToF on those time-stamps. In the case of a LOS condition, the signal power of the unobstructed first path F_1 as it arrives at the references can be calculated based on the distance reported using Friis' path loss $FPath$ formula[17,19]. The $FPath$ formula depicted in equation (6).

$$FPath_i = TX + G + 20log_{10}(c) - 20log_{10}(4\pi \times f_c \times (r_i + e_{LOS})) \tag{6}$$

Where, $FPath_i$ is the received power level at the reference point i in dBm, TX is the transmitted power in dBm, G included the antenna gains of the transmitting and receiving antennas, c is the speed of light = $3x10^8$ m/s, f_c is the center frequency of the channel used in Hz, r_i is the reported distance in meter from UWB ranging. In the LOS condition, the levels of noise N_m compared to the $FPath_i$ have little impact on the ToF accuracy. However, in case of high noise values and NLOS signal path, the unobstructed first path F_1 can be buried in the noise[23,24]. The calculation of estimated $FPath_i$ in NLOS condition is depicted in equation (7)[9,10,17]

$$FPath_i = 10log_{10}(\frac{F_1^2 + F_2^2 + F_3^2}{N^2}) - A \tag{7}$$

Where, F_2, and F_3 are the first path amplitudes in multipath propagation. N is the preamble accumulation count value, and A is the predefined constant of 113.77 dBm for the pulse repetition frequency R_f of 16 MHz or 121.74 dBm for a R_f of 64 MHz. In the traditional approach, we need manually compare the resultant of $FPath_i$ with the estimated RSL to identify LOS/NLOS condition[9,10]. With the information of the channel impulse response CIR , the pulse repetition frequency used R_f, and the received RX preamble symbols RX, the calculation of estimated RSL is depicted in equation (8).

$$RSLi = 10log_{10}(\frac{(CIR)(2^{17})}{N^2}) \tag{8}$$

As depicted in Table 1 , we select fourteen related UWB channel quality indicators that can be used as LOS and NLOS detection features. In NLOS environment, knowing all the parameters above, a machine-learning based NLOS detection method can be constructed, and the actual distance can be determined later using range bias b correction on the mitigating and filtering process. In this paper, we develop an intelligent NLOS detection based on the machine learning approach. Since NLOS detection is the binary problem, we construct a decision-tree learning model as the solution.

3 | PROPOSED NLOS DETECTION

3.1 | Dataset

Representative data with a different degree of multipath effects and ranging errors in the real-world environment is needed to make a good NLOS detection model. In this work, to build the model, we use EWINE UWB LOS and NLOS dataset which published online in[25]. UWB channel C_n number 2 with central frequency f_c 3.9936 GHz and bandwidth B 499.2 MHz were used to collect this dataset. To get the best performance in case of NLOS signals and consequently better accuracy in first path signal detection, the longest synchronization header (SHR) preamble length of 4096 was also used. Besides, to extend the communication range, the transmitted preamble length 1024 symbols P_l and the maximal data rate B_m 110 kbps we used.

The measurements were taken on the seven different indoor environments: Office 1, Office 2, Small Apartment, Small Workshop, Kitchen with a Living Room, Bedroom and Boiler Room. The location of the reference point was fixed throughout the whole experiment while the tracking target was moved through all the predefined positions. Each measurement site had a fixed relative coordinate which simplified the calculation of the actual distances between the tracking target and the reference point. In total 42000 samples were taken, 3000 LOS samples and 3000 NLOS samples in each site. We combine 57% data samples from each site for training, and these samples were randomized to prevent overfitting of a model to particular locations. Since the magnitude of dataset varies with different high range, we normalize this dataset to the range [0,1]. We then labeled NLOS class as 1 and LOS class as 0.

3.2 | Building NLOS Classifier

The decision-tree learning is one of the most popular machine learning techniques used all along. The decision tree can be used for both classification and regression problems in data science. This algorithm usually used mostly because of the following reasons[26,27]

1. Decision tree often mimics the human level thinking for understanding and interpreting the data.
2. Decision tree lets us see the logic for the data interpretation, not a black box algorithm like other machine learning algorithms (i.e., Support Vector Machine, Neural Networks).

The whole idea of decision-tree learning is to create a tree for the entire training data and process a single outcome at every leaf or minimize the error in every leaf. Decision-tree places the best feature at tree root. The dataset then split into the subsets, each subset in such a way should contain data with the same value for a feature. Figure 2 depicted an example application of the decision tree learning. In this case, the decision-tree used for digit classification. We can see the logic of this algorithm in deciding digit classes from 0 to 9 based on the available features. Each node represents a used feature (i.e., id-285, id-254), each link represents a decision rule (yes or no), and each leaf represents an outcome/class (0 to 9).

Decision-tree learning follows the sum of product (SOP) representation or also known as disjunctive normal form (DNF). DNF is standardization or normalization of a logical formula which is a disjunction of conjunctive clauses[28]. A logical formula is considered to be in DNF if and only if it is a disjunction of one or more conjunctions of one or more literals. A DNF formula is in full disjunctive normal form if each of its variables appears exactly once in every conjunction. In decision-tree learning, for a class, every branch from the root of the tree to a leaf node having the same class is conjunction (product) of values, different branches ending in that class form a disjunction (sum).

It is a complicated step for deciding which feature to place at the root or different levels of the tree. By just randomly selecting any feature to be the root cannot solve the issue, and it may give us bad results with low accuracy. For solving the features selection problem, researchers worked and devised several solutions. Among existing solution, Information gain and Gini index are the popular criterions for features selection[29]. While using Information gain, we assume features to be categorical and while using Gini index, features are assumed to be continuous.

The most practical problem for generating a decision-tree model is over-fitting. The generated model is considered having an issue of over-fitting when the learning algorithm continues reducing the training dataset error, but the accuracy of the prediction test goes down. A post-pruning approach usually implemented to validate the generated model. Moreover, decision-tree learning also equipped with measures of variable importance. Importance score for a feature is calculated by summing the impurity reductions over all nodes in the tree where a split was made on that feature.

In this paper, we develop an NLOS detection model with the recursive decision-tree method. We implement decision-tree learning using the Recursive PARTitioning (RPART) routines packages[30]. We divide the process into two stages. In the first

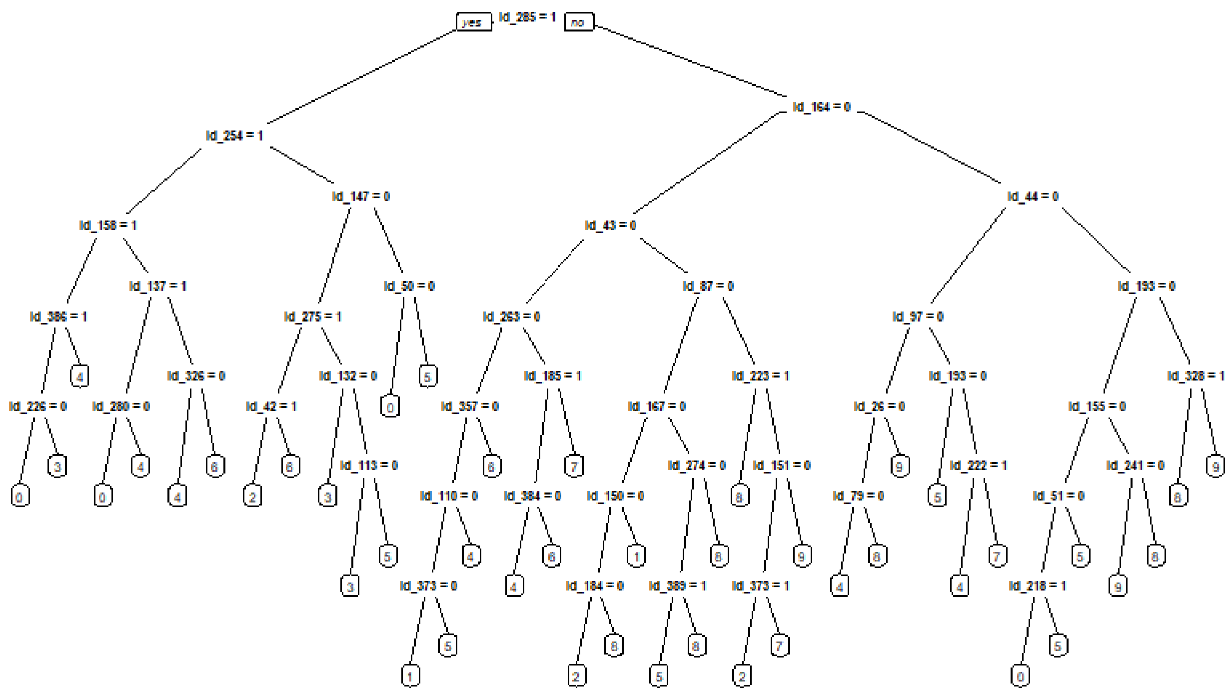


FIGURE 2 Example of the Decision-Tree Application

stage, we try to find the best single feature to split the training dataset into two groups. The training dataset is separated into each sub-group recursively until these subgroups either reach the level of preciseness or until no improvement can be made. In the second stage, we implement a cross-validation method to check and estimate the risk of the generated model.

The sample data consists of n observations from X classes for the classification process. The generated model will break these observations into k terminal groups, and to each of these groups, we assign a predicted class. Equation (9) depicted the probability P of some node of the tree A for future observations in the decision-tree classification.

$$P(A) = \sum_{i=1}^{X} \pi_i P\{x \in A | \tau(x) = i\} \approx \sum_{i=1}^{X} \pi_i n_A / n_i \tag{9}$$

Where, π_i $(i = 1, 2, ..., X)$ is the prior probabilities of each class, $\tau(x)$ is the true class of an observation x, where x is the vector of predictor features. n_i is the number of observations in the sample that are class i while n_A is the number of observations in node A.

The estimated risk R of nodes A and the whole constructed tree $R(T)$ are depicted in equation (10) and (11)

$$R(A) = \sum_{i=1}^{X} p(i|A) L(i, \tau(A)) \tag{10}$$

$$R(T) = \sum_{j=1}^{k} P(A_j) R(A_j) \tag{11}$$

Where Aj are the terminal nodes of the tree, $\tau(A)$ is the chosen class with the minimum risk to A; $L(i, \tau(A))$ is the loss matrix for incorrectly classifying of A. Moreover, in order to measure of impurity of the splitting fitness, we can use several impurity functions f for a node A as depicted in equation (12)

$$I(A) = \sum_{i=1}^{X} f(p_{iA}) \tag{12}$$

Where p_{iA} is the proportion of those in A that belong to class i for future samples. Depending on the level of preciseness, we want the $I(A) = 0$ when A is pure. So f must be concave with f(0) = f(1) = 0.

TABLE 2 List Parameters for the Control Aspects of the Splitting Fitness

Parameter	Function
minsplit	the minimum number of observations in a node for a split to be attempted
cp	the treshold value of fitness
minbucket	the minimum number of observations in a terminal/leaf node
maxsurrogate	the maximum number of surrogate variables to retain at each node
usesurrogate	how to use surrogates in the splitting fitness:
	0: display only,
	1: use surrogates in order to split subjects missing the primary variable,
	2: if all surrogates are missing then send the observation in the majority direction
xval	number of cross-validations
surrogatestyle	controls the selection of a best surrogate
	0: the program use the total number of correct classification for a potential surrogate variable
	1: uses the percent correct, calculated over non-missing value of the surrogate
maxdepth	set the maximum depth of any node of the final tree ($2^n - 1$)

To split the tree, we use Gini index criteria $f(p) = p(1 - p)$. We then use that split with maximal impurity reduction. In addition, we also implemented the altered-priors method for incorporating loss because of misclassification. We can use several parameters for the controlling aspects of splitting fitness. For example, if we set a tree model with the $minsplit$ to 20 and the complexity parameter cp is 0.01, it is means

- Minimum 20 number of observations must exist in a node for a split to be attempted
- Any split that does not decrease the overall lack of fit with the threshold of complexity parameter = 0.01 is not attempted.

Table 2 describes all the parameters for the control aspect of the splitting fitness.

4 | RESULTS

4.1 | LOS/NLOS Detection Results

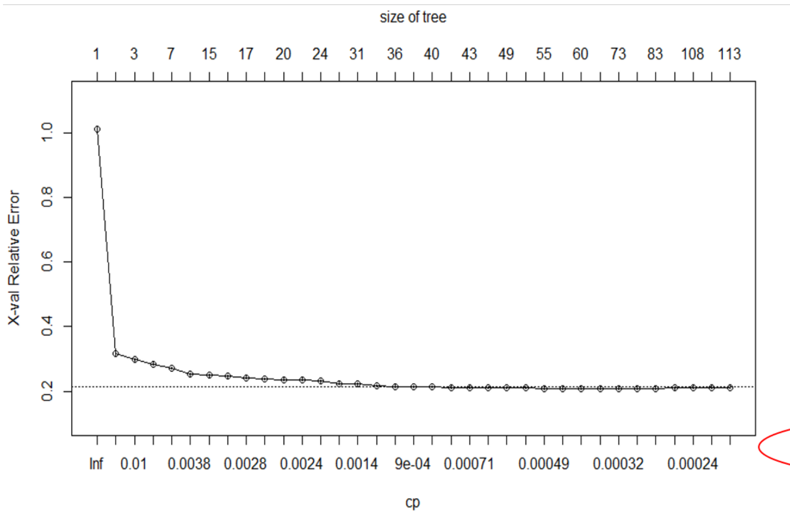
As mentioned in the previous section, we used decision-tree learning to build an intelligent NLOS detection method. The model was generated with the configured parameters depicted in Table 3 . The constructed model has correctly detected 91.95% both LOS and NLOS cases in training data. However, before we use the model to the testing data or the real-world application, we need to check and validate the generated model is not overfitted or independent of this dataset. Therefore, we validated our model using 10-Fold cross-validation method. The 10-Fold cross-validation method performed in the following steps:

1. Partition the original training dataset into 10 equal subsets. Each subset is called a fold named as f_1, f_2, ..., f_{10}.
2. For i = 1 to i = 10
   (a) Keep the fold f_i as validation set and keep all the remaining 9 folds in the cross-validation training set.
   (b) Training the generated model using the cross-validation training set and calculate the accuracy of the model by validating the predicted result against the validation set.
3. Estimate the accuracy of our model by averaging the accuracies derived in all the 10-cases of cross-validation.

Figure 3 depicted the result of the validation process. It is shown that the model was overfitted. Therefore, we need to prune back our model by selecting a tree size that minimizes the cross-validation error. In this case, the tree size should be under 72. Table 4 describes the important variables used in the tree construction and the classification results of our NLOS detection method based on decision-tree learning on the training and validation data.

TABLE 3 Configured Parameters for the Decision-Tree Model

criterion	samples size	minsplit	cp	minbucket	maxsurrogate	maxdepth
Gini index	24.000 samples	40	0.0002	13	5	30



	CP	nsplit	rel error	xerror	xstd
1	0.69075602	0	1.00000	1.01167	0.0064562
2	0.01083604	1	0.30924	0.31608	0.0047100
3	0.00941902	2	0.29841	0.29691	0.0045908
4	0.00594593	3	0.28899	0.28474	0.0045118
5	0.00439693	6	0.27115	0.27123	0.0044208
6	0.00329249	12	0.24373	0.25273	0.0042901
7	0.00325081	14	0.23714	0.24898	0.0042627
8	0.00291740	15	0.23389	0.24564	0.0042381
9	0.00266733	16	0.23097	0.24248	0.0042145
10	0.00250063	18	0.22564	0.23773	0.0041786
11	0.00245895	19	0.22314	0.23614	0.0041665
12	0.00225056	21	0.21822	0.23581	0.0041640
13	0.00208385	23	0.21372	0.23081	0.0041254
14	0.00158373	29	0.20122	0.22364	0.0040691
15	0.00120864	30	0.19963	0.22147	0.0040518
16	0.00095857	33	0.19580	0.21655	0.0040121
17	0.00091690	35	0.19388	0.21522	0.0040012
18	0.00087522	37	0.19205	0.21405	0.0039916
19	0.00083354	39	0.19030	0.21380	0.0039896
20	0.00075019	41	0.18863	0.21230	0.0039772
21	0.00066683	42	0.18788	0.21189	0.0039738
22	0.00061126	44	0.18655	0.21130	0.0039690
23	0.00060432	48	0.18405	0.21105	0.0039669
24	0.00058348	52	0.18163	0.21089	0.0039655
25	0.00041677	54	0.18046	0.20930	0.0039523
26	0.00038899	56	0.17963	0.20930	0.0039523
27	0.00037509	59	0.17846	0.20889	0.0039489
28	0.00033342	63	0.17696	0.20880	0.0039482
29	0.00030563	72	0.17304	0.20855	0.0039461
30	0.00030008	75	0.17213	0.20897	0.0039495
31	0.00026859	82	0.16938	0.20922	0.0039516
32	0.00025006	95	0.16521	0.20989	0.0039572
33	0.00022228	107	0.16221	0.21155	0.0039710
34	0.00020839	110	0.16154	0.21205	0.0039752
35	0.00020000	112	0.16112	0.21222	0.0039765

FIGURE 3 Cross-Validation

TABLE 4 Variable Used for Tree Construction and Classification Accuracy

variables used	training accuracy	validation accuracy
CIR_p, F_1, F_2, F_3, N_m, ToF, RX, N_s	91.35%	89.92%

We then evaluated the performance of the pruned model using the testing data from the seven different indoor environments. Using several standard metrics in the confusion matrix[31], with the actual class is NLOS = 1, the result assignments fall into four categories are as follow.

- true positive (TP): instances that are classified as the actual class
- true negative (TN): instances that are correctly classified as not being the actual class
- false positive (FP): instances that are misclassified as the actual class (type 1 error)
- false negative (FN): instances from the actual class that are misclassified as another class (type 2 error)

The detail of the performance metrics is as follow.

- Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$ is the percentage of correctly classified instances.
- Precision $\frac{TP}{TP+FP}$ is the percentage of correctly classified NLOS instances within all the instances that were classified as NLOS instances.
- Sensitivity $\frac{TP}{TP+FN}$ is the true positive rate or a fraction of correctly classified instances within NLOS class.

TABLE 5 Detection Results

location-site	accuracy	precision	sensitivity	F1
office 1	90.32%	88.01%	91.73%	89.70%
office 2	89.12%	87.93%	89.78%	88.67%
small apartment	90.72%	88.74%	92.33%	90.54%
small workshop	90.10%	85.95%	91.54%	88.90%
kitchen with living room	89.80%	87.81%	90.94%	89.22%
bedroom	90.33%	88.96%	91.20%	90.14%
boiler room	90.50%	90.44%	91.81%	91.25%

- F1 or harmonic mean of precision and sensitivity $\frac{(2*precision*sensitivity)}{(precision+sensitivity)}$

Table 5 summarized our evaluation results. The decision-tree model has correctly accurate detecting 90% of both LOS and NLOS cases on the seven different indoor environments.

4.2 | Real World Applications

The actual impact of our NLOS detection method on indoor location tracking accuracy cannot be extracted from table 5 . To evaluate the impact in the real world applications, we implemented the NLOS detection method as a new component in our existing indoor location tracking system as depicted in Figure 4 . We run the experiment to locate sixteen moving targets in the Force Training System (FTS)[32] indoor training facilities. With the operational UWB channel 2 and the data rate 110 kbps, we placed two reference points near the ceiling level. To estimate the continuous location of the target, we used reported TOF from TWR ranging. We then localized possible target locations by following equations (13) and (14)

$$x = \frac{r_1^2 - r_2^2 - a^2}{2a} \tag{13}$$

$$y = \sqrt{r_1^2 - x^2} \tag{14}$$

Where a is the distance between the reference point 1 and the reference point 2, r_1 and r_2 are the estimated distance between the reference point 1 and the reference point 2 respectively. x and y are the geometrical positions of possible target locations.

To find the best location estimation for each time t, we then selected the minimum distance tp between the target initial location and the possible target locations as depicted in equation (15).

$$tp = min(\sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2}, \sqrt{(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2}) \tag{15}$$

Where (x, y, z) represents the initial target location, (x_1, y_1, z_1) and (x_2, y_2, z_2) represent the possible target location.

For our evaluation, we measured both the distance and location estimation error using the mean absolute error (MAE), and root means square error (RMSE) respectively. We set the threshold error value to 30 cm, regarding the typical source error e_{LOS} as mentioned in section 2. Equation (16) and (17) depicted the MAE and RMSE calculation.

Where n is the number of iteration samples, $\hat{r}(i)$ denotes the true distance of the target and $r(i)$ is the reported distance from the ToF measurement. $(x(i), y(i), z(i))$ represent the real target location and $(\hat{x}(i), \hat{y}(i), \hat{z}(i))$ represent the estimated target location.

$$MAE = \frac{\sum_{i=1}^{n} ||r(i) - \hat{r}(i)||}{n} \tag{16}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (x(i) - \hat{x}(i))^2 + (y(i) - \hat{y}(i))^2 + (z(i) - \hat{z}(i))^2}{n}} \tag{17}$$

$$P_{NLOS}(Ref_i) = p(1 - p)^2 \tag{18}$$

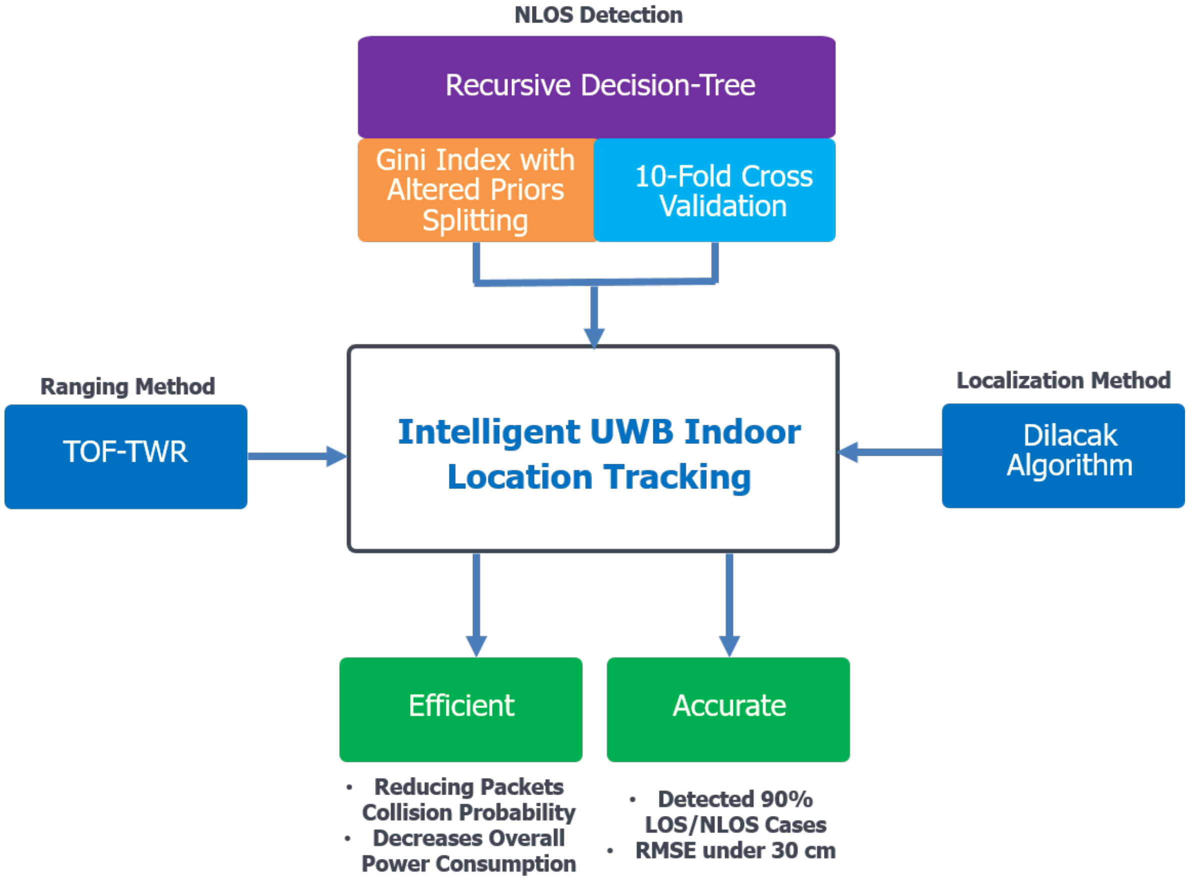


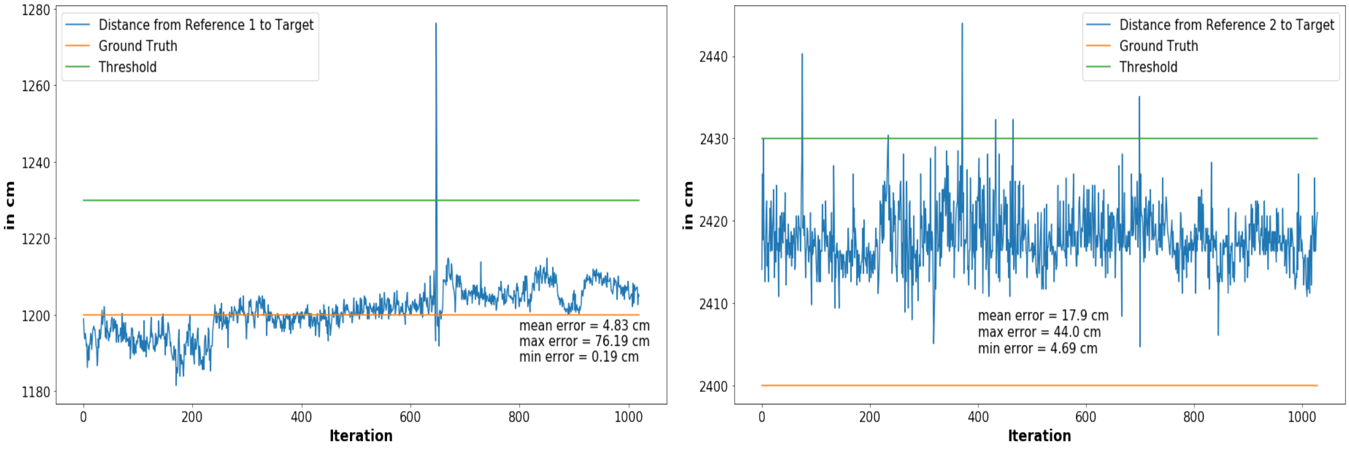
FIGURE 4 Intelligent UWB Indoor Location Tracking

$$P_{NLOS}(Ref_i \cap Refj) = P_{NLOS}(Ref_i)P_{NLOS}(Ref_j) \tag{19}$$

Figure 5 depicted the results of the error estimation in both LOS and NLOS conditions. The result showed that when the reference point was experiencing NLOS condition (Figure 5 b and Figure 5 d, the trend of error is increasing and over than the threshold value. In a system that uses three or more reference points, at least two reference points must be reporting a LOS range[3]. Therefore the system can still maintain the affordable error. In this system, the probability of NLOS P_{NLOS} is modeled in equation (18) and (19). Where i and j are the reference ID, p is the probability that references experiencing NLOS. This model assumes that the most likely NLOS scenario encountered will be a single reference point experiencing NLOS. However, since we use only two references point for tracking a moving object, if one of the reference points experiencing NLOS, the need for NLOS detection is indispensable. When the NLOS signal detected, we send this information to the mitigation and filtering module. This information will be used for either bias b correction or the ranging result elimination. Figure 6 illustrate the indoor location tracking result before and after applying the NLOS detection method. Since the sensitivity of our NLOS detection method is around 91%, the unacceptable location estimation mostly eliminated.

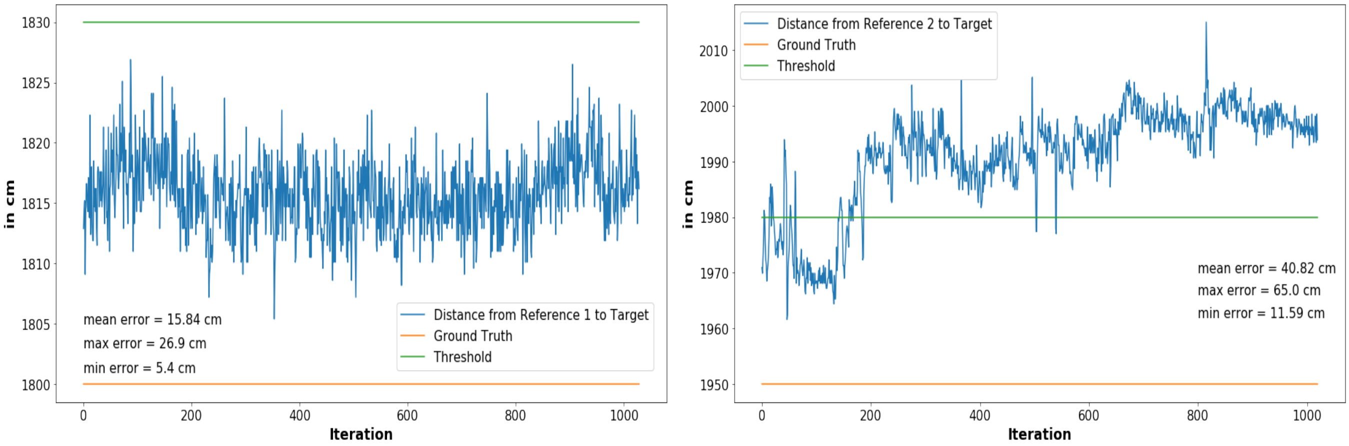
4.3 | Related Works

We have presented in this paper our work to develop an intelligent NLOS detection method. The result of this work can be used for the UWB indoor location tracking accuracy improvement. Table 6 depicted the detailed comparison of our proposed solution with the relevant works in the UWB indoor location tracking systems research and development.



(a) Estimation in LOS condition ($r_1 = 12m$)

(b) Estimation in NLOS condition ($r_2 = 24m$)



(c) Estimation in LOS condition ($r_1 = 18m$)

(d) Estimation in NLOS condition ($r_2 = 19.5m$)

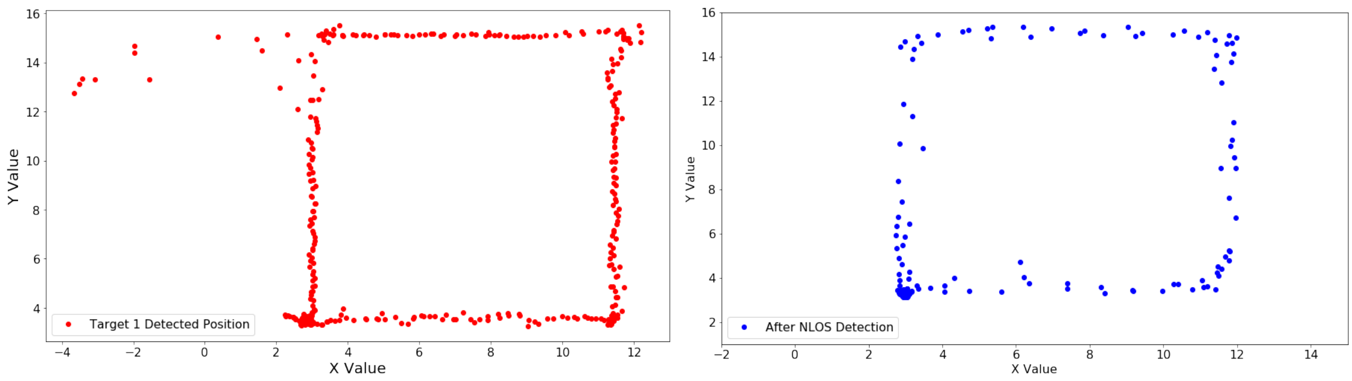
FIGURE 5 Estimation Errors in LOS and NLOS Conditions

TABLE 6 Feature in Existing and Proposed System

Authors (Year)	Ranging	Localization	NLOS Detection	RMSE / Sensitivity
Gururaj, K et al. (2017) [33]	TOF	Trilateration	Waveform Statistics (Mean Excess Delay)	51 cm
Silva, B et al. (2016) [16]	TOF	Trilateration	Waveform Statistics (Kurtosis)	Not Stated
Wymeersch, H et al. (2012) [34]	TOF	4 Methods	Direct Mitigation (SVM and GP Regression)	50 cm (threshold)
Van Nguyen, T et al. (2015) [35]	TOF	Cooperative	Machine Learning (RVM Classifier)	100 cm / 90%
Bregar, K et al. (2016) [23]	TOF	Cooperative	Machine Learning (MLP Classifier)	59 cm / 87%
Our proposed solution	TOF	Dilacak	Machine Learning (Decision-Tree)	41 cm / 91%

5 | CONCLUSIONS

NLOS detection is an unavoidable task for improving the accuracy of the UWB indoor location tracking system. Based on the UWB channel quality indicators, a machine-learning method can be developed for identifying the NLOS condition. In this paper, we proposed an intelligent NLOS detection method using decision-tree learning. We generated an NLOS detection model



(a) Before NLOS Detection

(b) After NLOS Detection

FIGURE 6 Indoor Location Tracking Results: Before vs After NLOS Detection

using the recursive decision-tree learning which divided into two stages. In the first stage, using the Gini index with altered-priors splitting criteria, we try to find the best single feature to split the training dataset into two groups. The training dataset is separated into each sub-group recursively until these subgroups either reach the level of preciseness or until no improvement can be made. In the second stage, we implemented 10-fold cross-validation to check and estimate the risk of the generated model. The important features that were used to build generated model are the Total Channel Impulse Response Power, the all First Path Amplitudes (part 1, part 2, and part 3), the Maximum Noise Value, the Time-of-Flight Report, the Received RX Preamble Symbols, and the Standard Deviation of Noise. Based on our evaluation using several standard metrics, the constructed model has correctly detected around 90% both LOS and NLOS cases on the different indoor environments.

To evaluate the impact of the NLOS detection method in the real world applications, we implemented the proposed algorithm as a part of our existing indoor location tracking system. We run the experiment to locate sixteen moving targets in the FTS indoor training facilities. When the NLOS signal is detected, we send this information to the mitigation and filtering module. This information will be used for either the bias correction or the ranging result elimination. Since the sensitivity of our NLOS detection method is around 91%, the unacceptable location estimation mostly eliminated. For the future of our works, we need to enhance the decision-tree model for detecting multi-states NLOS in the several severity conditions.

ACKNOWLEDGMENTS

This research is supported by "The Project of Conversion by the R&D Results" through the Ministry of Trade, Industry, and Energy (MOTIE) and the Korea Institute for Advancement of Technology (KIAT) (N0002498, 2018).

Author contributions

Ardiansyah Musa conceived the basic idea of the proposed method. It evolved into the final form, as presented in this paper with discussions and contributions from Gde Dharma Nugraha. Ardiansyah Musa, Hyojeong Han, and Gde Dharma Nugraha implemented the experiments in the Force Training System indoor facilities with supervision from Seungho Seo and Juseuk Kim. Finally, Deokjai Choi supervised the logic and managed the overall research.

Conflict of interest

The authors declare no potential conflict of interests.

References

1. Shit RC, Sharma S, Puthal D, Zomaya AY. Location of Things (LoT): A Review and Taxonomy of Sensors Localization in IoT Infrastructure. *IEEE Communications Surveys Tutorials.* 2018;:1-1.

2. Dardari D, Closas P, Djuric PM. Indoor Tracking: Theory, Methods, and Technologies. *IEEE Transactions on Vehicular Technology.* 2015;64(4):1263-1278.

3. Alarifi A, Al-Salman A, Alsaleh M, et al. Ultra Wideband Indoor Positioning Technologies: Analysis and Recent Advances. *Sensors.* 2016;16(5).

4. Xiao J, Zhou Z, Yi Y, Ni L.M. A Survey on Wireless Indoor Localization from the Device Perspective. *ACM Comput. Surv..* 2016;49(2):25:1–25:31.

5. Maduskar D, Tapaswi S. RSSI based adaptive indoor location tracker. *Scientific Phone Apps and Mobile Devices.* 2017;3(1):3.

6. Passafiume M, Maddio S, Cidronali A. An Improved Approach for RSSI-Based only Calibration-Free Real-Time Indoor Localization on IEEE 802.11 and 802.15.4 Wireless Networks. *Sensors.* 2017;17(4).

7. Karapistoli E, Pavlidou F, Gragopoulos I, Tsetsinas I. An overview of the IEEE 802.15.4a Standard. *IEEE Communications Magazine.* 2010;48(1):47-53.

8. Ruiz A. R. Jimenez, Granja F. Seco. Comparing Ubisense, BeSpoon, and DecaWave UWB Location Systems: Indoor Performance Analysis. *IEEE Transactions on Instrumentation and Measurement.* 2017;66(8):2106-2117.

9. Cotera P, Velazquez M, Cruz D, Medina L, Bandala M. Indoor Robot Positioning Using an Enhanced Trilateration Algorithm. *International Journal of Advanced Robotic Systems.* 2016;13(3):110.

10. Andre GF, Fernandes D, Catarino AP, Monteiro JL. Performance Analysis of ToA-Based Positioning Algorithms for Static and Dynamic Targets with Low Ranging Measurements. *Sensors.* 2017;17(8).

11. Ridolfi M, Vandermeeren S, Defraye J, et al. Experimental Evaluation of UWB Indoor Positioning for Sport Postures. *Sensors.* 2018;18(1).

12. Choliz J, Hernandez ÃĄ, Valdovinos A. A Framework for UWB-Based Communication and Location Tracking Systems for Wireless Sensor Networks. *Sensors.* 2011;11(9):9045–9068.

13. Van HT, Verbeke B, De-Poorter E, Moerman I. Optimizing Time-of-Arrival Localization Solutions for Challenging Industrial Environments. *IEEE Transactions on Industrial Informatics.* 2017;13(3):1430-1439.

14. Onalaja O, Adjrad M, Ghavami M. Ultra-wideband-based multilateration technique for indoor localisation. *IET Communications.* 2014;8(10):1800-1809.

15. Musa A, Han H, Nugraha G.D, Choi D, Seo S, Kim J. A Design of Indoor RTLS by Use of the UWB-WSN based Two Reference Points. In: The 2nd International Conference on Applied Electromagnetic Technology (AEMT); 2018.

16. Silva B, Hancke GP. IR-UWB-Based Non-Line-of-Sight Identification in Harsh Environments: Principles and Challenges. *IEEE Transactions on Industrial Informatics.* 2016;12(3):1188-1195.

17. DecaWave Application Node: APS011 version 1.0 https://www.decawave.com/sites/default/files/resources/aps011_sources_of_error_in_twr.pdfLatest accessed: 2018-12-23; .

18. Ferrari P, Flammini A, Sisinni E, et al. Timestamping and Ranging Performance for IEEE 802.15.4 CSS Systems. *IEEE Transactions on Instrumentation and Measurement.* 2014;63(5):1244-1252.

19. Ridolfi M, Velde S, Steendam H, De Poorter E. Analysis of the Scalability of UWB Indoor Localization Solutions for High User Densities. *Sensors.* 2018;18(6).

20. Suwatthikul C, Chantaweesomboon W, Manatrinon S, Athikulwongse K, Kaemarungsi K. Implication of anchor placement on performance of UWB real-time locating system. In: 2017 8th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES):1-6; 2017.

21. Despaux F, Adrien VB, Katia JR, Thierry V. N-TWR: An accurate time-of-flight-based N-ary ranging protocol for Ultra-Wide band. *Ad Hoc Networks.* 2018;79:1 - 19.

22. Sang CL, Adams M, Hormann T, Hesse M, Porrmann M, Rockert U. An Analytical Study of Time of Flight Error Estimation in Two-Way Ranging Methods. In: 2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN):1-8; 2018.

23. Bregar K, Hrovat A, Mohorcic M. NLOS Channel Detection with Multilayer Perceptron in Low-Rate Personal Area Networks for Indoor Localization Accuracy Improvement. In: the 8th Jozef Stefan International Postgraduate School Students Conference; 2016.

24. Kulmer J, Hinteregger S, Grobwindhager B, et al. Using DecaWave UWB transceivers for high-accuracy multipath-assisted indoor positioning. In: 2017 IEEE International Conference on Communications Workshops (ICC Workshops):1239-1245; 2017.

25. UWB LOS and NLOS Data Set https://github.com/ewine-project/UWB-LOS-NLOS-Data-SetLatest accessed: 2018-12-15; .

26. Witten Ian H., Frank Eibe, Hall Mark A., Pal Christopher J.. *Data Mining, Fourth Edition: Practical Machine Learning Tools and Techniques*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 4th ed.2016.

27. Rokach L, Maimon O. *Data Mining with Decision Trees: Theory and Applications*. Series in machine perception and artificial intelligence,World Scientific; 2014.

28. Borowik B, Karpinskyy M, Lahno V, Petrov O. *Theory of Digital Automata*. Intelligent Systems, Control and Automation: Science and Engineering,Springer Netherlands; 2012.

29. Muharram MA, Smith GD. Evolutionary Feature Construction Using Information Gain and Gini Index. In: Genetic Programming:379–388,Springer Berlin Heidelberg; 2004; Berlin, Heidelberg.

30. An Introduction to Recursive Partitioning Using the RPART Routines, 2018 https://cran.r-project.org/web/packages/rpart/vignettes/longintro.pdfLatest accessed: 2019-02-03; .

31. Kulin M, Fortuna C, De-Poorter E, Deschrijver D, Moerman I. Data-Driven Design of Intelligent Wireless Networks: An Overview and Tutorial. *Sensors.* 2016;16(6).

32. FTS System Indoor Forces Training Facilities http://ftssystem.co.kr/eng/index.htmLatest accessed: 2019-01-20; .

33. Gururaj K, Rajendra AK, Song Y, Law CL, Cai G. Real-time identification of NLOS range measurements for enhanced UWB localization. In: 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN):1-7; 2017.

34. Wymeersch H, Marano S, WM Gifford, Win MZ. A Machine Learning Approach to Ranging Error Mitigation for UWB Localization. *IEEE Transactions on Communications.* 2012;60(6):1719-1728.

35. Van Nguyen T, Jeong Y, H Shin, MZ Win. Machine Learning for Wideband Localization. *IEEE Journal on Selected Areas in Communications.* 2015;33(7):1357-1380.

36. DecaWave DW1000: How To Use, Configure, and Program the DW1000 UWB Transceiver https://thetoolchain.com/mirror/dw1000/dw1000_user_manual_2.09.pdfAccessed: 2018-06-20; .

AUTHOR BIOGRAPHY

Ardiansyah Musa. He received the B.Eng in Computer Engineering from Universitas Indonesia in 2010. In 2012, he joined the Advanced Network Lab (ANL), School of Electronics and Computer Engineering, Chonnam National University (CNU), South Korea and obtained his M.Eng in Computer Science in year 2014. Since 2017, he rejoined ANL CNU pursuing Doctoral Degree in Computer Science. His main research interest related to the Future Internet Technology, Sensor Networks, and Machine Learning for Smart Grid and Energy Informatics.

Gde Dharma Nugraha. He received the B.Eng in Electrical Engineering and Master Degree in Computer Engineering from Universitas Indonesia in 2005 and 2009 respectively. Since September 2014, he has been with the Advanced Network Lab, School of Electronics and Computer Engineering, Chonnam National University, Gwangju, South Korea pursuing Doctoral degree in Electronics & Computer Engineering. His interest in research spans from Big Data Processing, Big Data Analytics, Energy Management, Embedded System, and Computing Network.

Hyojeong Han. He is currently a 4th-year undergraduate student in the Major of Software Engineering, School of Electronics and Computer Engineering, Chonnam National University, Gwangju, South Korea. Since the beginning of 2017, he has been with Advanced Network Lab, as a laboratory assistant. His main research interests include Distributed Network Systems, Sensor Networks and Programming for the Internet of Things.

Deokjai Choi. He received the B.S., M.S in Computer Science from Seoul National University, Korea in 1982 and from KAIST 1984 respectively and also received Ph.D. in Computer Science and telecommunication from University of Missouri-Kansas City, USA in 1995. Since 1996 until now, he has been serving as Professor in Advanced Network Lab, School of Electronics and Computer Engineering, Chonnam National University, Korea. His main research interests include topics on Context-awareness, Pervasive Computing, Sensor Networks, and Future Internet.

Seongho Seo. He received the bachelor degree from Honam University, Gwangju, South Korea in 1997. He is currently the Chief Executive Officer (CEO) of the Force Training System (FTS), a firearms training system and simulators provider. His main research interest includes the development of training and simulation system for the military or armed force.

Juseok Kim. He received the bachelor degree from Chosun University, Gwangju, South Korea in 2007. He is currently the technical conductor of the Force Training System (FTS), firearms training system and simulators provider. His main research includes the development of hardware and software equipment for the military or armed force supplies.