A Deep Learning Approach for Localization Systems of High-Speed Objects

Sekchin Chang
Department of Electrical and Computer Engineering, University of Seoul, Seoul, Korea
Corresponding author: Sekchin Chang (e-mail: schang213@uos.ac.kr).

“ This research was supported by Research Program through the National Research Foundation of Korea (NRF) (2017R1A2B4005105).”

ABSTRACT
This paper addresses a novel deep learning technique for localization systems of high-speed mobile objects such as autonomous vehicles. The presented localization method consists of rough and fine localizations. The rough localization exploits the modified Kalman filtering, which produces the rough location estimates of a high-speed object. Due to an inappropriate threshold value, the rough estimates often lead to a divergence in the modified Kalman filtering. In this paper, the fine localization suppresses the divergence. The proposed fine localization is based on a deep learning technique. Using the rough estimates, the deep learning method classifies the current position of the high-speed object into an appropriate region. Based on the classified region, the rough location estimates are refined into the fine location estimates in the fine-localization step. Experimental results verify that the deep learning approach overcomes the weakness of the modified Kalman method in the localization. The results also show that the proposed method outperforms the conventional Kalman approach in the localization of high-speed objects.

INDEX TERMS
Deep Learning, High-Speed Objects, Kalman Filter, Localization

I. INTRODUCTION
Recently, autonomous vehicle systems have emerged as main applications especially in the area of consumer electronics [1]. Moreover, 5G cellular systems and internet-of-things further facilitate autonomous vehicle systems [2], [3]. A localization system is required for proper operation of autonomous vehicles [4]. Since autonomous vehicles move with high speed in outdoor environments, the localization system needs to locate high-speed objects accurately. For accurate localization, localization systems usually rely on time difference of arrival (TDoA) of microwave signals [5]-[8]. For practical positioning, TDoA is also a more desirable measurement than received signal strength (RSS) [9] and direction of arrival (DoA) [10]. However, TDoA-based localization requires a strong line-of-sight (LOS) component for accurate location estimation. In the cases of high-speed objects such as autonomous vehicles, the received microwaves often exhibit non-line-of-sight (NLOS) components due to severe channel effects including deep fading, large Doppler spread, and severe inter-symbol interference (ISI) [11]. Therefore, the NLOS propagation renders the localization performance very poor. Conventional localization systems usually rely on extended Kalman filtering (EKF) [12] since it contributes to accurate tracking of mobile objects in noisy environments. However, even the EKF gives poor tracking performance if the received signal lacks the LOS component. Therefore, the usage of the conventional EKF is not sufficient enough to achieve a desirable tracking performance for high-speed objects such as autonomous vehicles. Some localization does not rely on microwave data (such as TDoA, RSS, and DoA), but inertial data in order to avoid the channel effects. Especially, the localization approach [13] tracks the vehicles fairly well even without any aid of GPS. However, the position estimate is not accurate enough to fully support the autonomous vehicles. For high-speed objects, a modification has been applied to the EKF in order to guarantee excellent localization performance even under the condition that the LOS component is absent in the received signal [14]. The modified EKF method adopts a criterion. The criterion determines if the TDoA information is considerably distorted due to the severe channel effects. Once severe distortion is found, the update process of the EKF discards the contaminated TDoA measurements. The modified EKF exhibits robustness over the detrimental channel effects. However, this criterion approach is based on a predefined threshold. According to the system aging and the channel variations, the threshold should be set to a
new value for reliable criterion. Otherwise, the modified EKF tends to diverge in the tracking because of a wrong decision. Therefore, the modified EKF has a potential divergence if some fixed threshold is used [14].

This paper proposes a novel deep learning technique for localization systems of high-speed mobile objects such as autonomous vehicles. The deep learning approach effectively suppresses the potential divergence of the modified EKF. Various deep learning methods have been applied to main consumer electronics including voice recognizers, automatic translation systems, and real-time object tracking systems [15]. In this paper, the localization system consists of rough and fine localizations. The rough localization employs the modified EKF, which estimates the location of high-speed object roughly. The fine localization relies on the presented deep learning technique in order to effectively suppress a potential divergence of the modified EKF. Among various deep-learning architectures, the fine localization adopts the recurrent neural network (RNN) [16] since it can utilize previous information recurrently. Using the previous location estimates as well as the current estimate, the RNN classifies the current position of the high-speed object into an appropriate region. Then, the fine localization further refines the rough location estimate using the information associated with the classified region. The fine localization completely eliminates a potential divergence of the modified EKF.

Experimental evaluation reveals the validity of the proposed deep learning approach. The experimental results show that the deep learning approach can effectively suppress a potential divergence of the modified EKF in the fine localization. The results also exhibit that the presented scheme is superior to the conventional EKF in terms of tracking performance for high-speed objects.

The contributions of this paper are as follows:

- We incorporated the proposed fine localization into the localization system, which detects and suppresses a potential divergence of the modified EKF. Especially, we proposed the adaptive refinement in the fine localization in order to suppress the divergence. The adaptive refinement varies the threshold of the modified EKF. This suppresses the divergence of the modified EKF.
- In the fine localization, we exploited the RNN in order to detect the divergence. Since the RNN utilizes the previous locations as well as the current location, it is suitable for the region classification of mobile objects. If the RNN returns the inappropriate region, the fine localization detects the divergence.

II. RELATED WORK

For the real-time localization of high-speed objects, the related work mainly relies on the modified EKF [14]. Figure 1 illustrates the service model of the related work. As depicted in Figure 1, the modified EKF consists of prediction step, observation step, and threshold comparison. In the modified EKF, the threshold comparison is added to the conventional EKF, which just includes the prediction step and the observation step [9]. In the conventional EKF, the prediction step theoretically calculates the current position of a mobile object using its previous position. Then, the theoretical position is transferred to the observation step. Using the TDoA measurements, the observation step tries to refine the predicted position. Finally, the refined position is transferred to the prediction step for theoretical calculation of the next position. Therefore, the prediction step interacts with the observation step for tracking of a mobile object. In the case of conventional EKF, the refined position is very close to the true position if the receivers measure the TDoA values very accurately. Furthermore, the refined position is fairly accurate if the TDoA measurements include mild errors. However, the conventional EKF tends to produce distinct distortions in the localization when the TDoA measurements include too heavy errors. Such errors usually occur when the receivers fail in detecting the LOS components in the received signals. The receivers often exhibit the detection failure especially when the mobile object moves with high speed.

![FIGURE 1. The service model of the related work.](image1.png)

![FIGURE 2. The procedure of the threshold comparison.](image2.png)
Therefore, the related work utilizes the modified EKF of Figure 1. As illustrated in Figure 1, the threshold comparison is the unique module in the modified EKF. Figure 2 depicts the procedure of the threshold comparison. In the first step, the threshold-comparison module theoretically calculates TDoA values using the theoretical position, which was transferred from the prediction step. In the second step, the module compares the difference with a pre-defined threshold. If the difference is smaller than the threshold, the observation step uses all the TDoA measurements for the position refinement. Otherwise, the comparison module replaces the TDoA measurements with the theoretical TDoA values. In Figure 2, the larger differences indicate that the TDoA measurements include too heavy errors to refine the position. Therefore, the observation step excludes the heavily noisy TDoA measurements in the position refinement. Finally, the modified EKF suppresses noticeable distortions in the localization.

In the related work, the modified EKF uses a fixed value as the threshold. The experimental results showed fairly good localization performance in various tracks [14]. However, the fixed threshold endangers the localization system, which may produce a divergence especially when the system becomes old or the channel conditions abruptly change. Therefore, the threshold needs to be varied adaptively according to a condition. Accordingly, the position needs to be refined adaptively. A deep learning approach detects the condition for the adaptive threshold and the adaptive refinement, which is described in section III-C in detail.

III. LOCALIZATION SYSTEM BASED ON DEEP LEARNING APPROACH

A. SYSTEM MODEL

Figure 3 illustrates the system model of the localization system based on the proposed deep learning method. The system model consists of transmitter, time-of-arrival (ToA) receivers, TDoA server, rough localization server, and fine localization server. In Figure 3, a high-speed object under consideration employs one transmitter, which generates microwave signals for location estimation. In the figure, each ToA receiver measures a ToA value from the received signal. Therefore, \( N + 1 \) ToA values (\( T_i; i = 0, 1, 2, \ldots, N \)) are measured since there are \( N + 1 \) ToA receivers in Figure 3. Then, the TDoA server calculates \( N \) TDoA values (\( \tau_i; i = 1, 2, \ldots, N \)) from the received \( N + 1 \) ToA values as follows:

\[
\tau_i = T_i - T_0, \quad (1)
\]

where \( \tau_i \) denotes the TDoA value for receiver \( i \) \((i = 1, 2, \ldots, N)\). In (1), \( T_0 \) indicates the reference ToA. Therefore, receiver 0 is used as the reference receiver for TDoA values in Figure 3. Using the \( N \) TDoA values, the rough location server produces a rough estimate. For the rough estimation, the rough location server relies on the modified EKF. As stated earlier, the rough estimates may significantly be deviated from true positions due to a divergence of the modified EKF. To avoid the divergence, the rough estimate is refined into a fine location estimate in the fine location server. The fine server utilizes the presented deep-learning approach, which effectively eliminates the divergence. The suggested deep-learning relies on the RNN for a region classification, which determines if a divergence happens in the rough localization.

\[
\mathbf{F}_{k-1} = \mathbf{F}
\]

\[
\mathbf{P}_{k-1} = \mathbf{F}^T \mathbf{I} + \mathbf{Q}
\]

where \( \mathbf{F} \) denotes the RNN for a region classification, which determines if a divergence happens in the rough localization.
where \( p_{x,k} \) and \( p_{y,k} \) are the x-position and the y-position, respectively of the rough location estimate of high-speed object. In (4), \( v_{x,k} \) and \( v_{y,k} \) denote the x-direction velocity and the y-direction velocity, respectively of high-speed object. In (4), \( a_{x,k} \) and \( a_{y,k} \) are the x-direction acceleration and the y-direction acceleration, respectively of high-speed object. In (3), \( P_k \) denotes the error covariance matrix at the \( k \)th discrete-time index, and the matrix \( Q \) is defined as follows:

\[
Q = E[w_kw_k^H],
\]

where \( w_k \) denotes the process noise vector at the \( k \)th discrete-time index, and \( E[ \cdot ] \) is the operator of ensemble average.

For the operation of the update phase, the observation vector \( y_k \) is defined as follows:

\[
y_k = \mathbf{h}(x_{k|k-1}) + v_k,
\]

where \( \mathbf{h} \) and \( v_k \) denote nonlinear measurement vector and measurement noise vector, respectively. In the update phase, the EKF also performs the steps of (7) to (9) as follows:

\[
G_k = P_{k|k-1}^H (H P_{k|k-1} H^T + R)^{-1}.
\]

\[
P_k = (I - G_k H) P_{k|k-1}.
\]

\[
x_k = x_{k|k-1} + G_k [y_k - \mathbf{h}(x_{k|k-1})].
\]

In (7), the matrices \( H \) and \( R \) are defined in (10) and (11), respectively as follows:

\[
H = \frac{\partial \mathbf{h}(x_{k|k-1})}{\partial x_{k|k-1}},
\]

\[
R_k = E[v_k v_k^T].
\]

While the EKF repeats the prediction phase and the update phase, the updated state vector \( x_k \) of (9) includes the rough location estimate \( (p_{x,k},p_{y,k}) \) of high-speed object at the \( k \)th discrete-time index. When channel conditions are poor, the observation vector \( y_k \) of (6) is heavily contaminated because the measurement noise vector \( v_k \) becomes dominant. This considerably reduces the estimation accuracy.

In order to make the traditional EKF robust over the channel effects, the modified EKF uses the threshold comparison in addition to the traditional procedure of (2) to (11) as follows:

\[
y_{i,k} = \begin{cases} h_i, & \text{if } |ct_i - h_i| > \text{thr} \\ c & \text{otherwise} \end{cases}
\]

where \( y_{i,k} \) denotes the \( i \)th element of the observation vector \( y_k \), \( h_i \) is the \( i \)th element of the matrix \( H \), \( c \) denotes the light speed, and \( i = 1, 2, \ldots, N \). In (12), \( \text{thr} \) denotes the predefined threshold. In (12), \( h_i \) and \( ct_i \) are the prediction and the observation values, respectively. If the observation value includes a heavy noise, the difference between \( ct_i \) and \( h_i \) becomes larger than the threshold value. In this case, the prediction value \( (h_i) \) replaces the observation value \( (ct_i) \) as indicated in (12). This prevents the reduction of estimation accuracy. However, the difference may be larger than the threshold due to system aging and channel variations even in the cases of good observations. In this case, the prediction value \( (h_i) \) is too excessively used in (12). This may cause a severe divergence in the location estimation of (9). In order to prevent the divergence, the threshold needs to be varied adaptively in (12). The fine localization performs the adaptive variation of the threshold.

**C. FINE LOCALIZATION**

![Figure 4](image)

The fine localization refines the rough estimates in order to avoid a potential divergence of the modified EKF. The fine localization relies on the RNN (which belongs to the deep learning category) in order to determine if the potential divergence occurs in the procedure of the modified EKF. Figure 4 illustrates the architecture of the RNN for the fine localization. The RNN refines the rough location.
estimate into an appropriate region. In Figure 4, the current input vector \( s_k \) includes the current location estimate at the \( k \)th discrete-time index as follows:

\[
s_k = [p_{x,k} \ p_{y,k}]^T. \tag{13}
\]

In the RNN of Figure 4, the previous input vector \( \tilde{s}_{k-1} \) is defined as follows:

\[
\tilde{s}_{k-1} = [p_{x,k-l} \ p_{y,k-l}]^T, \tag{14}
\]

where \( p_{x,k-l} \) and \( p_{y,k-l} \) are the \( x \)-position and the \( y \)-position, respectively of the previous location estimate of high-speed object at the \((k-l)\)th discrete-time index. In (14) \( l = 1, 2, \ldots, L \), where \( L \) denotes the memory value of the RNN. In Figure 4, the output signal \( r_{k,c} \) denotes the value of the \( c \)th class region, and is calculated as follows:

\[
r_{k,c} = f(\tilde{r}_{k,c}), \tag{15}
\]

where \( c = 1, 2, \ldots, C \). In (15) and Figure 4, \( C \) denotes the number of class regions. In (15), \( f(\cdot) \) denotes the logistic function of the RNN, and is defined as follows:

\[
f(x) = \frac{1}{1 + e^{-x}}. \tag{16}
\]

In (15), the intermediate output signal \( \tilde{r}_{k,c} \) consists in the intermediate output vector \( \tilde{r}_k \) as follows:

\[
\tilde{r}_k = [\tilde{r}_{k,1} \ \tilde{r}_{k,2} \ \cdots \ \tilde{r}_{k,C}]^T. \tag{17}
\]

In (17), \( \tilde{r}_k \) can be achieved as follows:

\[
\tilde{r}_k = Wd_k, \tag{18}
\]

where \( W \) denotes the \( C \times J \) weight matrix of the (final) hidden-to-output layer in Figure 4. In (18), the final-hidden-layer vector \( d_k \) includes the elements of the final hidden layer as follows:

\[
d_k = [f(d_{1,k}) \ f(d_{2,k}) \ \cdots \ f(d_{J,k})]^T, \tag{19}
\]

where \( J \) denotes the number of the elements of the vector \( d_k \). In (19), the elements \( \tilde{d}_{1,k}, \tilde{d}_{2,k}, \ldots, \tilde{d}_{J,k} \) constitute in the intermediate vector \( \tilde{d}_k \), which is calculated as follows:

\[
\tilde{d}_k = Ms_k + Ud_{k-1}, \tag{20}
\]

where \( M \) and \( U \) denote the \( J \times 2 \) matrix of the input-to-hidden layer and the \( J \times J \) matrix of the previous-hidden-to-final-hidden layer, respectively. In (20), the previous-hidden-layer vector \( \tilde{d}_{k-1} \) is achieved from the intermediate previous-vector \( \tilde{d}_{k-1} \) as in (19). At the \((k-l)\)th discrete-time index, the intermediate previous-vector \( \tilde{d}_{k-1} \) is calculated as follows:

\[
\tilde{d}_{k-1} = Ms_{k-1} + Ud_{k-2-l}, \tag{21}
\]

where \( l = 1, 2, \ldots, L \). Note that the previous location estimate and the current location estimate are used in the previous input vector \( \tilde{s}_{k-l} \) of (21) and the current input vector \( s_k \) of (20), respectively. In Figure 4, the matrices \( W, M, \) and \( U \) are trained using the backpropagation through time (BPTT) algorithm [17].

FIGURE 5. The procedure of the fine localization based on the RNN.

Figure 5 illustrates the procedure of the fine localization based on the RNN of Figure 4. Using the current and the previous location estimates, the RNN determines the class region to which the high-speed object belongs. Using the
output signals \((r_{k,1}, r_{k,2}, \ldots, r_{k,C})\) of the RNN, the class region is selected as follows:

\[
\hat{c} = \text{arg max } \left[ r_{k,1}, r_{k,2}, \ldots, r_{k,C} \right],
\]

where \(\hat{c}\) denotes the index of the selected class region. As indicated in (22), the RNN determines the region corresponding to the maximum output as the selected class to which the high-speed object belongs. As indicated in Figure 5, the fine localization checks if the selected class of the RNN is the right track region. If the right region is true (\(Y\) in Figure 5: normal mode), the rough estimate is just set to the fine location estimate. Otherwise (\(N\) in Figure 5: divergent mode), the fine localization performs the adaptive refinement of the location estimate. Then, the refined estimate is used as the input vector to the RNN. The RNN determines the class region of the refined estimate. If the class implies the right track region (\(Y\) in Figure 5), the refined estimate is set to the fine location estimate. Otherwise (\(N\) in Figure 5), the fine localization continues the adaptive refinement of the location estimate until the RNN produces the right track region.

### IV. EXPERIMENTAL EVALUATION

Experimental results exhibit the effectiveness of the proposed deep learning approach for localization systems.

<table>
<thead>
<tr>
<th>Item</th>
<th>Property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum speed</td>
<td>60 km/h</td>
</tr>
<tr>
<td>Signal type for TDoA</td>
<td>UWB</td>
</tr>
<tr>
<td>Number of generated signals each 1s</td>
<td>30</td>
</tr>
<tr>
<td>Number of TDoA measurements</td>
<td>6</td>
</tr>
<tr>
<td>Protocol for transmitter and receiver</td>
<td>IEEE 802.15.4a</td>
</tr>
<tr>
<td>Number of tracks</td>
<td>3</td>
</tr>
</tbody>
</table>

Table I summarizes the experimental environments for the proposed method. Table I indicates that high-speed objects move with the maximum speed of 60 km/h in this evaluation. Each mobile object is equipped with a consumer transmitter [14]. The transmitter generates ultra-wideband (UWB) signals [18] in order to increase the time resolution of TDoA measurements. Therefore, the consumer receivers [14] measure the TDoA values from the received UWB signals. Table I also depicts that the transmitter generates 30 UWB signals each 1 s for accurate localization of high-speed objects. Table I reveals that 6 TDoA measurements are used in the localization. Table I also depicts that the consumer transmitter and receiver follow the IEEE802.15.4a standard [19] for the UWB signals. Table I indicates that this evaluation relies on 3 tracks for modeling of the roads on which high-speed objects move. In this evaluation, the tracks model the roads of 1000 m, 1300 m, and 1900 m. In (23), \(\alpha\) is set to 1.5.

![FIGURE 7. The consumer transmitter and its chip set](image_url)

Figure 7(a) and (b) illustrate the consumer transmitter and its chip-set, respectively for the experimental evaluation. The weight of the transmitter is just 35 g. Therefore, it can easily be attached to the mobile object for the localization. The consumer transmitter includes the single chip-set of Figure 7(b). The chip set is the DW1000 of DecaWave [20]. The DW1000 is a fully integrated single chip-set and is compliant to IEEE802.15.4a for UWB.
signals [19]. Therefore, the chip-set is widely utilized for TDoA-based localization systems.

Figure 8 illustrates the 4 receivers out of the consumer receivers for the experimental evaluation. As illustrated in Figure 8, the receivers are installed on high tower, which increases the detection probability of the UWB signals. As stated earlier, the receivers follow the IEEE802.15.4a standard for the UWB signals [19]. Note that lots of receivers (more than 50 receivers) are installed around the experimental tracks for reliable measurement of TDoA values.

In this experimental evaluation, the RNN adopts the structure of Figure 4. Table II summarizes the parameters (including the layer number) of the RNN.

![Figure 8. The consumer receives](image1)

**TABLE II**  
THE PARAMETERS OF THE RNN

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value/Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of the elements of input vector</td>
<td>$s_1$</td>
</tr>
<tr>
<td>Number of outputs (class regions): $C$</td>
<td>3</td>
</tr>
<tr>
<td>Number of layers ($L$)</td>
<td>5</td>
</tr>
<tr>
<td>Dimension of weight matrix $W$</td>
<td>$3 \times 50$</td>
</tr>
<tr>
<td>Dimension of weight matrix $M$</td>
<td>$50 \times 2$</td>
</tr>
<tr>
<td>Dimension of weight matrix $U$</td>
<td>$50 \times 50$</td>
</tr>
</tbody>
</table>

We used real-world data for training and evaluation in this paper. In other words, we made a real-world experiment. We achieved the real-world distances based on TDoA information from a racing stadium. The commercial transmitter of Figure 3 sends 30 UWB signals within 1s. Therefore, the localization system of Figure 3 produces one fine-location estimate every 1/30 s for real-time operations. In Figure 3, the location servers indicate the computer server system that estimates the location of the mobile object. The computer server system includes the modified EKF and the proposed fine localization (including RNN). The computer server system is capable enough to estimate the location of the mobile object within 1/30 s. In this evaluation, we made an off-line training for the RNN. For the RNN training, we used the real-world data. After we had already completed the RNN training, we used the trained RNN for the fine localization in real-time mode. Since the real-time localization does not include the training procedure, it does not have any overhead for the training.

As stated earlier, we made an off-line training for the RNN. As the training data, we used the real-world data. Figure 9 illustrates the class regions that the deep learning approach produces. In Figure 9, class regions I, II, and III denote on-track domain, upper off-track domain, and lower off-track domain, respectively. The upper off-track includes the area outside the upper boundary of the track. The lower off-track encompasses the area outside the lower boundary of the track. Therefore, class regions II and III indicate that the localization of mobile objects is in the divergent mode. On the other hand, class region I implies that the localization is in the normal mode. Using the rough location estimates, the deep learning architecture (RNN) of Figure 4 determines the mode (normal mode or divergent mode) of the rough localization as stated earlier. In Figure 4, the matrices $W$, $M$, and $U$ are trained for the 3 tracks (1000 m, 1300 m, and 1900 m).

![Figure 9. The class regions of the deep learning](image2)

**TABLE III**  
THE REAL-WORLD DATA OF CONSUMER TRANSMITTER AND RECEIVERS

<table>
<thead>
<tr>
<th>Time stamps (s)</th>
<th>Receiver ID</th>
<th>Distances based on TDoA (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>140134.793781</td>
<td>51</td>
<td>13.7281</td>
</tr>
<tr>
<td>140134.793781</td>
<td>52</td>
<td>19.3629</td>
</tr>
<tr>
<td>140134.793781</td>
<td>36</td>
<td>28.1693</td>
</tr>
<tr>
<td>140134.793781</td>
<td>10</td>
<td>100.3472</td>
</tr>
<tr>
<td>140134.793781</td>
<td>18</td>
<td>116.5290</td>
</tr>
<tr>
<td>140134.793781</td>
<td>1</td>
<td>146.0965</td>
</tr>
<tr>
<td>140134.826560</td>
<td>51</td>
<td>14.4084</td>
</tr>
<tr>
<td>140134.826560</td>
<td>52</td>
<td>19.2362</td>
</tr>
<tr>
<td>140134.826560</td>
<td>36</td>
<td>28.1787</td>
</tr>
<tr>
<td>140134.826560</td>
<td>10</td>
<td>100.3753</td>
</tr>
<tr>
<td>140134.826560</td>
<td>18</td>
<td>116.6416</td>
</tr>
<tr>
<td>140134.826560</td>
<td>1</td>
<td>146.0026</td>
</tr>
</tbody>
</table>

Table III exhibits the real-world data of the consumer transmitter [Figure 7(a)] and the consumer receivers (Figure 8) at 2 time stamps. In Table III, the time column includes the time stamps of the consumer transmitter. The transmitter sends the UWB signal at the time stamp. Note that the time interval between the time stamps is 1/30 s. This indicates that the consumer transmitter sends 30 UWB signals each 1 s as stated earlier. In Table III, the receiver ID denotes the identification number of the receiver. As shown in Table III, 6 receivers are used at each time stamp. This indicates that the localization exploits the 6 TDoA
measurements as shown in Table I. In Table III, the column of distances includes the measured distance between the high-speed object and each receiver. The distance is calculated using the measured TDoA of (1) in each receiver. Note that Table III exhibits the partial real-world data (at only 2 time stamps). In Figures 10 to 13, the location estimates were achieved using the entire real-world data.

Experimental results are illustrated in Figures 10 to 13. Figure 10 indicates the case that the modified EKF successfully estimates the locations of the high-speed object under test. Figures 11 to 13 include the cases that the modified EKF is divergent. Therefore, Figures 11 to 13 also reveal the effectiveness of the presented deep learning method. The real-world TDoA measurements already include the noise values due to thermal noise, fading, and interference. The noise values heavily influenced the localization performance for the conventional EKF. The influences will be seen in Figure 10(b), Figure 11(a), Figure 12(b), and Figure 13(b).

Figure 10 exhibits a comparison of the location estimates for the track of 1000 m in the case of no divergence in the modified EKF. Since there is no divergence, this evaluation does not require the use of the fine localization based on the deep learning. Therefore, Figure 10 does not include the experimental results of the fine localization. Fig. 10(a) illustrates the track of 1000 m. In Figure 10(a), D and A represent departure and arrival, respectively of the high-speed object under test. Figure 10(b) shows the location estimates in the case of the conventional EKF. As illustrated in Figure 10(b), the conventional EKF generates noticeable distortions in the localization. This considerably reduces the estimation accuracy. In the conventional EKF, excessively inaccurate TDoA measurements cause such distortions. Figure 10(c) exhibits the location estimates in the modified EKF case. Figure 10(c) reveals that the modified EKF eliminated the distortions almost completely. Using the threshold comparison of (12), the modified EKF successfully excluded the excessively inaccurate TDoA measurements. This indicates that the modified EKF did not cause any divergence even with the fixed threshold of (12). However, the fixed threshold often causes severe divergence as shown in Figures 11 to 13.

Figure 11 exhibits a comparison of the location estimates for the track of 1000 m in the case of divergence in the modified EKF. Figure 11(a) exhibits the location estimates in the conventional EKF case. Like the case in Figure 10(b), Figure 11(a) also shows some distortions in the tracking estimation. Figure 11(b) exhibits the location estimates in the modified EKF case. Note that the rough localization employs the modified EKF. Figure 11(b) illustrates a divergence of location estimates in the rough localization. This is different from the case of Figure 10(c). Therefore, it is revealed that the threshold comparison of (12) does not always guarantee reliable estimation of tracking even in the same track. Figure 11(c) illustrates a comparison of tracking divergence and true tracking. Figure 11(c) indicates that the divergence approximately occurs at the position of (-412, 80). Figure 11(d) shows the location estimates in the case that the final localization is employed. As the first step, the final localization investigates the class regions of the high-speed object under test. In this evaluation, the deep learning approach finds that the rough location estimates are in the divergent mode after the coordinate (-412, 80) has been passed. After detecting the divergent mode, the final localization performs the adaptive refinement of Figure 6. This refines the divergent rough-estimate into the normal fine-estimate using the adaptive threshold of (23). As illustrated in Figure 11(d), the final localization effectively suppresses the huge divergence using the deep learning method. Figure 11(d) also indicates that the deep learning technique significantly contributes to a reliable tracking in the track of 1000 m.

**FIGURE 10.** The comparison of the location estimates for the track of 1000 m in the case of no divergence in the modified EKF.
Figure 11 includes a comparison of the location estimates for the track of 1300 m in the case of divergence in the modified EKF. Figure 12(a) illustrates the track of 1300 m. Figure 12(b) exhibits the location estimates in the conventional EKF case. As shown in Figure 12(b), the conventional EKF also renders apparent distortions in the tracking estimation. Figure 12(c) exhibits the location estimates in the modified EKF case. Figure 12(c) also reveals that a serious divergence occurs in the tracking. This indicates that the threshold comparison of (12) also failed a reliable localization in the track of 1300 m. Figure 12(d) shows the location estimates in the case that the final localization is employed. Figure 12(d) also reveals that the final localization can completely eliminate the profound divergence using the deep learning approach in the track of 1300 m.
Figure 13 shows a comparison of the location estimates for the track of 1900 m in the case of divergence in the modified EKF. Figure 13(a) illustrates the track of 1900 m. Figure 13(b) shows the location estimates in the conventional EKF case. As exhibited in Figure 13(b), the conventional EKF also renders distinct distortions in the localization. Figure 13(c) shows the location estimates in the modified EKF case. Figure 13(c) also reveals that a detrimental divergence occurs in the localization. This implies that the modified EKF failed in achieving a reliable localization using the threshold comparison of (12) in the track of 1900 m. Figure 13(d) exhibits the location estimates in the final localization case. Figure 13(d) also reveals that the final localization can entirely remove the enormous divergence using the deep learning method in the track of 1900 m.

As stated earlier, we achieved the real-world data from the racing stadium. The date that we achieved the data for Figures 11 to 13 is different from the date of Figure 10. The channel states usually change according to weather, moisture and system conditions. Therefore, the channel state of Figure 10 is different from those of Figures 11 to 13. In the evaluation, the fixed threshold is suitable for the channel state of Figure 10. However, the threshold is not suitable for the channel states of Figures 11 to 13.

In addition to the illustrations of Figure 11 to Figure 13, we empirically evaluated the 50, 30, and 10 divergence cases in tracks of 1000 m, 1300 m, and 1900 m, respectively. The fine localization entirely suppressed the divergences in all the cases. This indicates that the proposed fine localization can completely eliminate a potential divergence of the modified EKF. As further evaluation, we present the evaluation performance in terms of the root mean-squared error (RMSE) between the reference positions and the position estimates [13], [21]. As shown in Table IV, the proposed approach outperforms the conventional EKF and the modified EKF. Note that the modified EKF exhibits the worst performance since it renders such divergence.

<table>
<thead>
<tr>
<th>Proposed</th>
<th>Modified EKF</th>
<th>Conventional EKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.8 m</td>
<td>427.6 m</td>
<td>18.2 m</td>
</tr>
</tbody>
</table>

V. CONCLUSION

This paper presents a novel deep learning technique for localization systems, which are required for real-time localization of high-speed mobile objects such as autonomous vehicles. In this paper, the localization systems consist of rough and fine localizations. In the localization systems, the rough localization relies on the modified EKF, which employs a threshold comparison in order to enhance the localization performance. However, the threshold
comparison uses a pre-defined threshold value. This may cause the rough localization to be divergent due to system aging and channel variations. In order to effectively suppress a potential divergence of the rough localization, the fine localization employs the proposed deep learning approach. The deep learning technique is based on the RNN architecture. The RNN is very useful for localization since it can utilize the previous location estimates as well as the current location estimate. Based on the RNN architecture, the presented deep learning method determines the tracking mode of the high-speed object: normal mode or divergent mode. If the divergent mode is detected, the fine localization refines the rough location estimate using the adaptive refinement. Therefore, the fine localization can completely eliminate a potential divergence of the rough localization.

Experimental evaluation confirms the validity of the proposed deep learning approach for localization of high-speed objects. The experimental results reveal that the fine localization can effectively suppress a potential divergence of the rough localization using the deep learning approach. The results also exhibit that the presented method is superior to the conventional EKF as well as the modified EKF in terms of localization performance for various tracks. Finally, it is concluded that the proposed deep learning technique is very useful for localization systems of high-speed objects including autonomous vehicles.

REFERENCES