

Received April 29, 2020, accepted May 10, 2020, date of publication May 14, 2020, date of current version June 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2994281

# In-Air Continuous Writing Using UWB Impulse Radar Sensors

FAHEEM KHAN<sup>1</sup>, SEONG KYU LEEM<sup>1</sup>, (Graduate Student Member, IEEE),  
AND SUNG HO CHO<sup>1</sup>, (Member, IEEE)

Department of Electronics and Computer Engineering, Hanyang University, Seoul 04763, South Korea

Corresponding author: Sung Ho Cho (dragon@hanyang.ac.kr)

This work was supported by the Bio and Medical Technology Development Program of the National Research Foundation (NRF) and funded by the Korean Government (MSIT) under Grant NRF-2017M3A9E2064626.

**ABSTRACT** We developed an impulse radio ultra-wideband (IR-UWB) radar-based system that can recognize alphanumeric characters in midair without the need for any handheld device. The hardware consists of four IR-UWB radar sensors set up with a rectangular geometry. Writing a single character in midair results in artifacts that make some characters look similar on a position trajectory-based  $(x, y)$  plane, which makes them difficult to classify. Thus, we developed an algorithm that transforms 2D coordinate image data into trigonometric ratios (i.e., tangents) and plots them against the time axis to obtain unique images for training a convolutional neural network. An extended Kalman filter is used to obtain the 2D trajectories of hand motions. To evaluate our proposed method, we first applied it to characters that may be written in midair very simply without creating artifacts and compared its performance with that of a state-of-the-art digit classification algorithm. Then, we considered combining characters written midair with and without artifacts. After the individual character recognition, we combined the characters into words. We defined a specific marker based on an energy threshold to detect the start and end of a character for midair writing. The energy level was found to change drastically when the hand is pulled in and out of the radar plane. The proposed method was found to outperform the current state of the art at character classification when artifacts are present in the images.

**INDEX TERMS** alphabet writing, gesture recognition, impulse radio ultra-wideband, in-air writing, pattern analysis.

## I. INTRODUCTION

Gesture recognition allows a user to comfortably interact with a computer or other consumer electronic device for entertainment and/or communication without physical contact or voice commands. Different sensors have been considered for gesture recognition, such as cameras [1], gloves [2], and radiofrequency identification (RFID) [3]. However, sensors that are attached to the body [4] are often uncomfortable for the user, and vision-based sensors [5] suffer from privacy issues and do not work efficiently in dark or extremely bright environments. Radar-based gesture recognition has no privacy issues and can work well in different environments with various levels of illumination [6]. The impulse radio ultra-wideband (IR-UWB) approach is characterized by the

emission of extremely short pulses with very low power and no harmful effects on the human body. Thus, it can use a large part of the radio spectrum without disturbing the narrowband systems that already operate in different frequency bands. Other benefits of this approach are its robustness in harsh environments, high precision ranging, low power consumption, and high penetration capabilities [7]. IR-UWB has been used in many applications, such as multi-human detection [8], people counting [9], vital sign monitoring [6], [10]–[15], 3D positioning [16], gesture recognition [6], [17], [18], human–computer interaction for disabled people [19], and digital menu board implementation [20]. Although some studies have considered radar-based gesture recognition [21], [22], they used raw data such as spectrograms. Leem *et al.* [23] used an IR-UWB radar sensor and hand trajectories instead of raw data to recognize digits; however, they only considered simply written numeric

The associate editor coordinating the review of this manuscript and approving it for publication was Weimin Huang<sup>1</sup>.

characters that did not result in any artifacts, unlike in alphabetic writing, and used an already available handwritten dataset in the image processing field to train their convolutional neural network (CNN). However, midair alphabetic writing in different styles results in different artifacts because these characters are written in continuous fashion, which makes the resulting trajectories differ from those for characters written with a pen. Because of the artifacts, some characters may produce similar patterns that make them difficult to distinguish (e.g., “5” vs. “6” or “a” vs. “b”) when only the position trajectory is considered, which reduces the overall recognition accuracy. In this study, we incorporated temporal information between radar pulses (i.e., slow time in the radar literature) with 2D localization information to get the real-time trajectory and writing style for a particular alphanumeric character. Therefore, even if some characters have similar shapes on a position trajectory-based  $(x, y)$  image, they produce different patterns when the temporal information is included in the  $(x, y, t)$  image.

## II. PROBLEM STATEMENT AND RELATED WORK

In this study, we considered English alphanumeric characters for in-air writing. Some characters cannot be written continuously on paper without lifting the pen up and then down (e.g., “X,” “F”). In the case of midair alphabetic writing, however, the tracking algorithm continuously monitors the motion of the hand, which results in artifacts. A previous issue with radar-based gesture recognition using deep learning was that the raw data would change abruptly as the orientation or distance of the hand from the radar sensor changed, which reduced the accuracy [22]. The current state of the art of radar-based in-air handwriting [23] solved this problem by using the trajectory of the hand instead of raw data and then employing a CNN for classification. However, Leem *et al.* [23] only considered the numeric digits of 0–9 and did not discuss artifacts that may occur during in-air writing. However, writing complex alphabetic characters may cause artifacts where the real-time trajectory differs from the original character written with a pen on paper. Fig. 1 gives two examples: the characters “X” and “F.” The black lines show the trajectory that is the same as the character written on paper, while the red lines show the trajectories that result in artifacts (bc in Fig. 1(a) and cd in Fig. 1(b)). In this study, we examined how some of these artifacts generate similar patterns for different characters and thus reduce the classification accuracy. The main contributions of this work are as follows. It is the first to address the problem of artifacts that occur during midair writing using radar sensors. It is the first study on continuous in-air writing using radar sensors. In addition, we optimized our own CNN for radar-based image classification, which has a simpler structure than widely used pre-trained CNNs. We verified our results through the leave-one-person-out cross-validation (LOPO-CV) scheme, where one user is excluded from the training data. The objectives of this study were as follows:

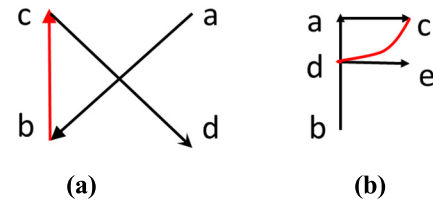


FIGURE 1. In-air handwritten alphabetic characters: (a) “X” and (b) “F.”



FIGURE 2. Experimental setup for the proposed method.

1. To solve the problem of artifacts related to in-air character writing.
2. To consider continuous character writing. Previous studies only classified individual characters, but in this study we used an energy threshold algorithm to segment the stream of radar data into blocks and then applied localization and classification algorithms to detect individual characters.

## III. PROPOSED METHOD FOR CHARACTER RECOGNITION

We used a setup consisting of four radar sensors, which were placed as shown in Fig. 2. Characters are written by hand on the plane set up by the four sensors. We used four sensors rather than three because each sensor had a narrow beam width (around  $60^\circ$ ). Covering the whole plane with only three sensors was difficult and led to low accuracy because the hand gestures sometimes did not occur within the beam widths of the transceivers, which reduced the radar cross-section (RCS) values. Using four sensors improved the recognition accuracy because of the diversity effect. Note that only one character was written on the plane at a time. The writing was continuous in that one character was followed by another, but they shared the same space. As stated above, the two main objectives of this study were to classify characters individually and detect the exact intervals within which characters are written.

Fig. 3 shows the block diagram of our proposed method for detecting continuous handwriting. After the raw data are obtained from the radar sensors, which are actually the signal reflected from the hand and the background environment, the static clutter due to the background signal needs to be removed. Then, the index of the maximum magnitude sample

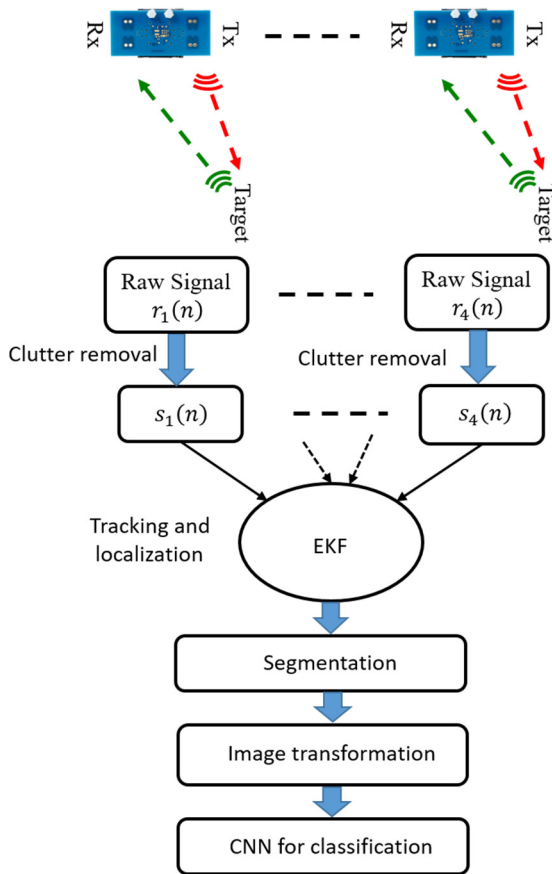


FIGURE 3. Block diagram of the proposed method for character recognition.

needs to be identified for each slow time signal component. This process is repeated for the whole slow time duration of the gesture. An EKF with a median filter is used to get the position trajectory of the hand during a gesture. Since hand tracking using trilateration technique is a non-linear problem, so the EKF gives optimal results compared to classical KF. A position velocity (PV) model is used to model the hand motion. The median filter is used to remove outlier values before the EKF step. After the trajectory is determined, the tangent of the  $x, y$  data is found, and the tangent ratio is plotted against the time axis. The main reason for using the trigonometric ratio instead of  $(x, y)$  coordinate data is that we can easily plot the ratio along slow tune without adding additional axis. The resulting trigonometric ratio plot against slow time contains the writing style information which improves the classification accuracy for characters with artifacts. The stored images are then processed to be compatible with the CNN, which is used to classify the pattern of each gesture corresponding to a specific character. We used a simple architecture for the CNN because the images are not very complex. We fine-tuned the hyper parameters for the CNN structure to ensure fast and accurate character recognition. Because our focus was on continuous writing, we also developed a technique for detecting characters from a continuous stream that

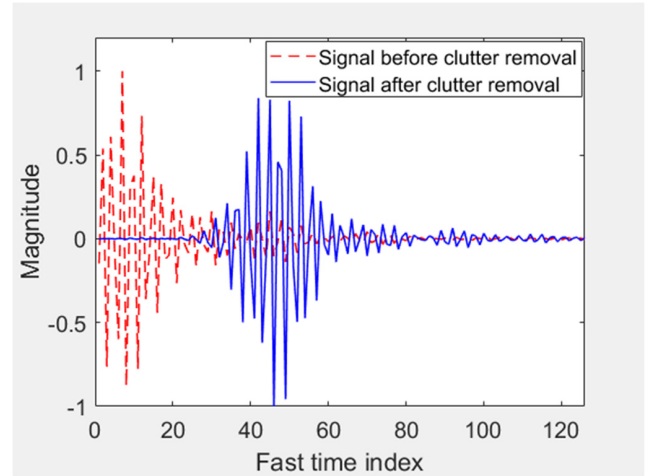


FIGURE 4. Hand-reflected signals before and after clutter removal.

uses a marker for the start and end of individual characters. This technique is based on the principle that, if the user’s hand is inside the plane of the radar sensors, then its RCS will be greater (i.e., higher energy), while a hand position outside the plane will result in a smaller RCS (i.e., lower energy). Hence, an energy threshold can be set during the training period. Each step is discussed in detail in the following sections.

**A. CLUTTER REMOVAL**

The signal reflected from the hand contains information on the gesture as well as the background. We used a background subtraction filter to remove unwanted echoes (i.e., clutter) [24]. The simple loopback filter is represented as follows:

$$c_m(n) = \alpha c_{m-1}(n) + (1 - \alpha) r_m(n) \tag{1}$$

$$s_m(n) = r_m(n) - c_m(n) \tag{2}$$

where  $m$  is the slow time index,  $n$  is the fast time index,  $\alpha$  is the estimated ratio of signal to clutter,  $c_m(n)$  is the clutter signal,  $s_m(n)$  is the signal from which the clutter signal is removed, and  $\alpha$  is the weighting constant that controls the sensitivity of the clutter removal process. We set  $\alpha$  to 0.85 in our experiments. Fig. 4 shows the signal before and after clutter removal. The normalized values of the signal amplitude in the fast time range (i.e., within a radar pulse) are shown for easy comparison. The signal before clutter removal is represented by a dotted red line and initially had higher values (samples 1–25), which indicates the clutter signal. The signal after clutter removal is represented by a solid blue line, where the main signal due to the hand gesture is amplified around sample 48.

**B. POSITIONING WITH THE EXTENDED KALMAN FILTER**

The input signals from the four radar sensors are represented by  $r_1(n), r_2(n), r_3(n)$ , and  $r_4(n)$ , respectively. The clutter-free signals  $s_1(n), s_2(n), s_3(n)$ , and  $s_4(n)$  are obtained with the background subtraction filter described

in Section III-A. We used the time-of-arrival (TOA) of the hand with respect to each radar sensor as the index value for the maximum magnitude in slow time. After the TOA is estimated for the four radar sensors, we use the EKF to track the hand in midair, which we implemented in a PV model. The detailed algorithm for EKF-based positioning using multiple sensors is given by Khan *et al.* [25]. The variables for the state space representation of the EKF are defined as follows. The state vector for the PV model is

$$x_k = [x_k, y_k, v_{x_k}, v_{y_k}] \quad (3)$$

The state transition matrix is

$$F = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

The observation vector  $z_k$  for 2D space is

$$z_k = [d_1, d_2, d_3, d_4] \quad (5)$$

The relation between the distances and coordinates of the target and radar sensors is given by

$$d_i = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \quad i = 1, \dots, 4 \quad (6)$$

where  $d_i$  is the distance from the  $i^{\text{th}}$  radar sensor to the hand and  $(x_i, y_i)$  is the position of the radar sensor. The objective is to estimate the hand position  $(x, y)$  from the noisy observation. Because the update state is nonlinear, it needs to be linearized with the following Jacobian matrix  $H_k$ :

$$H_k = \begin{bmatrix} (x_k - x_1)/(d_1) & (y_k - y_1)/(d_1) & 0 & 0 \\ (x_k - x_2)/(d_2) & (y_k - y_2)/(d_2) & 0 & 0 \\ (x_k - x_3)/(d_3) & (y_k - y_3)/(d_3) & 0 & 0 \\ (x_k - x_4)/(d_4) & (y_k - y_4)/(d_4) & 0 & 0 \end{bmatrix} \quad (7)$$

Applying the EKF to the position data obtains the trajectory of the hand motion for in-air writing. Fig. 5 shows the localization results for the digit “6” and character “b.” These characters make similar 2D patterns on the  $(x, y)$  plane that are difficult to classify using only  $x, y$  coordinate data. Although the patterns look similar, they are created differently. The digit “6” is usually drawn with the bottom circle counterclockwise in the order shown by the green arrows in Fig. 5(a), while the character “b” is drawn with the bottom circle clockwise in the order shown by the green arrows in Fig. 5(b). In other words, the trajectories are drawn in different orders. With radar, trajectories are obtained sequentially according to time (i.e., the slow time index), so it is easy to determine what order trajectories are written according to time. Based on these radar properties, two characters that look similar but differ in the order in which they are written can be differentiated.

We constructed images by using our proposed method given in Section III.C, which we plotted in two different ways: the  $x$  coordinate vs slow time and  $y$  coordinate vs slow time. Fig. 6 indicates a clear difference between the images even to the naked eye. The patterns differ especially after sample 35,

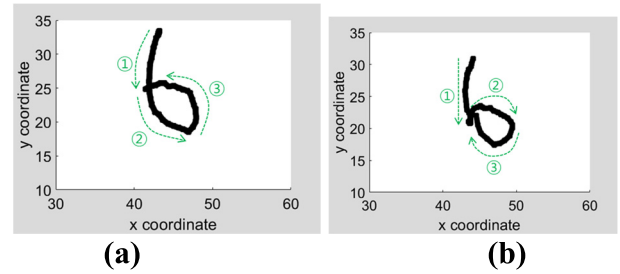


FIGURE 5. 2D images of similar-looking characters: (a) “6” and (b) “b.”

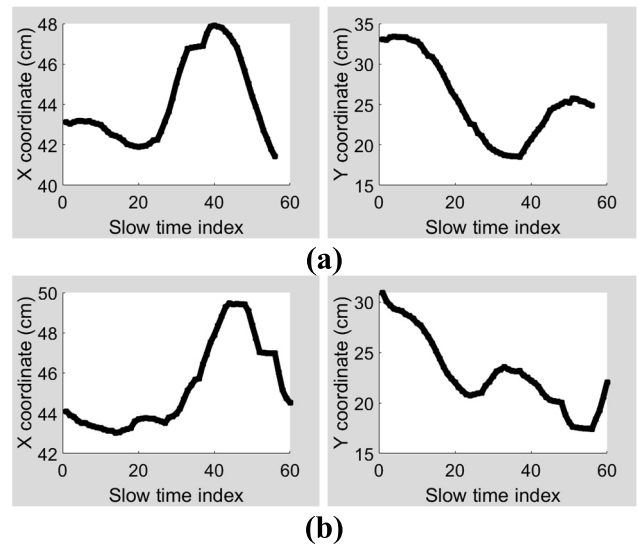


FIGURE 6. Images of the characters (a) “6” and (b) “b,” which look different when the  $x, y,$  and  $t$  coordinates are considered.

which represents the bottom portion of these characters being written. Based on our observations, we developed an image transformation method that includes both the  $(x, y)$  coordinates and time ( $t$ ) of the positioning data. In the following section, we explain the image transformation method in detail with an example.

### C. IMAGE TRANSFORMATION FROM $(X, Y)$ TO $(Y/X, T)$

We need to create an image that uses both positioning data (i.e.,  $x$  and  $y$  coordinates) and slow time data. However, the three variables cannot be plotted simultaneously on a 2D image, which has only two axes. For three-dimensional representation, trajectory information within a dataset is so rare that using it would be inefficient. Instead, we use the tangent angle transformation method to take the ratio of the  $y$  and  $x$  coordinates and plot it against slow time to obtain a 2D image that incorporates all three variables with a unique shape for each character. For practical application, we do not want to restrict the user to drawing characters in a specific area rather than anywhere on the virtual plan. Therefore, we first cancel the effect of the shift in distance from the origin in the horizontal and vertical directions by subtracting the mean horizontal and vertical values from each horizontal

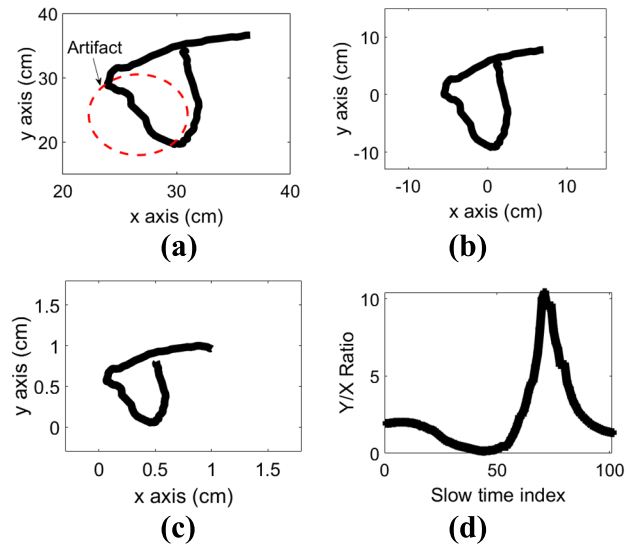


and vertical value. This can cause some values to become negative, so we add the absolute minimum value to both axes to shift the character shape to the positive quadrant. Then, we find the ratio of the vertical and horizontal axis values and plot it against slow time to get the transformed image. Because humans naturally cannot control the exact speed and duration to write a specific gesture, we cancel the effect of the writing speed by resizing the resultant image to a constant size. The steps of the algorithm are presented below.

**Algorithm 1** Image Construction From Positioning Data With the EKF

1. **Procedure**
2. **Input:**
3. Length of gesture:  $k$  slow time steps
4. Estimated position data  $(x_k, y_k)$
5. **Output:**
6. Image of  $y/x$  ratio plotted against slow time  $k$
7. **DC removal** w.r.t  $x$  and  $y$  axes
8. **for**  $i = 1: k$
9.  $x_i = x_i - \mu_x$ ;
10.  $y_i = y_i - \mu_y$ ;
11. **end for**
12.  $\mu_x$  and  $\mu_y$  are mean values of the  $x$  and  $y$  coordinates
13. **Shift** to positive coordinates
14. **for**  $i = 1: k$
15.  $x_i = x_i + \text{abs}(\min(x)) + 1$
16.  $y_i = y_i + \text{abs}(\min(y)) + 1$
17. **end for**
18. The value of 1 is added so that the next step does not result in infinity or zero.
19. **Normalize:** The values are normalized between zero and 1 as follows:
20. **for**  $i = 1: k$
21.  $x_i = x_i / \max(x)$
22.  $y_i = y_i / \max(y)$
23. **end for**
24. **Determine** the ratio of the horizontal and vertical axis values
25. **for**  $i = 1: k$
26.  $\tan_i = y_i / x_i$
27. **end for**
28. **Plot** the  $y/x$  ratio against the slow time index  $k$
29. **Resize** the image to  $100 \times 100$  pixels to cancel the effect of the writing speed. Without resizing, slow writing will result in a larger image size and vice versa.
30. **End procedure**

Fig. 7(a) shows the initial image obtained from the  $(x, y)$  coordinate data for the character ‘‘T.’’ Fig. 7(b) shows the image after the DC removal step is applied to nullify the distance shift effect. Fig. 7(c) shows the image after the normalization step, and Fig. 7(d) shows the image transformed to the tangent ratio vs slow time. After the image is constructed, we use a CNN classifier to extract the features from the images and train the network, as discussed in the next section.



**FIGURE 7.** Image patterns for the character ‘‘T’’: (a) original 2D  $(x_k, y_k)$  image, (b) after DC removal, (c) after normalization, and (d) after tangent ratio is plotted against slow time.

**D. GESTURE CLASSIFICATION WITH A CNN**

We used a CNN to classify the image patterns. CNNs are extensively used as a deep learning technique that mimics the human vision system [26]. A CNN consists of convolutional, pooling, and fully connected layers. In convolutional layers, the key features of the input image are extracted by a convolutional filter. These layers have several feature extraction filters, and each filter performs a convolution operation while sliding the input image to generate a feature map. The first convolutional layer extracts partial features such as the edge component of the input image, and later convolutional layers extract global features [27]. Pooling layers reduce the total data size by subsampling operations. They reduce the number of weights and biases to be optimized so that the CNN can be optimally trained without overfitting. Pooling methods include max pooling to select the maximum value, median pooling to select the median value, and mean pooling to select the mean value. In fully connected layers, the input image is classified through the output of the last convolutional layer and the deep neural network. In the flattening process, the output data of the convolutional layer are converted into one-dimensional data which are inputted to the deep neural network. The output of the deep neural network is applied to the softmax layer to calculate the probability that the input image is classified into each category.

Fig. 8 shows the CNN structure used in this study to extract the optimal features from the hand gesture image pattern. The CNN structure of the proposed method consists of five convolutional layers and four pooling layers. The number of convolutional layers and size of the convolutional filter were optimized through trial and error to achieve the desired accuracy. For example, the accuracy was highest with five convolutional layers and similar with five convolutional layers, so the number of convolutional layers was set to five.

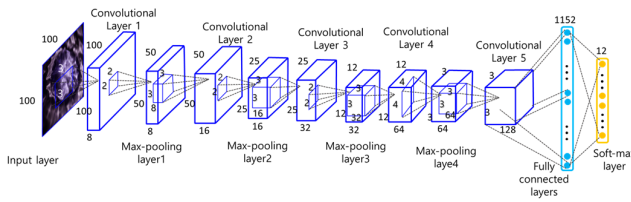


FIGURE 8. Structure of the convolutional neural network.

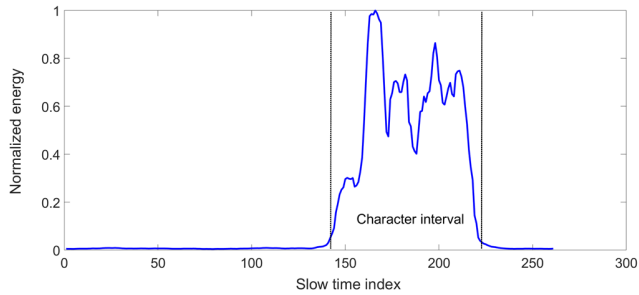


FIGURE 9. Character interval identification with the energy threshold algorithm.

The rest of the hyperparameters were determined in a similar manner. The rectified linear unit (ReLU)  $f(x) = \max(0, x)$  was used as an activation function because it performed better than the existing tanh or sigmoid function [28]. In addition, the max pooling technique was used because recent studies have demonstrated its excellent performance compared to other pooling techniques such as median pooling and mean pooling. The CNN was trained with the backpropagation algorithm, and parameters were updated through stochastic gradient descent with momentum [29]. The initial values of the weights were set to a normal distribution with a mean of 0 and standard deviation of 0.01, and the initial bias was set to 0.

**E. CHARACTER INTERVAL SEGMENTATION ACCORDING TO THE SIGNAL ENERGY**

Finding the start and end of a character is very important. Because of the narrow bandwidth of the antenna, the signal magnitude dropped abruptly when the hand was taken outside the writing plane. Thus, we separated characters according to the signal magnitude by taking our hand in and out of the writing plane after a character was finished and before starting the next one. Algorithm 2 presents the steps in detail. Because this method depends on the reflected signal energy, it is specific to radio sensors. It provides good performance and yet is a very simple technique for separating characters.

Fig. 9 shows some characters written in a continuous fashion. The interval for each character was identified with Algorithm 2 based on the energy reflected from the hand. Samples 146–228 showed a higher energy level, which indicates when the hand was moving in the writing plane, while the other samples showed low energy values. Hence, the gesture interval was accurately identified with the algorithm.

**Algorithm 2** Recognizing Character Intervals Inside a Continuous Stream According to the Signal Energy

1. **Procedure**
2. **Input:**
3. Raw data from four radar sensors:  $r_p(m, n)$   $p = 1 \dots 4$ , radar index,  $n = 1 \dots N$ , slow time  $m = 1 \dots M$ , fast time index
4. **Output:**
5. Markers along slow time showing the start and end of a character
6. **Remove clutter:**
7. Remove clutter from signal with the background subtraction filter presented in Section III.A to obtain the clutter-free signal  $s_p(m, n)$
8. **Train** the system and find the average energy value of the signal
9.  $E_{av\_training} = \sum_{p=1}^4 \sum_{n=1}^{N-1} \sum_{m=1}^M (s_p(m, n)^2) / (4M(N-1))$
10. **Compare** the current energy value of the latest sample with the threshold value obtained during the training session
11.  $E_{av\_current} = \sum_{p=1}^4 \sum_{m=1}^M (s_p(m, n)^2) / (4M)$
12. **If** ( $E_{av\_current} > E_{av\_training}$ )
13. Start or end of character
14. **End**
15. **End procedure**

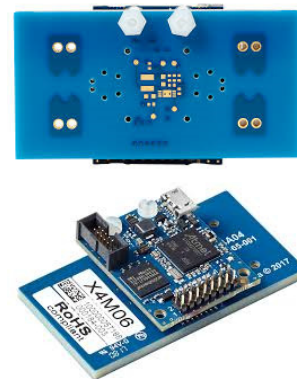


FIGURE 10. Novelda IR-UWB radar sensor (4x).

**IV. EXPERIMENTAL RESULTS AND DISCUSSION**

We performed experiments to verify the effectiveness of the proposed method at character classification.

**A. HARDWARE SETUP**

We placed four IR-UWB radar sensors at fixed locations as shown in Fig. 2 to make a virtual plane for in-air hand writing. As discussed previously, we used four radar sensors because the transceivers had a narrow beam width (around 65°) that made it difficult to cover the whole plane with only two or three radar sensors. Using four sensors improved the recognition accuracy. Fig. 10 shows the Xethru

TABLE 1. Radar parameters.

Parameters	Value
Output power	-12.6 dBm
Center frequency	8.748 GHz
Pulse repetition frequency	400 MHz
Bandwidth (-10 dB)	2.95 GHz
Range resolution	6.4 mm
Beam width	65°
Staggered PRF sequence length	220 cycles
No. of antenna arrays per radar chip	1 Tx and 1 Rx

X4 (Novelda, Norway) IR-UWB radar module used in this study. Table 1 gives the parameters of the radar sensors.

**B. CLUTTER REMOVAL**

Fig. 11 shows the clutter removal results for the slow and fast times of a gesture. The signal in Fig. 11(a) clearly contains some clutter information at samples 80–100, which made target tracking difficult. However, Fig. 11(b) shows that this high-amplitude signal was removed after clutter removal.

**C. LOCALIZATION AND IMAGE CONSTRUCTION RESULTS**

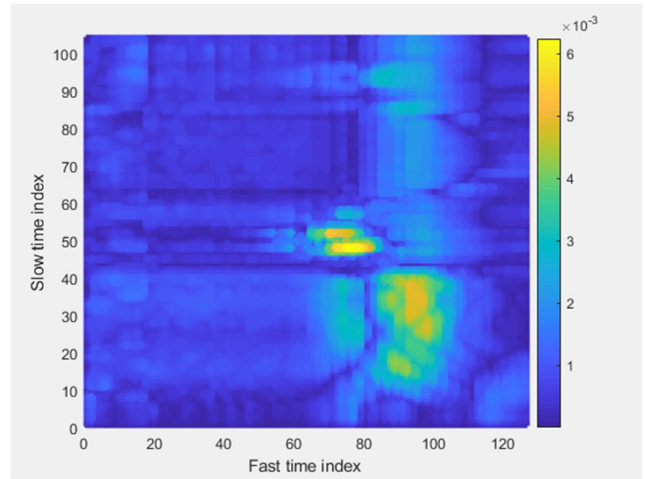
The results for some characters are presented here. We used the MATLAB software for image processing and classification with deep learning. Fig. 12 plots the images of some characters using both conventional positioning data and our proposed transformation method. Figs. 12(g) and 12(i) show that the conventional positioning data can lead to confusion between the digits “6” and “5.” However, Figs. 12(h) and (j) show that the transformed images for the corresponding digits are clearly different. Hence, the transformed images are unique even if the 2D localization data are affected by artifacts and show similar patterns.

**D. CLASSIFICATION RESULTS OF THE STATE-OF-THE-ART AND PROPOSED METHODS FOR CHARACTERS WITH ARTIFACTS**

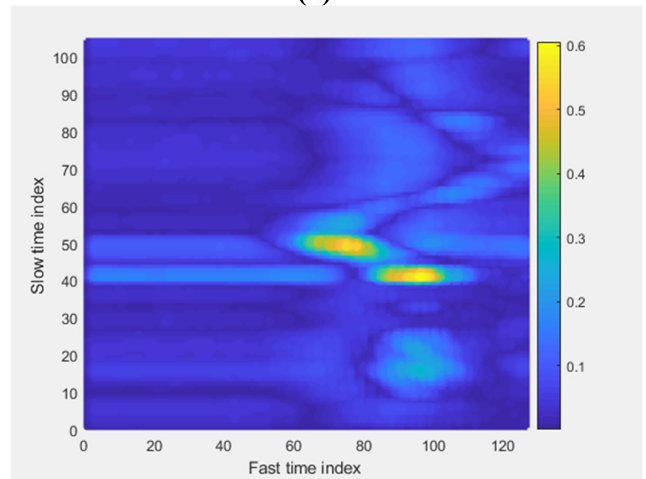
We compared the classification results of the state-of-the-art 2D trajectory-based method [23] and our proposed method for characters with artifacts by using a confusion matrix. For our experiments, we used three human males between 27 and 32 years old to perform gestures. For all cases, we used 100 samples for training, while 300 samples were used for testing. As per the leave-one-person-out cross-validation (LOPO-CV) scheme, one person did not participate in the training session and was only included in the test session to show the independence of the algorithm with regard to the hand shape and size of a person.

1) COMPARISON OF CLASSIFICATION RESULTS FOR CHARACTERS (DIGITS WITHOUT ARTIFACTS)

The accuracy results of the conventional and proposed methods were compared for the digits 0–9. First, the data were collected for training: 10 samples for each character.



(a)



(b)

FIGURE 11. Signal (a) before and (b) after clutter removal for the whole gesture duration.

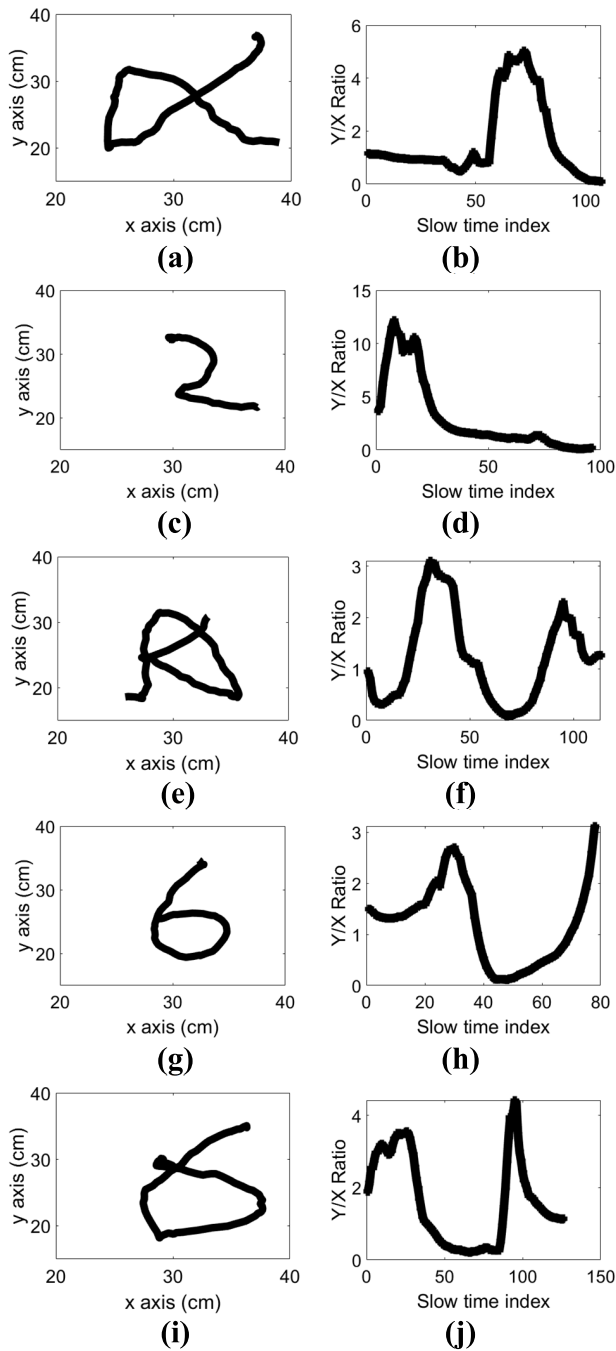
TABLE 2. Accuracy results of the state-of-the-art method for digits without artifacts.

Ac/Pr.	0	1	2	3	4	5	6	7	8	9
0	.96	0	0	0	0	0	0	0	0	.03
1	0	1	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	0
6	.03	0	0	0	0	0	.96	0	0	0
7	0	0	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	0	0	0	1

After training, we used 30 gestures to test each character. Tables 2 and 3 indicate that the accuracy results did not differ much for characters without artifacts. Thus, we next considered the accuracy for characters with artifacts.

2) COMPARISON OF CLASSIFICATION RESULTS FOR CHARACTERS WITH ARTIFACTS

We selected 10 alphanumeric characters that result in artifacts during in-air writing for comparison. The first 10 gesture



**FIGURE 12.** (a, c, e, g, i) Images based on 2D positioning data and (b, d, f, h, j) transformed images plotting the tangent ratio against slow time for the characters “X”, “2”, “A”, “6”, and “5”.

samples of each character were collected for training, and the next 30 gestures were used for testing. Tables 4 and 5 present the recognition accuracy results of the conventional and proposed methods. The accuracy of the conventional method decreased because some characters with artifacts produced (x, y) patterns similar to those of other characters. In contrast, the proposed method drastically improved the recognition accuracy by adding the time information to the 2D coordinates because this captured not only the shape of the character

**TABLE 3.** Accuracy results of the proposed method for digits without artifacts.

Ac/Pr.	0	1	2	3	4	5	6	7	8	9
0	1	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0
4	0	0	0	0	1	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	0
6	0	0	0	0	0	0	1	0	0	0
7	0	0	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	0	0	0	1

**TABLE 4.** Accuracy results of the state-of-the-art method for characters with/without artifacts.

Ac/Pr.	A	a	b	D	V	O	T	X	5	6
A	1	0	0	0	0	0	0	0	0	0
a	0	.93	0	0	0	.03	0	.03	0	0
b	0	0	.90	0	0	0	0	0	0	.1
D	0	0	0	.96	0	.03	0	0	0	0
V	0	0	0	0	1	0	0	0	0	0
O	0	0	0	.1	0	.9	0	0	0	0
T	0	0	0	0	0	0	1	0	0	0
X	0	.13	0	0	0	0	0	.86	0	0
5	0	0	.03	0	0	0	0	0	.83	.13
6	0	0	.13	0	0	0	0	0	.06	.8

**TABLE 5.** Accuracy results of the proposed method for characters with/without artifacts.

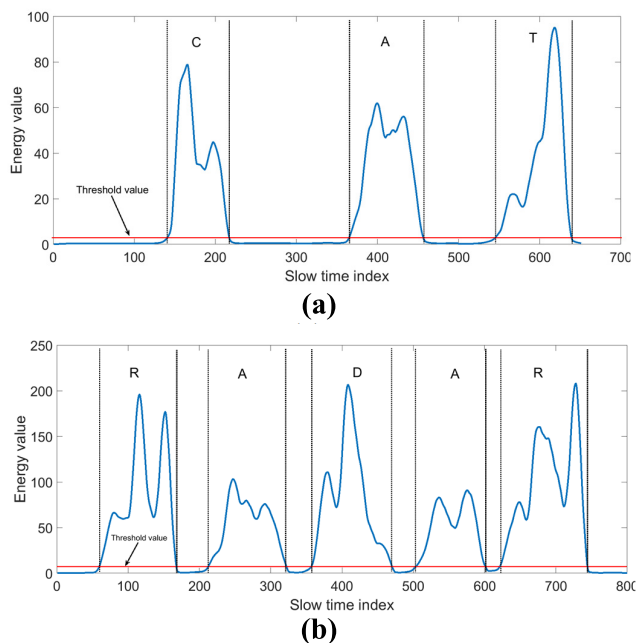
Ac/Pr.	A	a	b	D	V	O	T	X	5	6
A	1	0	0	0	0	0	0	0	0	0
a	0	1	0	0	0	0	0	0	0	0
b	0	0	1	0	0	0	0	0	0	0
D	0	0	0	1	0	0	0	0	0	0
V	0	0	0	0	1	0	0	0	0	0
O	0	0	0	0	0	1	0	0	0	0
T	0	0	0	0	0	0	1	0	0	0
X	0	0	0	0	0	0	0	1	0	0
5	0	0	0	0	0	0	0	0	1	0
6	0	0	0	0	0	.06	0	0	0	.93

written midair but also the writing style of each character. For example, the classification accuracy for “5” and “6” was much improved because, although these two characters have similar shapes, the writing styles are different. Similarly, the proposed method greatly improved the recognition accuracy of the characters “X” and “a.”

**E. ACCURACY RESULTS FOR CONTINUOUS CHARACTER WRITING**

In continuous writing, there is no constraint on the interval between characters. This means that the interval between two consecutive characters depends upon the user’s intention and comfort. We applied the energy threshold algorithm to certain words to demonstrate the energy levels when a character was being written and the interval between two characters. Fig. 13 shows the segmentation results for the words “CAT” and “RADAR.” In order to show some diversity, we intentionally made the intervals between characters of variable





**FIGURE 13. Segmentation results with the energy threshold algorithm when writing the words (a) “CAT” and (b) “RADAR.”**

length. For example, the intervals between characters are slightly longer for “CAT” than for “RADAR.” The energy level in slow time was much higher when the hand was in the radar plane than when it was outside it. The energy level was very low without the hand because we had already removed the static clutter signal through background subtraction. The large difference in energy levels with and without the hand in the radar plane showed that the energy threshold algorithm was very effective at segmenting the characters during continuous writing. The segmentation accuracy was 100% in this study. We tested a set of 10 words for each word length ranging from two to seven characters. Thus, a total of 60 words was used to evaluate the segmentation accuracy of the energy threshold algorithm for radar-based continuous writing.

## V. CONCLUSIONS

In this study, we classified characters using IR-UWB radar sensors and deep learning through a CNN. In our proposed method, 2D positioning data reflected from our hand in midair is collected by radar sensors and preprocessed with a localization algorithm before being transformed into tangent ratio data along a slow time axis. The resulting image is inputted to the CNN for character classification. An energy threshold algorithm was also developed for accurate character segmentation of continuous writing in midair. The main objective of our study was to overcome the problems caused by artifacts that occur in midair writing. We showed that our proposed method using transformed images improved the accuracy even for cases with artifacts. The accuracy was improved by 8.2% for 10 characters. In addition, the energy

threshold algorithm accurately segmented the characters of midair handwriting. In future research, we plan to extend our work to all characters, including special characters, so that a complete in-air keyboard can be developed.

## ACKNOWLEDGMENT

(Faheem Khan and Seong Kyu Leem are co-first authors.)

## REFERENCES

- [1] B. Bauer and H. Hienz, “Relevant features for video-based continuous sign language recognition,” in *Proc. 4th IEEE Int. Conf. Face Gesture Recognit.*, Grenoble, France, Mar. 2000, pp. 440–445.
- [2] J. Weissmann and R. Salomon, “Gesture recognition for virtual reality applications using data gloves and neural networks,” in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Washington, DC, USA, 1999, pp. 2043–2046.
- [3] Y. Zou, J. Xiao, J. Han, K. Wu, Y. Li, and L. M. Ni, “GRfid: A device-free RFID-based gesture recognition system,” *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 381–393, Feb. 2017.
- [4] C. Zhu and W. Sheng, “Wearable sensor-based hand gesture and daily activity recognition for robot-assisted living,” *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 41, no. 3, pp. 569–573, May 2011.
- [5] S. S. Rautaray and A. Agrawal, “Vision based hand gesture recognition for human computer interaction: A survey,” *Artif. Intell. Rev.*, vol. 43, no. 1, pp. 1–54, Jan. 2015.
- [6] F. Khan and S. Cho, “A detailed algorithm for vital sign monitoring of a stationary/non-stationary human through IR-UWB radar,” *Sensors*, vol. 17, no. 2, p. 290, 2017.
- [7] J. R. Fernandes and D. Wentzloff, “Recent advances in IR-UWB transceivers: An overview,” in *Proc. IEEE Int. Symp. Circuits Syst.*, Paris, France, May 2010, pp. 3284–3287.
- [8] J. W. Choi, S. S. Nam, and S. H. Cho, “Multi-human detection algorithm based on an impulse radio ultra-wideband radar system,” *IEEE Access*, vol. 4, pp. 10300–10309, 2016.
- [9] J. W. Choi, D. H. Yim, and S. H. Cho, “People counting based on an IR-UWB radar sensor,” *IEEE Sensors J.*, vol. 17, no. 17, pp. 5717–5727, Sep. 2017.
- [10] Y. Lee, J.-Y. Park, Y.-W. Choi, H.-K. Park, S.-H. Cho, S. H. Cho, and Y.-H. Lim, “A novel non-contact heart rate monitor using impulse-radio ultra-wideband (IR-UWB) radar technology,” *Sci. Rep.*, vol. 8, no. 1, 2018, Art. no. 13053.
- [11] J.-Y. Park, Y. Lee, Y.-W. Choi, R. Heo, H.-K. Park, S.-H. Cho, S. H. Cho, and Y.-H. Lim, “Preclinical evaluation of a noncontact simultaneous monitoring method for respiration and carotid pulsation using impulse-radio ultra-wideband radar,” *Sci. Rep.*, vol. 9, no. 1, pp. 1–12, Dec. 2019.
- [12] S. Leem, F. Khan, and S. Cho, “Vital sign monitoring and mobile phone usage detection using IR-UWB radar for intended use in car crash prevention,” *Sensors*, vol. 17, no. 6, p. 1240, 2017.
- [13] V. Nguyen, A. Q. Javaid, and M. A. Weitauer, “Harmonic path (HAPA) algorithm for non-contact vital signs monitoring with IR-UWB radar,” in *Proc. IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Rotterdam, The Netherlands, Oct. 2013, pp. 146–149.
- [14] A. Lazaro, D. Girbau, R. Villarino, and A. Ramos, “Vital signs monitoring using impulse based UWB signal,” in *Proc. 41st Eur. Microw. Conf.*, Manchester, U.K., 2011, pp. 135–138.
- [15] F. Khan, J. W. Choi, and S. H. Cho, “Vital sign monitoring of a non-stationary human through IR-UWB radar,” in *Proc. 4th IEEE Int. Conf. Neww. Infrastruct. Digit. Content*, Beijing, China, Sep. 2014, pp. 511–514.
- [16] J. W. Choi and S. H. Cho, “3D positioning algorithm based on multiple quasi-monostatic IR-UWB radar sensors,” in *Proc. IEEE Radar Conf. (RadarConf)*, Seattle, WA, USA, May 2017, pp. 1531–1535.
- [17] S. Ahmed, F. Khan, A. Ghaffar, F. Hussain, and S. Cho, “Finger-Counting-Based gesture recognition within cars using impulse radar with convolutional neural network,” *Sensors*, vol. 19, no. 6, p. 1429, 2019.
- [18] F. Khan and S. H. Cho, “Hand based gesture recognition inside a car through IR-UWB radar,” in *Proc. ICEIC*, Phuket, Thailand, 2017, pp. 154–157.
- [19] F. Khan, S. K. Leem, and S. H. Cho, “Human–computer interaction using radio sensor for people with severe disability,” *Sens. Actuators A, Phys.*, vol. 282, pp. 39–54, Oct. 2018.

- [20] A. Ghaffar, F. Khan, and S. H. Cho, "Hand pointing gestures based digital menu board implementation using IR-UWB transceivers," *IEEE Access*, vol. 7, pp. 58148–58157, 2019.
- [21] B. Dekker, S. Jacobs, A. S. Kossen, M. C. Kruijthof, A. G. Huizing, and M. Geurts, "Gesture recognition with a low power FMCW radar and a deep convolutional neural network," in *Proc. Eur. Radar Conf. (EURAD)*, Nuremberg, Germany, Oct. 2017, pp. 163–166.
- [22] Y. Kim and B. Toomajian, "Hand gesture recognition using micro-Doppler signatures with convolutional neural network," *IEEE Access*, vol. 4, pp. 7125–7130, 2016.
- [23] S. K. Leem, F. Khan, and S. H. Cho, "Detecting mid-air gestures for digit writing with radio sensors and a CNN," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp. 1066–1081, Apr. 2020.
- [24] M. Piccardi, "Background subtraction techniques: A review," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, The Hague, The Netherlands, Oct. 2004, pp. 3099–3104.
- [25] R. Khan, S. U. Khan, S. Khan, and M. U. A. Khan, "Localization performance evaluation of extended Kalman filter in wireless sensors network," *Procedia Comput. Sci.*, vol. 32, pp. 117–124, Jan. 2014.
- [26] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural Comput.*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017.
- [27] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. ECCV*, Munich, Germany, 2014, pp. 818–833.
- [28] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. PMLR*, Fort Lauderdale, FL, USA, 2011, pp. 315–323.
- [29] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.



**FAHEEM KHAN** was born in Bannu, Pakistan, in August 1988. He received the Ph.D. degree in electronics and computer engineering from Hanyang University, South Korea, in 2018. He is currently working as a Postdoctoral Fellow with the Radar Sensors Laboratory, Hanyang University, where he is also a Postdoctoral Researcher. He has published several conference papers and articles in reputed journals. His research interests include radar signal processing, computer vision, object tracking and localization, gestures recognition, and vital signs monitoring using IR-UWB radar sensors.



**SEONG KYU LEEM** (Graduate Student Member, IEEE) was born in Sejong-si, South Korea, in November 1980. He received the B.S. degree in electrical engineering from Korea University, Seoul, South Korea, in 2003, and the M.S. degree in electrical engineering and computer science from Seoul National University, Seoul, in 2005. He is currently pursuing the Ph.D. degree in electronics and computer engineering with Hanyang University, Seoul. His current research interests include RF impairment compensation for radio communication systems and radar signal processing.



**SUNG HO CHO** (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Utah, Salt Lake City, USA, in 1989. From 1989 to 1992, he was a Senior Member of Technical Staff with the Electronics and Telecommunications Research Institute, Daejeon, South Korea. He joined the Department of Electronic Engineering, Hanyang University, Seoul, South Korea, in 1992, where he is currently a Professor. His research interests include applied signal processing, machine learning for signal processing, context aware computing, radar sensors, and smart space and wireless networks. Since 2008, he has been a recipient of the High-Level Foreign Experts Fellowship at the Beijing University of Posts and Telecommunications, Beijing, China, led by the Ministry of Education of China.

...