



Article

Radar Application: Stacking Multiple Classifiers for Human Walking Detection Using Micro-Doppler Signals

Jihoon Kwon ^{1,2}  and Nojun Kwak ^{1,*} ¹ Graduate School of Convergence Science and Technology, Seoul National University, Seoul 08826, Korea² Hanwha Systems Yongin R&D Center, Yongin-si, Gyeonggi-do 17121, Korea

* Correspondence: nojunk@snu.ac.kr; Tel.: +82-31-888-9166

Received: 16 July 2019; Accepted: 23 August 2019; Published: 28 August 2019



Abstract: We propose a stacking method for ensemble learning to distinguish micro-Doppler signals generated by human walking from background noises using radar sensors. We collected micro-Doppler signals caused by four types of background noise (line of sight (LoS), fan, snow and rain) and additionally considered micro-Doppler signals caused by human walking combined with these four types of background noise. We firstly verified the effectiveness of a fully connected deep neural network (DNN) to classify 8 types of signals. The average accuracy was 88.79% for the test set. Then, we propose a stacking method to combine two base classifiers of different structures. The average accuracy of the stacking method on the test set was 91.43%. Lastly, we designed a modified stacking method to reuse feature information stored at the previous stage and the average test accuracy increased to 95.62%. This result shows that the proposed stacking methods can be an effective approach to improve classifier's accuracy in recognizing human walking using micro-Doppler signals with background noise.

Keywords: human detection; radar sensor; radar machine learning; stacking learning; micro-doppler signal

1. Introduction

Radar sensors are relatively robust to weather conditions and have the advantage of being able to detect large areas. In particular, it can be applied as a helpful sensor for detecting humans in outdoor conditions for security purposes. For detection, Doppler signals of radar are essential information to filter moving targets under environments with complicated background noise. For air surveillance radar, since the speed of a target is very fast, it may be not challenging to classify target signals from background noises. However, human walking motions induce complex frequency modulation that includes the Doppler shift by a torso and the additional Doppler shifts of the side-band in the returned signal; this phenomenon is known as the micro-Doppler effect [1]. In particular, since the speed of human walking is much slower than that of airplanes or vehicles, classifying micro-Doppler signals by humans walking from background noise signals can be a severe problem. Since the bandwidth of micro-Doppler signals of human walking motion can overlap the bandwidth of micro-Doppler signals of background noise, it can be challenging to filter two signals individually. Figure 1 describes the difficulty of detecting human walking in an outdoor environment.

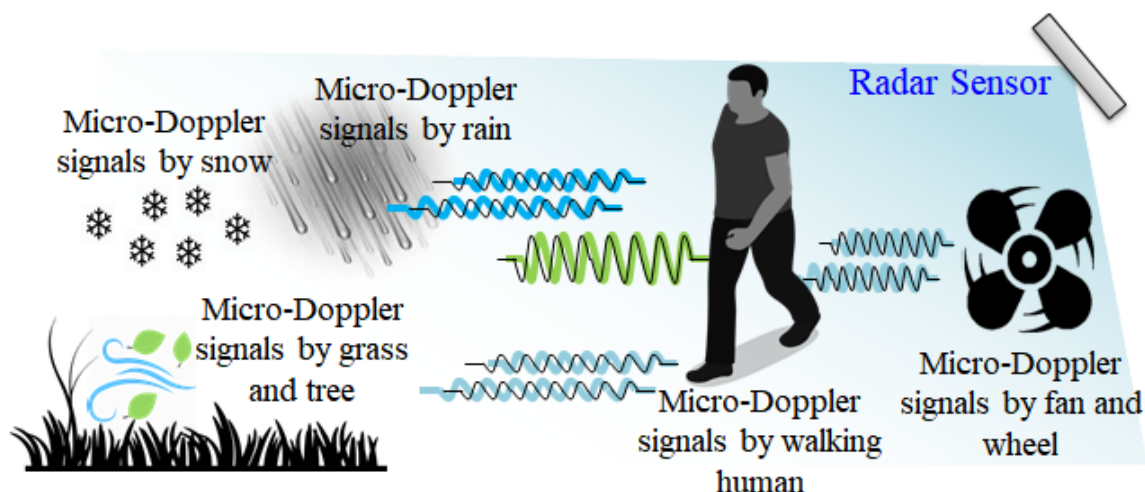


Figure 1. The difficulties of recognizing human walking by micro-Doppler signals. The background noise should also be considered when detecting human walking on the ground. Also, recognizing background noise can provide useful information.

Processing techniques that analyze or recognize patterns of received signals can be useful approaches to overcome these limitations. Much research has investigated detection problems in Doppler radar signals using micro-Doppler signatures or pattern recognition techniques. According to the reported studies, many researchers have described or proposed mathematical equations or approaches for extracting a meaningful feature vector that can affect the classifier's estimation accuracy [2–7]. Similarly, other studies tried to find meaningful features through statistical analysis [5–8]. These approaches need a relatively long pre-processing time (window time) to extract a feature vector from the micro-Doppler spectrogram. For example, the authors of Reference [2] reported that six features are extracted by processing data over a three-second period. Another work [3] analyzed the classification accuracy according to the dwell time and reported that three or four seconds were required to get 90% accuracy. Since this window time directly affects the response and update speed of a radar sensor, a radar sensor's response speed is lowered as we increase this window time.

Also, References [9,10] showed that a model having parameters or mathematical equations could be used to formalize micro-Doppler signals of human walking motion and the virtual features of a human walking motion can be generated. However, making a model that can cover all human activities and background noise may be challengeable. Also, statistical analysis or modeling of highly variable micro-Doppler signals for extracting an essential feature vector may be quite challenging approaches. Instead of these previous approaches, recent deep learning gradually reconstructs and generates features through a process of passing several hidden layers in deep networks. If we design deep learning well, we can easily extract a feature vector without complex calculations for pre-processing or simulation models.

Recent studies related to the convolutional neural network, which is a type of deep learning methods, have been researched for recognizing micro spectrogram signals and they showed improved classification accuracy by deep learning technology [11–13]. However, since the long window time is still required to obtain micro-Doppler spectrogram images, it may negatively affect the sensor's response speed. For example, there are limitations to the application of these approaches in short-range sensors that need to recognize targets quickly. On the other hand, in order to reduce the window time, References [14,15] proposed methods to directly extract features from raw data without any pre-processing and using them as the input to the designed classifier and they showed the possibility of classifying micro-Doppler signals without generating image data. However, since the amount of information in the input data for extracting the feature is also reduced according to the decreased

window time, the classification accuracy may degrade. So, it should be seriously considered how to improve the classification accuracy of the corresponding classifier.

To tackle this problem, in this paper, we propose a stacking method as an ensemble method of deep networks to distinguish micro-Doppler signals caused by humans walking from the background noise using radar sensors. The stacking method can be a practical approach to improve classifier’s accuracy without increasing the window time. About the latest research, Reference [16] reported that a stacking deep neural network (DNN) method improves performance for acoustic signal processing. In the field of image recognition, Reference [17] reported that stacking could be a useful approach to improve performance. Also, stacked auto-encoders to detect a human fall using radar have been researched [18].

About the need to include background noise signals in this research, since the background noise has the potential to distort the micro-Doppler effects generated by human walking, we should consider the effect of background noise in a design classification algorithm for outdoor applications. Also, if the classification algorithm can classify background status, it is possible to get helpful information about environmental conditions.

This paper is organized as follows. Section 2 describes the micro-Doppler signals collected in an outdoor environment and Section 3 describes the design of classifiers. Section 4 shows the experimental results and Section 5 concludes this paper.

2. Micro-Doppler Signals

2.1. Backgrounds

Moving targets generate Doppler frequency in the radar return signal due to velocity. Since most targets are not rigid bodies, different parts of a target may have additional vibrations and rotations. For example, when a person walks, his arms naturally swing. These movements generate additional Doppler shifts, called micro-Doppler effects. Much research regards these to be useful information for identifying the target.

The returned radar signal by a moving rigid body is modeled as shown in Equations (1) and (2), where f_0 is the carrier frequency, A is the intensity of the receiving signal, Δv is the moving speed of the target, c is the light speed, $\Delta f = \frac{\Delta v}{c} f_0$ is the Doppler shift and θ is the phase.

$$x_r(t) = A \cdot \exp(j(2\pi(1 + \frac{\Delta v}{c})f_0t + \theta)) \tag{1}$$

$$x_r(t) = A \cdot \exp(j(2\pi(f_0 + \Delta f)t + \theta)) \tag{2}$$

The returned radar signal having the micro-doppler effect can be modeled as a form of a combination of M sine-wave signals as shown in Equation (3).

$$x_r(t) = \sum_{i=1}^M A_i \cdot \exp(j(2\pi(f_0 + \Delta f_i)t + \theta_i)) \tag{3}$$

After the application of a matched filter and a low pass filter, we can get the base-band signal as shown in Equation (4).

$$x_b(t) = \sum_{i=1}^M A_i \cdot \exp(j(2\pi\Delta f_i t + \theta_i)) \tag{4}$$

Equation (5) shows the discrete signal sampled by the sampling rate T_s and Equation (6) shows the signal form converted by DFT (Discrete Fourier Transform), where N is the number of samples.

$$x_s[n] = x_b(t) |_{T_s} \tag{5}$$

$$X[k] = \sum_{n=0}^{N-1} x_s[n] \exp(-j2\pi kn/N) \tag{6}$$

The DFT signal $X[k]$ is calculated from N samples of x_s , which contains information on the variation of micro-Doppler signals over time. Therefore, even if a spectrogram image used by the previous studies is not applied, we can extract the feature information related to micro-Doppler effects from $X[k]$.

To reflect variability of $X[k]$ over time, we devise a method of utilizing previously generated feature information. Equation (7) shows the feature generated by a base classifier. And, we reuse this feature information as the input data for a blender to combine base classifiers in the proposed stacking method. Equation (8) shows the feature data for a blender. The structure of $Base_Classifier(\cdot)$ is shown in Figure 2 and the optimal weights of $Base_Classifier(\cdot)$ are calculated after training.

$$Feature(t) = Base_Classifier(X[k]) \tag{7}$$

$$Feature_{blender}(t) = \sum_{\tau=0}^j Feature(t - \tau) \tag{8}$$

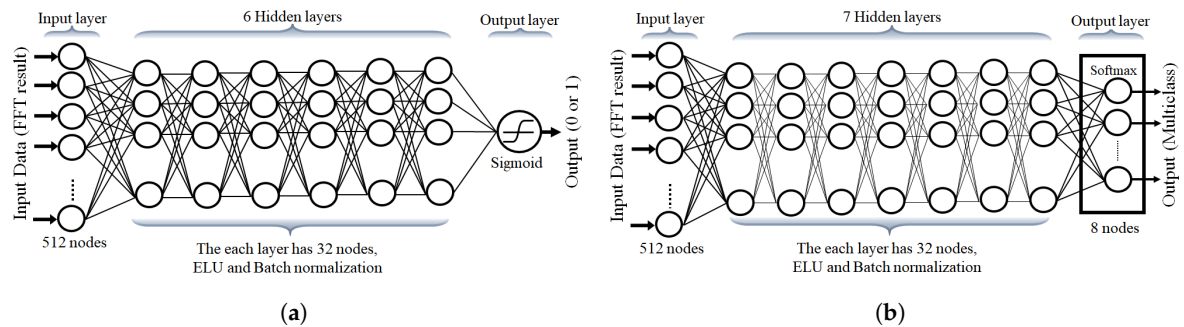


Figure 2. The binary classifier and multiclass classifier. (a) Binary classifier; (b) Multiclass classifier.

2.2. Radar Sensor Board and Specifications

We designed the Doppler radar sensor board. Figure 3 shows the block diagram of the low-cost & short-range 24 GHz Doppler Radar and Figure 4 shows the radar sensor board implemented in this research. Radar sensor board has two main part of RF-Antenna circuits and processing circuits. RF-Antenna circuits include the micro-strip patch antenna, CW signal generator, Tx amplifier (Drive amp.), low noise amplifier and homodyne I/Q mixer (direct conversion) and two-stage cascaded baseband amplifier. The processing circuits have the micro-controller having AD converter and interface for the external data transmission and power circuits. Table 1 shows the specifications of 24 GHz Doppler radar.

Table 1. The specifications of 24 GHz Doppler radar.

Items	Specifications	Ref.
Operating frequency	24.125 GHz	
RF output	16 dBm	TX port
Antenna Gain	6 dBi	with feed loss
Detection range	20 m (typ.)	Human walking (adult)
Detection angle	-40° ~ +40°	Azimuth
Micro-controller	Freescle Kinetis k60	Clock: 100 MHz
Processing time	1 s	
Sampling rate	1024 samples/s	
Size	60 mm × 60 mm × 15 mm	

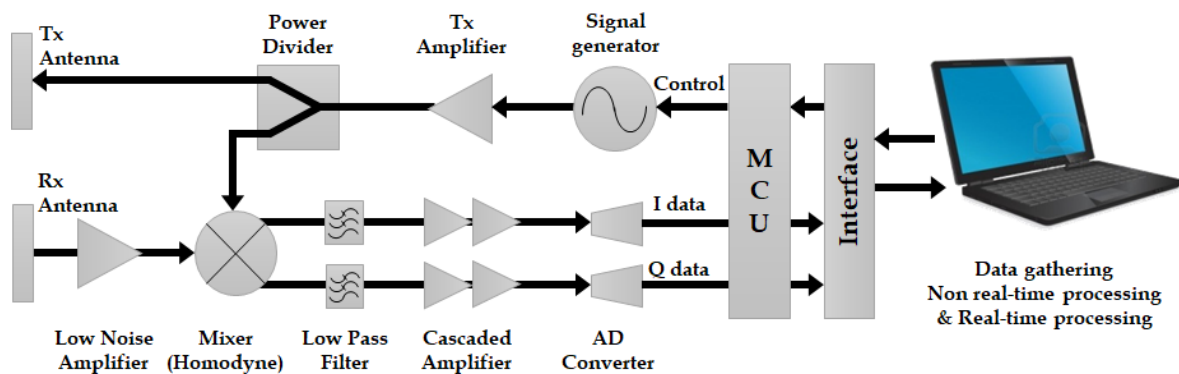


Figure 3. Block diagram of the low-cost & short-range 24 GHz Doppler Radar.

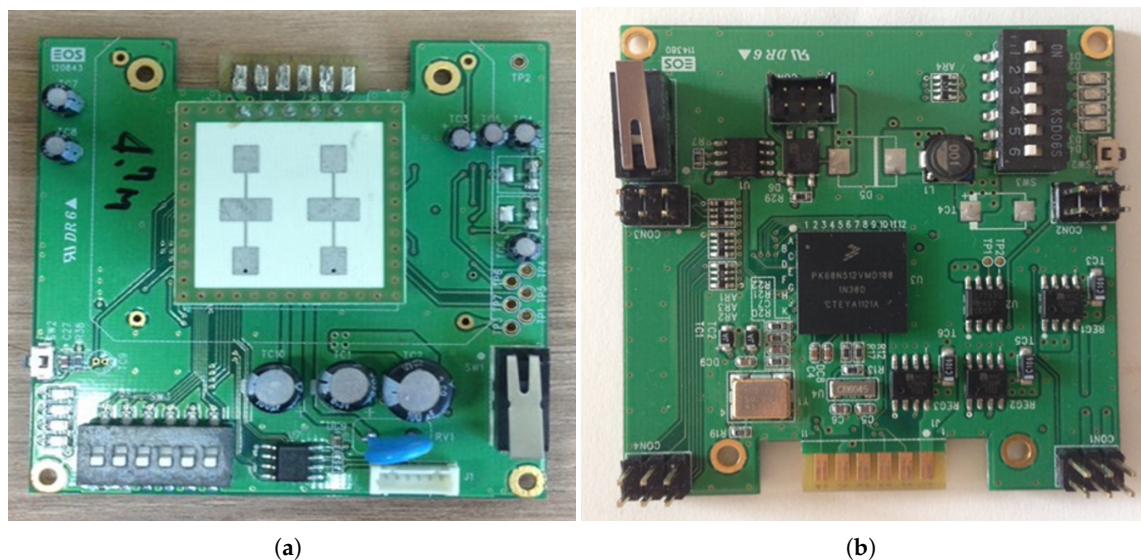


Figure 4. Twenty-four GHz Doppler radar sensor board implemented in this research. (a) Front; (b) Back.

2.3. Data Acquisition

Figure 5 shows the experimental setup used to gather signals. The test site is an open space with no obstacles. We considered micro-Doppler signals caused by four types of background noise (Line of sight (LoS), fan, snow and rain). These noise signals can be understood as common background noises that can frequently be received in real outdoor environments. We additionally considered human walking micro-Doppler signals combined with these four types of background noise; that is, we gathered the eight types of signals in total whose examples are shown in Figure 6. The Doppler radar was installed at a selected test site and raw data were gathered for several days. In particular, to see rain and snow effects, we collected a rain and snow background noise dataset.

To measure signals generated by human walking motions, the experimenter only moved within the radar detection area. We set the detection area through the pre-test and marked the boundary on the ground. Three adults (height: 173 cm, 177 cm, 182 cm) participated as the experimenter. The experimenter selected and moved the direction that the experimenter wanted within the detection area. We consider only human walking motions of regular walking and fast walking as general cases. These walking movements are the movements that we use in our daily life. Unusual human movements, such as crawling and irregular movements without direction and purpose, were excluded. To collect Doppler signals caused by background noises, we used a limited experimental environment to avoid mixing different background noises. The fan signals were collected using two types of ordinary electric fans for home use. The fan signal was measured at a distance of 5 m from the radar.

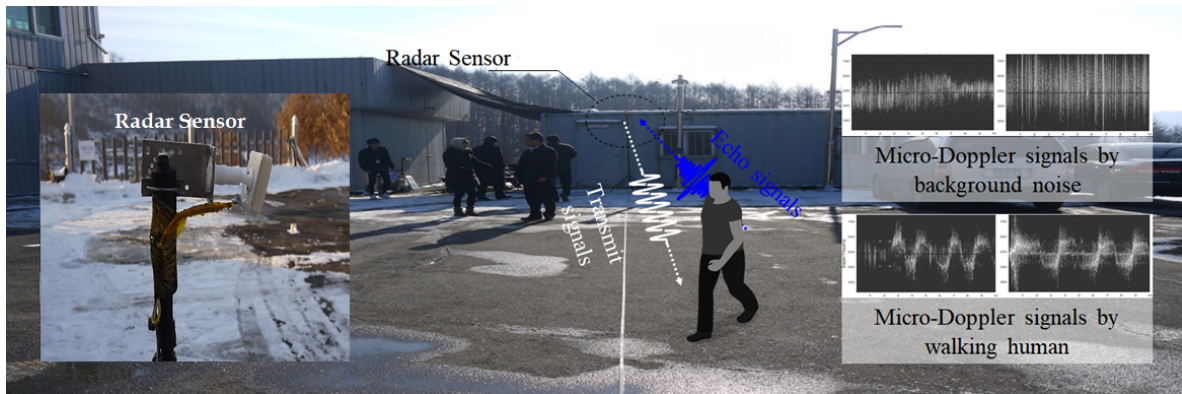


Figure 5. The experimental setup: the radar was installed outdoors to collect various background noise.

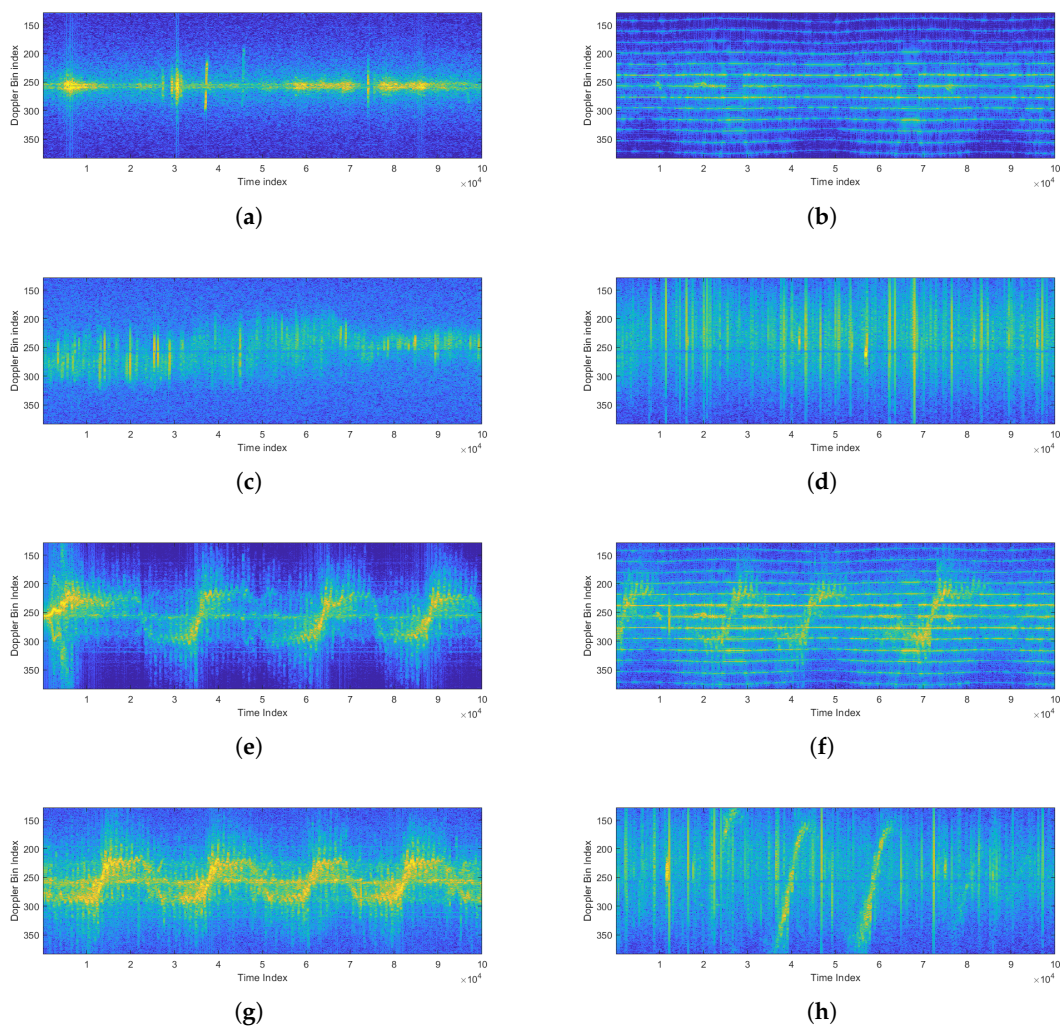


Figure 6. The micro-Doppler spectrograms of eight types. These show the spectrogram images during about 90 s. From these spectrograms, we can understand the frequency characteristics of each class. The signal by an electric fan has harmonic frequencies and the signals by snow and rain are relatively broadband frequency characteristics. In the case of the human walking signal, it can be aware that the signal intensity returned by torso is stronger than other parts. Therefore, it shows that it is possible to classify classes using the frequency characteristics of a signal instead of spectrogram images. (a) Outdoor environment (LoS); (b) Fan; (c) Snow; (d) Rain; (e) Human walking; (f) Human walking with fan; (g) Human walking with snow; (h) Human walking with rain.

The experiments under clear weather conditions were done for about three days and micro-Doppler signals for rain and snow were collected for two days each on the day when the weather conditions were met. The amount of rainfall and snowfall was according to the weather forecast and we measured rain and snow signal under moderate rainfall and snowfall conditions. We carefully notify that this experiment does not cover all weather conditions.

2.4. Pre-Processing for Raw Data

The processing methods of previous research require mathematical equations for extracting feature vectors from I/Q signal shown in Figure 3. In these methods, after extracting a feature vector by feature extraction algorithms using mathematical equations, the classification algorithm (e.g., SVM or DNN) processes this generated feature vector. However, the optimization of this approach could be quite complicated because of the difficulty of finding an essential feature vector and it requires comparatively long pre-processing time due to pre-processing for this feature extraction. Meanwhile, this paper does not use complex mathematical equations but also aims to extract features and classify targets in one second of processing time. We do not use any pre-processing methods for feature extraction and we use only Fourier transform in pre-processing for classification. We use signals of frequency domain converted by Fourier transform as the input data of DNN. The trained DNN automatically extracts and generates the features and it predicts the optimal results.

3. Classifier Design

We firstly design and test a binary classifier and a multiclass classifier using the most widely used support vector machine (SVM). Since SVM has been widely applied as a pattern recognition algorithm for detecting human motion by radar [2,5,19,20], we use SVM as a reference algorithm for comparison to our algorithm proposed in this study.

Before designing the stacking multiple classifiers, we design and verify the basic DNN (multiclass classifier) for classifying 8 types of signal. Then, we propose a stacking method to increase the average accuracy. To design the stacking method, we combine the multiclass classifier and the binary classifier. The binary classifier is designed to classify background noise signals (a, b, c and d) and human walking signals (e, f, g and h) in Figure 6. The stacking method can be used to complement each classifier. Lastly, the modified stacking method is designed to reflect the variability of $X[k]$ over time.

3.1. Support Vector Machine

SVM is one of the most popular machine learning techniques. SVM is known as a soft margin classification and it classifies class by finding the optimal hyper-plane to distinguish two data. The optimal hyper-plane means the hyperplane with the maximum margin between two data. An important consideration in using SVM is the use of kernel trick. In this paper, we apply the Gaussian RBF(Radial Basis Function), which is one of the most widely used kernel trick. The input of SVM directly uses $X[k]$ without any pre-processing.

Since SVM is a binary classifier, it is necessary to use several SVMs to design a multiclass classifier. Typically there are the one-versus-all and the one-versus-one method. If the number of the given classes is M , the one-versus-all approach trains eight binary classifiers that classify between one specific type and others. Moreover, the final result selects the highest score among each classifier's decision score. In the case of the one-versus-one method, since we should consider all combination cases of $M \times (M - 1)/2$ classifiers, the one-versus-one approach is more complicated. Nevertheless, since it can give better performance, we apply the one-versus-one approach in this study.

3.2. Deep Neural Network

We designed the binary classifier and the multiclass classifier using deep-neural-network (DNN). Figure 2a shows the architecture of the binary classifier and Figure 2b shows the architecture of the multiclass classifier. The DNN based binary classifier has one input layer, six hidden layers and one

output layer. The input layer has 512 nodes which corresponds to N , the hidden layer has 32 nodes in each layer and the output layer has one node. The Exponential Linear Unit (ELU) is used as the activation function, which has been reported to have an excellent performance [21]. The activation function for the output layer used the sigmoid function. Likewise, The DNN based multiclass classifier has one input layer, seven hidden layers and one output layer. The input layer has 512 nodes which corresponds to N , the hidden layer has 32 nodes in each layer and the output layer has eight nodes. The activation function for the output layer used the SoftMax. We use Batch Normalization (BN) for performance improvement, which helps train the optimal scale (mean and variance) for each layer’s input data. Generally, it is known that BN in DNN can produce a large enhancement in the result [22].

3.3. Stacking Classifier

We can expect a better result by combining the prediction results obtained from several predictors called ensembles; this learning method is called ensemble learning [19]. The idea of ensemble learning is that, even though each predictor is a weak classifier, a combination of many ensembles may become a strong predictor. Bagging, boosting and stacking are known as prevalent methods of ensemble learning and we adopt the stacking method in this paper.

Unlike the simple voting method, the stacking is a way to train a new predictor model on top of the last layer that aggregates predictions. The new predictor on the last layer is called the blender or meta-learner. A stacking method is a useful tool for combining classifiers with different structures [23,24]. It enables us to design a new classifier that combines the already-trained binary classifier (Figure 2a) and the multi-class classifier (Figure 2b). The binary classifier determines whether there is a human walking in the detection area and the multi-class classifier is designed to classify eight cases individually. The blender is optimized after combining these two classifiers to improve classification accuracy performance. Figure 7 shows the proposed structure of the stacking method.

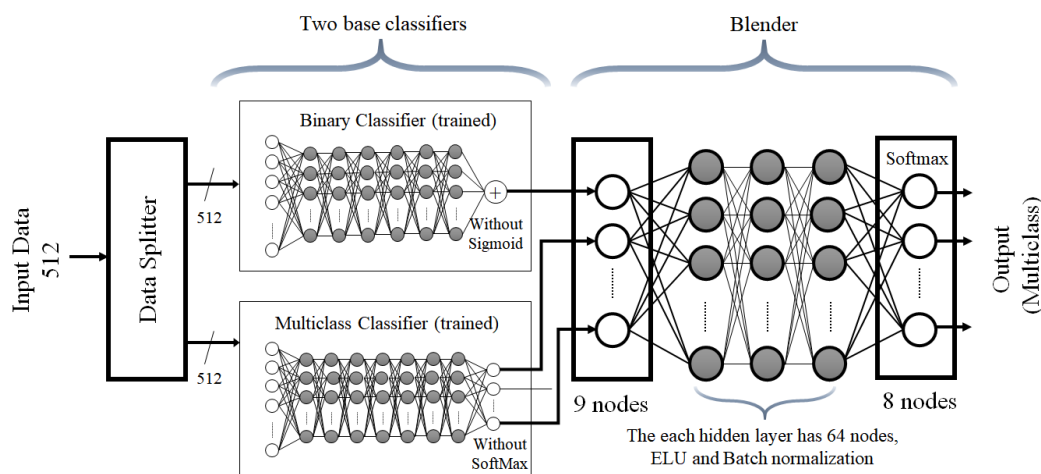


Figure 7. The designed stacking method structure: The blender combines the binary classifier with the multiclass classifier that has already been trained.

3.4. Modified Classifier

We propose the stacking multiple classifiers for the fast processing within 1 s and improving classification accuracy. Figure 8 shows the structure of the modified stacking method. The modified stacking method stores and reuses the multiclass classifier’s outputs to reflect continuous variability of micro-Doppler effect over time. The multiclass classifier produces $feature(t)$ and the blender combines three consecutive features as $feature_{blender}(t) = \{feature(t), feature(t - p), feature(t - 2p)\}$, where p is the time shift (0.25 s, 50% of the window size) and the total processing time is 1 s. The sampling rate is 1024 samples/s, the window size is 512 samples and the sliding interval is 256 samples (50%). The combined $feature_{blender}$ is used as the input feature for the blender classifier.

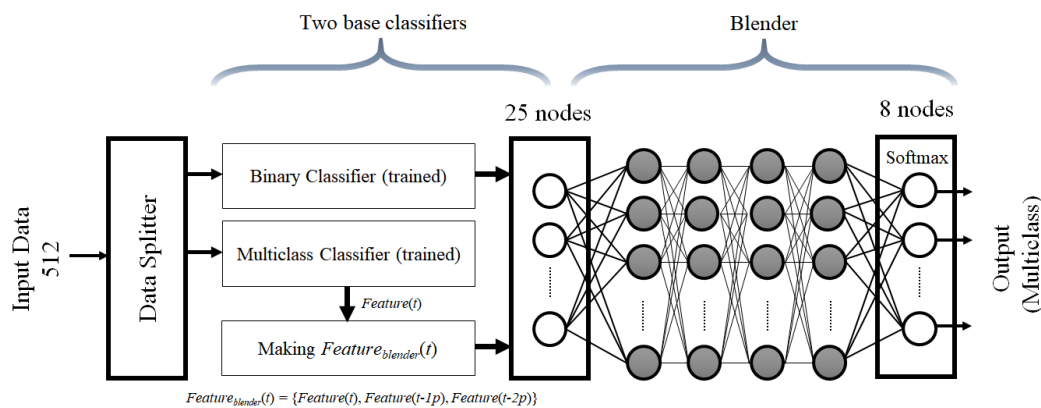


Figure 8. The modified stacking method structure: The outputs of the multiclass classifier are stored for a specific time and these outputs are used for input data to the blend.

Increasing the window size can improve accuracy performance but these lead to increasing the processing time. If only a sliding interval is reduced, classification performance can be degraded due to the decreasing of the amount of input information. Thus, the window time, sliding interval and the number of features should be taken into consideration together for designing a classifier. Increasing the number of features did not make a significant performance difference, because the change of micro-Doppler signals is not much within a short time of 1 s. On the other hand, increasing the sample size of the Fourier transform requires more samples in the time domain and this comes with an increase in window time and processing time. More importantly, since the number of nodes in the input and hidden layers of DNN should be increased, the training and processing computation of the DNN get exponentially complicated. We selected 512 samples, 50% sliding interval and three $\{feature(t), feature(t - p), feature(t - 2p)\}$ within 1 s through repeated experiments.

4. Experiment

We excluded severe outliers in the given dataset and did not apply any additional pre-processing related to feature selection or noise filtering. We collected (a) 40,000 outdoor environment data, (b) 40,000 fan data, (c) 40,000 snow data, (d) 40,000 rain data, (e) 40,000 human walking data, (f) 10,000 human walking data with fan, (g) 40,000 human walking data with snow and (h) 40,000 human walking data with rain. Half of the total collected data were used to train two base classifiers and the remaining half of the data was used to optimize the blender.

We used cross-validation for verification. 75% of the total samples were used for the training set, 12.5% were used for the validation set and the remaining 12.5% were used for the test set. After the training set optimizes the given algorithm model, we can analyze and verify the learning optimization and the generalized performance of the trained classifier through validation set and test set; where training accuracy is the classification accuracy we get when we apply the model on training set, while validation and test accuracy is the classification accuracy for validation data and test set.

The labels of data sets used to train and test the classifiers should be associated or the same. For example, the output labeling of the binary classifier is "0" or "1." "0" here means background noise signals without human walking. So the labeling of (a), (b), (c) and (d) by the multiclass classifier should be connected with "0" of the binary classifier. If "0" is associated with (e), (f), (g) or (h) in the given data set, it produces an incorrect result; likewise, the labels of the data set for learning the blender should be the same as those of the multiclass classifier. Also, in order to train and test the modified Stacking method, the components of $feature_{blender}(t)$ should be generated from the same class.

4.1. Support Vector Machine

We designed the binary classifier and the multiclass classifier using SVM. Figure 9 shows the ROC curve of the binary classifier and the confusion matrix of the multiclass classifier using SVM. Table 2 shows the accuracy performance of SVM based classifiers. Despite being highly optimized for the training set, the accuracy results on the verification set and test set have not improved. This result means that the algorithm is over-fitted for the training set and the generalization performance is poor.

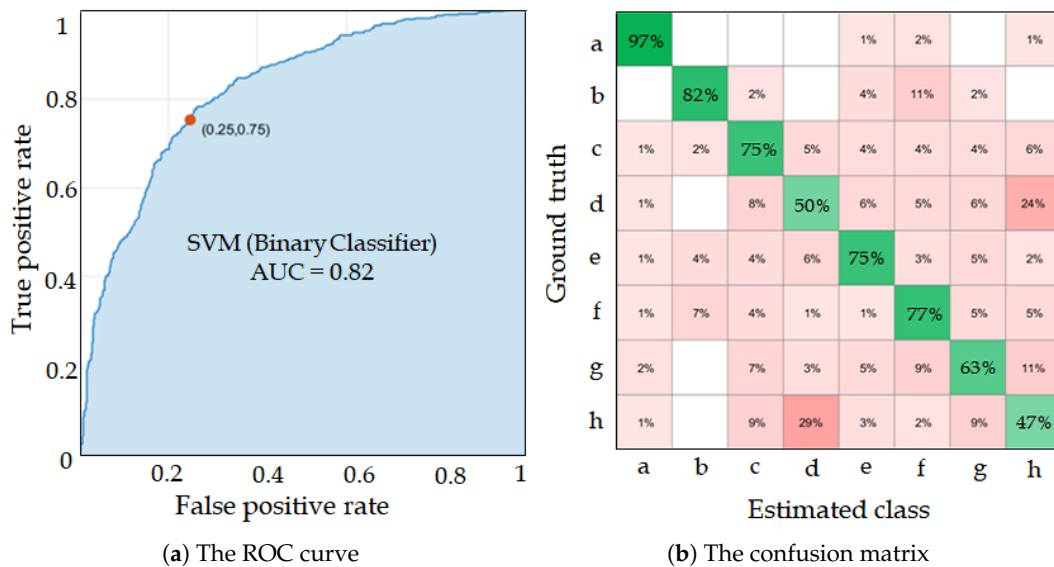


Figure 9. Support vector machine (SVM) test performances: (a) the ROC curve of the binary classifier, (b) The confusion matrix of the multiclass classifier.

Table 2. The accuracy performances of SVM based classifiers.

Network Structure	Training Accuracy	Validation Accuracy	Test Accuracy
SVM (Binary)	99.9%	75.8%	75.2%
SVM (Multiclass)	99.9%	71.8%	71.0%

4.2. Deep Neural Network

We designed and tested the DNN based binary classifier (Figure 2a) and the multiclass classifier (Figure 2b). Figure 10 shows the accuracy-loss graph of the DNN based binary classifier and the multiclass classifier. In the case of the binary classifier, the training accuracy was 93.42%, the validation accuracy was 89.71% and the test accuracy was 89.65%. And, in the case of the multiclass classifier, The training accuracy was 96.88%, the validation accuracy was 88.71% and the test accuracy was 88.79%. Table 3 shows the confusion matrix of the DNN based multiclass classifier. We reduced fluctuation in the validation loss graph by lowering the learning rate of Adam optimizer and adjusting the batch size but there was a limit to reducing the fluctuation. DNN showed a definite performance improvement over SVM.

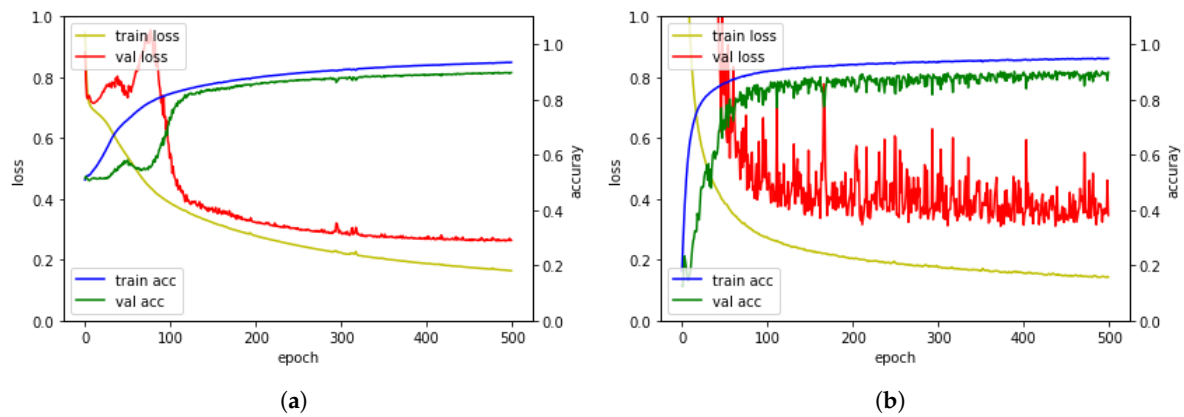


Figure 10. Deep Neural Network (DNN): the accuracy and loss graph of training dataset and validation dataset. (a) Binary classifier; (b) Multiclass classifier.

Table 3. The confusion matrix of the DNN based multiclass classifier - Base classifier (*: Estimated class, Unit: %).

	(a *)	(b *)	(c *)	(d *)	(e *)	(f *)	(g *)	(h *)
(a)	96.02	0.24	0.24	0.12	0.47	0.55	1.93	0.43
(b)	0.17	95.62	0.12	0.04	0.29	3.56	0.17	0.04
(c)	1.00	0.96	84.16	1.15	1.71	2.19	6.37	2.47
(d)	0.28	0.20	2.85	77.71	1.68	2.29	1.88	13.11
(e)	2.30	0.99	0.37	1.11	90.60	0.53	3.12	0.99
(f)	0.82	3.61	0.27	0.43	0.93	90.41	2.21	1.32
(g)	4.62	0.20	1.43	0.84	1.79	0.76	89.84	0.52
(h)	0.80	0.04	2.19	4.85	0.64	2.23	3.14	86.12

4.3. Stacking Method

We designed and tested the stacking method (Figure 7). Figure 11a and Table 4 show the accuracy result of the stacking method. We reused the already-trained two base classifier. The learning process of stacking method was smooth that is shown in Figure 11a. The accuracy-loss graph for epoch was very smooth, which means that it is easy to determine which point has the optimal performance. The training accuracy was 92.13%, the validation accuracy was 91.49% and the test accuracy was 91.43%.

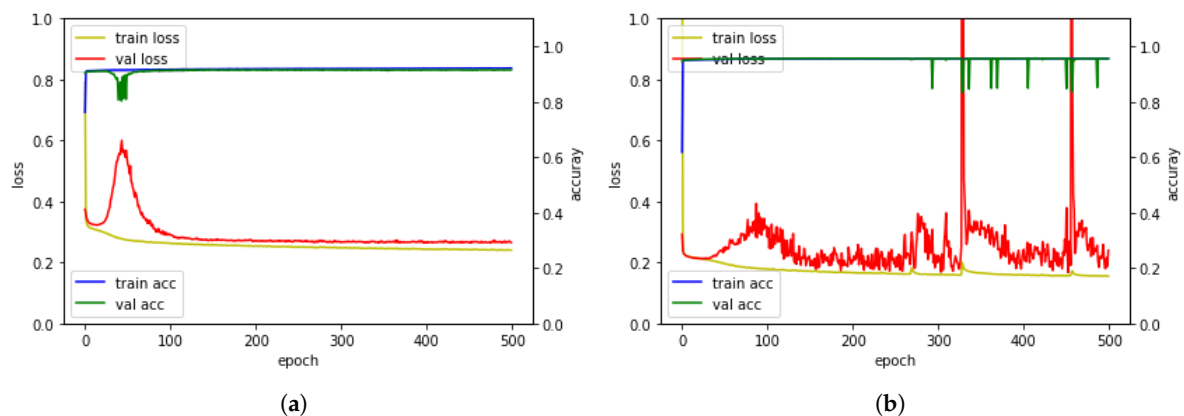


Figure 11. The accuracy and loss graph of training dataset and validation dataset. (a) Stacking method; (b) Modified method.

Table 4. The confusion matrix of the stacking method (*: Estimated class, Unit: %).

	(a *)	(b *)	(c *)	(d *)	(e *)	(f *)	(g *)	(h *)
(a)	95.96	0.08	0.85	0.36	0.44	0.36	1.53	0.40
(b)	0.28	94.90	0.52	0.24	0.28	3.45	0.24	0.08
(c)	0.93	0.43	91.31	2.06	0.27	0.89	2.33	1.79
(d)	0.28	0.16	2.36	85.49	1.34	1.22	1.02	8.14
(e)	0.40	0.65	0.57	1.45	93.58	0.36	1.94	1.05
(f)	0.32	3.69	0.40	0.76	0.44	91.65	1.16	1.57
(g)	1.62	0.21	2.62	1.08	1.62	0.66	90.57	1.62
(h)	0.28	0.00	1.22	7.18	0.75	0.83	1.54	88.21

4.4. Modified Stacking Method

We designed and tested the modified stacking method (Figure 8). Figure 11b and Table 5 show the final results of the modified stacking method. The training accuracy was 95.58%, the validation accuracy was 95.55% and the test accuracy was 95.62%.

Table 5. The confusion matrix of the modified stacking method (*: Estimated class, Unit: %).

	(a *)	(b *)	(c *)	(d *)	(e *)	(f *)	(g *)	(h *)
(a)	97.16	0.00	0.28	0.12	0.20	0.24	1.44	0.56
(b)	0.08	97.75	0.04	0.04	0.00	1.97	0.12	0.00
(c)	0.36	0.04	97.13	0.57	0.24	0.08	0.81	0.77
(d)	0.12	0.04	1.50	90.16	0.63	0.36	0.51	6.68
(e)	0.46	0.21	0.21	0.62	97.52	0.12	0.58	0.29
(f)	0.40	1.50	0.24	0.28	0.24	96.56	0.40	0.40
(g)	0.88	0.00	1.57	0.48	0.84	0.52	94.90	0.80
(h)	0.32	0.00	0.56	4.09	0.36	0.28	0.52	93.89

5. Conclusions and Discussion

We have proposed the stacking method for ensemble learning to improve the classification accuracy of the classifier to recognize micro-Doppler signals. The stacking method is to combines the classification models of different structures. Also, we additionally designed and tested the modified stacking method that reflects the variability of micro-Doppler effects over time.

Table 6 shows the experimental results. The stacking method of ensemble learning showed a performance improvement of about 2.6% (test accuracy) over the DNN based multiclass classifier. Figure 10b showed that the DNN based multiclass classifier could no longer improve its performance due to over-fitting and fluctuation. However, Figure 11a showed that the stacking method could be an effective alternative to optimize. In other words, the stacking method can give better learning process when the learning rate of the DNN is too long or too difficult due to its complex structure. We can get a more efficient learning process through the stacking that properly divides and combines the given structure.

Table 6. The accuracy performances of the tested classifiers.

Network Structure	Training Accuracy	Validation Accuracy	Test Accuracy
SVM (Binary)	99.9%	75.8%	75.2%
DNN (Binary)	93.4%	89.7%	89.7%
SVM (Multiclass)	99.9%	71.8%	71.1%
DNN (Multiclass)	96.9%	88.7%	88.8%
Stacking method (Multiclass)	92.1%	91.5%	91.4%
Modified stacking method (Multiclass)	95.6%	95.6%	95.6%

Lastly, we designed and tested the modified stacking method to improve the accuracy of the classifier. The result showed a performance increase of about 6.8% (test accuracy) over the DNN based multiclass classifier. This result means that the performance can get improvement by using information that reflects the variability of micro-Doppler effects over time.

If we wanted to put more FFT data into the input of the DNN based multiclass classifier, the input node would have increased by k multiple of 512. This approach may lead to an optimization problem due to a more complex structure. On the other hand, the proposed modified stacking method is a useful approach to reuse already learned classifiers (the DNN based classifiers) to combine the micro-Doppler information over time because it does not increase the input node and does not complicate the structure. Therefore, since the proposed modified stacking method can give an alternative to simplify the structure of the classifier's model and this method can provide a useful learning process.

The results of this experiment are based on datasets obtained in a limited environment. Nevertheless, this study showed that using a well-trained DNN can be an effective way to classify classes without any additional signal processing to extract features. Also, we verified the stacking multiple method as an effective alternative to improve the classification accuracy because of being able to reflect the micro-Doppler variability within a given window time. We expect that we can solve more challengeable classification problems by stacking and reusing well-known classifiers already trained.

Author Contributions: J.K. obtained raw data and conducted all the experiments. As an academic adviser, N.K. advised J.K. on the experimental settings, classifier architectures and overall organization of the paper.

Funding: This work was supported by the ICT R&D program of MSIP/IITP, Korean Government (2017-0-00306).

Conflicts of Interest: The authors declare no conflict of interest.

References

- Chen, V.C.; Li, F.; Ho, S.S.; Wechsler, H. Micro-Doppler effect in radar: Phenomenon, model, and simulation study. *IEEE Trans. Aerosp. Electr. Syst.* **2006**, *42*, 2–21. [[CrossRef](#)]
- Kim, Y.; Ling, H. Human activity classification based on micro-Doppler signatures using a support vector machine. *IEEE Trans. Geosci. Remote Sens.* **2009**, *47*, 1328–1337.
- Fioranelli, F.; Ritchie, M.; Gürbüz, S.Z.; Griffiths, H. Feature diversity for optimized human micro-Doppler classification using multistatic radar. *IEEE Trans. Aerosp. Electr. Syst.* **2017**, *53*, 640–654. [[CrossRef](#)]
- Karabacak, C.; Gurbuz, S.Z.; Gurbuz, A.C.; Guldogan, M.B.; Hendeby, G.; Gustafsson, F. Knowledge exploitation for human micro-Doppler classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2125–2129. [[CrossRef](#)]
- Kim, Y.; Ha, S.; Kwon, J. Human detection using Doppler radar based on physical characteristics of targets. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 289–293.
- Van Eeden, W.D.; de Villiers, J.P.; Berndt, R.J.; Nel, W.A.; Blasch, E. Micro-Doppler radar classification of humans and animals in an operational environment. *Expert Syst. Appl.* **2018**, *102*, 1–11. [[CrossRef](#)]
- Du, L.; Li, L.; Wang, B.; Xiao, J. Micro-Doppler feature extraction based on time-frequency spectrogram for ground moving targets classification with low-resolution radar. *IEEE Sens. J.* **2016**, *16*, 756–3763. [[CrossRef](#)]
- Li, Y.; Du, L.; Liu, H. Hierarchical classification of moving vehicles based on empirical mode decomposition of micro-Doppler signatures. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 3001–3013. [[CrossRef](#)]
- Van Dorp, P.; Groen, F.C.A. Feature-based human motion parameter estimation with radar. *IET Radar Sonar Navig.* **2008**, *2*, 135–145. [[CrossRef](#)]
- Chen, V.C.; Tahmoush, D.; Miceli, W.J. Radar micro-Doppler signatures: Processing and applications. *Inst. Eng. Technol.* **2014**, *2*, 139–185.
- Trommel, R.P.; Harmanny, R.I.A.; Cifola, L.; Driessen, J.N. Multitarget human gait classification using deep convolutional neural networks on micro-doppler spectrograms. In Proceedings of the 2016 European Radar Conference (EuRAD), London, UK, 5–7 October 2016; pp. 81–84.
- Kim, Y.; Moon, T. Human detection and activity classification based on micro-Doppler signatures using deep convolutional neural networks. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 8–12. [[CrossRef](#)]

13. Chen, Z.; Li, G.; Fioranelli, F.; Griffiths, H. Personnel recognition and gait classification based on multistatic micro-Doppler signatures using deep convolutional neural networks. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 669–673. [[CrossRef](#)]
14. Kwon, J.; Kwak, N. Human detection by neural networks using a low-cost short-range Doppler radar sensor. In Proceedings of the 2017 IEEE Radar Conference (RadarConf), Seattle, WA, USA, 8–12 May 2017.
15. Kwon, J.; Lee, S.; Kwak, N. Human Detection by Deep Neural Networks Recognizing Micro-Doppler Signals of Radar. In Proceedings of the 2018 15th European Radar Conference (EuRAD), Madrid, Spain, 26–28 September 2018.
16. Wu, Z.; Valentini-Botinhao, C.; Watts, O.; King, S. Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, QLD, Australia, 19–24 April 2015; pp. 4460–4464.
17. Low, C.-Y.; Park, J.; Teoh, A.B.-J. Stacking-Based Deep Neural Network: Deep Analytic Network for Pattern Classification. *IEEE Trans. Cybern.* **2019**, *11*, 1589–1596. [[CrossRef](#)] [[PubMed](#)]
18. Jokanović, B.; Amin, M. IEEE Transactions on Aerospace and Electronic Systems. *IEEE Trans. Aerosp. Electr. Syst.* **2017**, *54*, 180–189.
19. Bryan, J.D.; Kwon, J.; Lee, N.; Kim, Y. Application of ultra-wide band radar for classification of human activities. *IET Radar Sonar Navig.* **2012**, *6*, 172–179. [[CrossRef](#)]
20. Lee, S.; Yoon, Y.J.; Lee, J.E.; Kim, S.C. Human–vehicle classification using feature-based SVM in 77-GHz automotive FMCW radar. *IET Radar Sonar Navig.* **2017**, *11*, 1589–1596. [[CrossRef](#)]
21. Clevert, D.-A.; Thomas, U.; Sepp, H. Fast and accurate deep network learning by exponential linear units (elus). *arXiv* **2015**, arXiv:1511.07289.
22. Ioffe, S.; Christian, S. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1511.07289.
23. Li, D.; John, C.P. Ensemble deep learning for speech recognition. In Proceedings of the Fifteenth Annual Conference of the International Speech Communication Association, Singapore, 14–18 September 2014.
24. Saso, D.; Bernard, Ž. Is combining classifiers with stacking better than selecting the best one? *Mach. Learn.* **2016**, *54*, 255–273.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).