Convolutional Neural Network-Based Target Detector Using Maxpooling and Hadamard Division Layers in FM-Band Passive Coherent Location

Geun-Ho Park[1] · Ji Hun Park[2] · Hyoung-Nam Kim[2,*]

Abstract

The constant false alarm rate (CFAR) has been widely used in radar systems to detect target echo signals because of its simplicity. With the recent development of different types of neural networks (NNs), NN architecture-based target detection methods are also being considered. Several studies related to NN-based target detectors have introduced multi-layer perceptron-based and convolutional neural network (CNN)-based structures. In this paper, we propose a CNN-based target detection method in frequency modulation (FM)-band passive coherent location (PCL). We improved the detection performance using a maxpooling layer and a Hadamard division layer, which are parallelly placed with a CNN layer. Moreover, in our method there is no need to determine the specific cell configuration (e.g., cell under test, reference cells, and guard cells) because the proposed method obtains the trained kernels by end-to-end learning. We show that the trained kernels help in the extraction of either signal or noise components. Through the simulations, we also prove that the proposed method can yield an improved receiver operating characteristic compared to that of a cell-averaging CFAR detector for FM-band PCL in a homogeneous environment.

Key Words: Constant False Alarm Rate, Convolutional Neural Network, Passive Coherent Location, Target Detection.

I. INTRODUCTION

A passive coherent location (PCL) is a type of radar system exploiting third-party transmitters that were initially constructed for broadcasting and communication systems [1–3]. Like other radar receivers, the PCL determines the target presence or absence via the well-known detection technique called the cell-averaging constant false alarm rate (CA-CFAR) [4], which is particularly useful in a homogeneous noise background. PCLs can also employ variations of CFAR, such as greatest of (GO) CFAR, smallest of CFAR, order statistic CFAR, and trimmed-mean CFAR [4], which are useful in various heterogeneous environments.

As the neural network (NN) has been the main focus of interest for researchers in recent years, there have been several trials to construct an NN architecture-based target detector and to obtain a better receiver operating characteristic (ROC) than that of traditional CFAR techniques [5–11]. Among them, [5–8] proposed using multi-layer perceptron (MLP)-based architectures either to construct a target detector or to identify the noise background. In [5], the authors considered the problem of the target detection method using artificial NNs in a K-distributed

environment. The authors in [6] designed a network to determine whether the background environment is homogeneous or heterogeneous, and this network was used only for environmental identification. Akhtar and Olsen [7, 8] trained the MLP-based detectors using the results of CA-CFAR and GO-CFAR. Convolutional neural network (CNN)-based architectures were also suggested in [9–11]. Lin et al. [9] trained a network to estimate the noise variance more accurately, even when the target existed in the range-Doppler domain. The authors in [10] trained the CNN model for target detection in the range-Doppler domain, but their NN determined the target presence or absence under a single range-Doppler map. Gusland et al. [11] presented a CNN-based detector that trained the entire image of the range-Doppler maps in a similar way as in [10].

This paper proposes a new target detection method using CNN architecture in a homogeneous environment for frequency modulation (FM) radio-based PCL. A significant difference in our way compared to previous approaches is the introduction of an input-wise normalization layer. This input-wise normalization layer gives the NN improved detection performance. By comparison, most of the previous works on CNN-based target detectors focused on the conventional CNN structures, which were initially used for the computer vision area. The MLP-based systems are still used for target detection tasks. Compared to the MLP structures, the CNN has advantages in computational complexity because of its sparse connectivity, and this feature allows the kernels to be efficiently trained.

We also show that the CNN model including a fully convolutional layer [12] can be used to efficiently perform cell-wise classification for target detection tasks in the cross-ambiguity function (CAF), similar to several other image processing techniques. The exploitation of a fully convolutional layer helps in training NNs for per-cell classification tasks. This structural feature allows the kernels to be designed in such a way that they are not limited to the traditional cell configuration of CFAR, such as cell under test (CUT), reference cell, and guard cell. Therefore, the cell configuration need not be explicitly considered at all stages.

To the best of our knowledge, our method is the first to use the input-wise normalization layer to improve the CA-CFAR technique. Related works [10, 11] using the CNN model only derive the detection result from a single range-Doppler map. To develop the CFAR-like detector, we propose adding a maxpooling layer and a Hadamard division layer, which are in parallel with the first CNN layer. In this paper, the two layers parallelly placed with the first CNN layer are defined as "an input-wise normalization layer."

Finally, we show that each trained kernel in the proposed method can carry out respective roles to extract the CUT and the sum of reference cells. The performance of the proposed

method is also derived in terms of its detection probability, number of false clusters, and false alarm rate. From the above discussion, we can observe that the ROC of the proposed method outperforms that of CA-CFAR under homogeneous conditions in the FM-based PCL system.

The remainder of this paper is organized as follows. The layer architecture of the proposed method is presented in Section II. The training details are described in Section III. The simulation results are given in Section IV, and we conclude the paper in Section V.

II. LAYER ARCHITECTURE

Fig. 1 illustrates the layer architecture of the proposed detector. The layer architecture has a CNN layer and two fully convolutional layers. The hidden unit of the first convolution layer is a rectified linear unit (ReLu), and the activation function of the output layer is a softmax function, which is applied for all cells of the output. The depth of the proposed layer architecture is shallow; however, it is sufficient to derive satisfactory detection performance, even when using a simple model.

As shown in Fig. 1, the squared magnitude of CAF, $\mathbf{X} \in \mathbb{R}^{M \times N}$, is fed to the input layer of the CNN. We used a two-dimensional rectangular kernel with a size of K (odd number) in the CNN layer, and the size of the feature map is reduced by $K - 1$ in both x (range) and y (Doppler frequency) dimensions because the zero padding is not applied in this step. For example, if the input size of CAF is $M \times N$, then the size of the feature map is $M_0 \times N_0$, where $M_0 = M - K + 1$ and $N_0 = N - K + 1$. To maintain the size of the feature map in the first CNN layer, zero padding can be applied. However, the padded zero values on the edges of the input data may increase the false alarm rate because the addition of zeros makes it appear to have a much lower noise power compared to no zero padding. Therefore, we concluded that it is better not to perform zero padding. At the output of the first CNN layer, the batch normalization [13] can be considered to train deep models, but this is not significant for our architecture because our proposed NN model has a shallow depth compared to recently developed im-
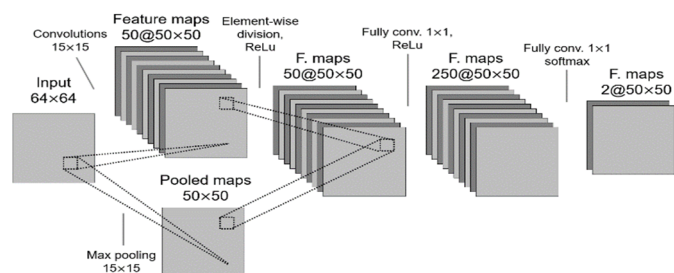

Fig. 1. Layer architecture of the proposed neural network-based target detection method.

age processing techniques.

The fully convolutional layer produces output with the same size as the feature map of the first CNN layer. As we aim to classify all the cells in two categories (i.e., target present and target absent), the kernel dimension of the output is equal to $R \times 2$, where R denotes the number of kernels in the first CNN layer. Finally, the softmax output layer transforms the feature maps to the probabilistic measures, and this result produces the classification labels.

The remarkable difference between our proposed architecture and the NNs for conventional image processing is that our NN architecture performs an input-wise normalization process via the Hadamard division layer and the maxpooling layer. The input-wise normalization constructed by the two layers helps the NN detect cells with relatively higher values than other noise components. As we do not focus on the detection technique with absolute numerical values, the input-wise normalization is sufficient for the target detector in this regard. The maxpooling layer is placed in parallel with the CNN layer. The Hadamard-wise division layer takes the feature maps of the CNN and the maxpooling layer as two inputs. Suppose we have matrices of the output feature map $\mathbf{Y}_r \in \mathbb{R}^{M_0 \times N_0}$ derived from rth kernel; then the output matrix $\mathbf{Z}_r \in \mathbb{R}^{M_0 \times N_0}$ corresponding to the rth kernel can be represented by

$$\mathbf{Z}_r = \mathbf{Y}_r \circ \mathbf{Q}, \; r = 0, 1, \dots, R - 1, \qquad (1)$$

where $\circ$ denotes an operator for Hadamard division (i.e., element-wise division), and $\mathbf{Q} \in \mathbb{R}^{M_0 \times N_0}$ denotes the output of maxpooling to the $K \times K$ grid with stride 1.

A simple example of input-wise normalization, given an input data $\mathbf{X}$, a kernel $\mathbf{W}$, and an output $\mathbf{Y}$ (without padding), is as follows:

$$\mathbf{X} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0.1 \\ 0 & 0.1 & 0 \end{bmatrix}, \mathbf{W} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \mathbf{Y} = \begin{bmatrix} 2 & 0 \\ 0 & 0.2 \end{bmatrix}. \qquad (2)$$

$\mathbf{X}$ and $\mathbf{Z}$ are then calculated by

$$\mathbf{Q} = \begin{bmatrix} 1 & 1 \\ 1 & 0.1 \end{bmatrix}, \; \mathbf{Z} = \mathbf{Y} \circ \mathbf{Q} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}. \qquad (3)$$

This example shows how the input-wise normalization enables detecting the specific features. In (2), $\mathbf{W}$ can be viewed as an extractor for anti-diagonal lines, and thus, the diagonal elements of $\mathbf{Y}$ become 2 and 0.2. However, the diagonal component of 0.2 in $\mathbf{Y}$ is also highlighted up to 2 in $\mathbf{Z}$ using the input-wise normalization process.

The input-wise normalization using maxpooling and Hadamard division has the advantage that it can be easily constructed in most NN frameworks, such as TensorFlow, PyTorch, and MATLAB. These frameworks support the maxpooling layer for various applications, and it is also easy to introduce the input-wise normalization. Without the input-wise normalization, the models might learn only the probability density of the noise process itself.

III. NETWORK TRAINING

1. Data-Generation Process

To train a model using the CNN approach, we consider the input data-generation and desired output generation. As described in the previous section, the input dataset comprises the pieces of magnitude-squared CAF $\mathbf{X}$, including at least one target signal. Fig. 2 shows the FM radio signal-based CAFs. As the instantaneous bandwidth of the FM radio message signal varies, we can see that the range resolution also changes depending on the message signals.

The corresponding desired output represents each cell as either target present or target absent, and therefore the output tensors have a size of $M_0 \times N_0 \times 2$ using one-hot encoding in the output nodes. Examples of the input data and the desired outputs are presented in Fig. 3. Note that the examples of the desired outputs in Fig. 3 (second row) represent binary hypothesis as 1 and 0.
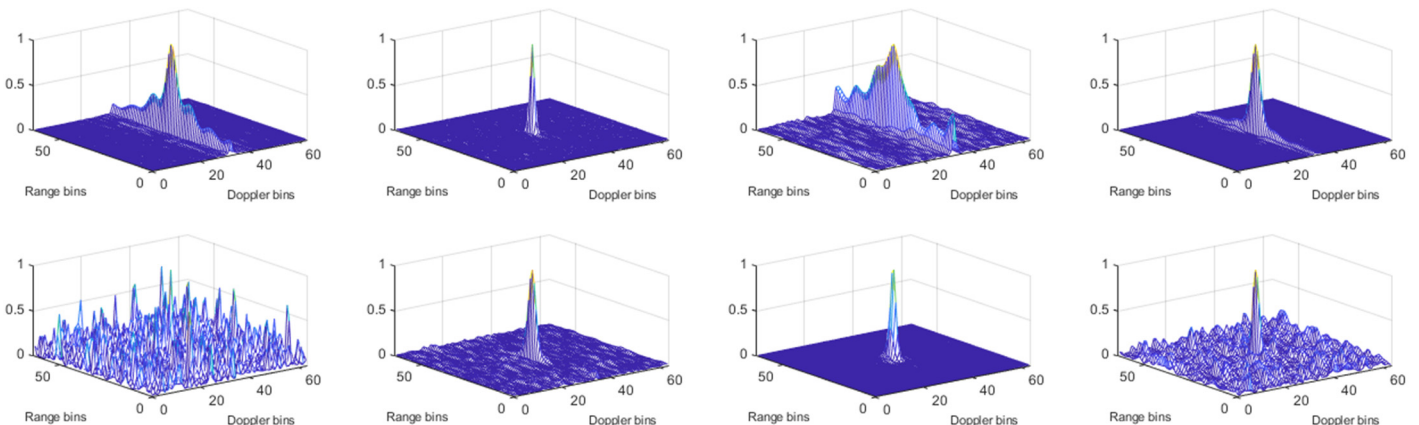

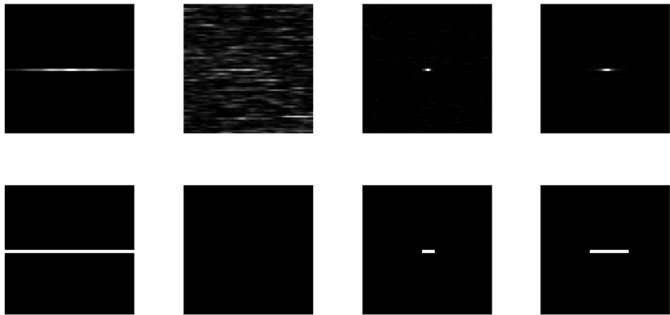Fig. 2. Examples of the FM radio signal-based cross-ambiguity functions.

Fig. 3. Examples of training input images and corresponding desired output images.

In the input data-generation stage, we considered the target signal to have a fractional sample delay and a fractional Doppler frequency shift. The fractional values are applied in the CAF to reflect the actual receiving environment. The element of the CAF, $X_{\tau,\nu}$ for $\tau = -M/2, \ldots, M/2 - 1$ and $\nu = -N/2, \ldots, N/2 - 1$ (M and N are even numbers) can be written as

$$X_{\tau,\nu} = \sum_{k=0}^{K-1} x(k)s^*(k - \tau + \Delta k)e^{-j2\pi(\nu - \Delta \nu)k/f_s}, \quad (4)$$

where K is the number of observation samples, $x(k) = s(k) + w(k)$, $s(k)$ denotes the complex envelope of an FM signal, $w(k) \sim \mathcal{N}(0, \sigma^2)$ denotes an independent and identically distributed (i.i.d.) complex Gaussian random process, $\Delta k \sim \mathcal{U}(-0.5, 0.5)$ ($\mathcal{U}$ stands for uniform distribution) is a fractional sample delay, $\Delta \nu \sim \mathcal{U}(-0.5, 0.5)$ Hz denotes the fractional Doppler frequency shift, and f_s represents the sampling frequency.

In addition to the fractional values, we also varied the signal-to-noise ratio (SNR) of the target echo signal. To train the NN as a robust detector for all SNRs of interest, we randomly determine the SNR of the target echo signal. For our training data-generation, we sampled an SNR value in the uniform distribution of $\mathcal{U}(-50, -10)$ dB.

The result of CA-CFAR can produce the desired output vector $\mathbf{y}_{\tau,\nu}$ for each τ and ν. For example, if the CFAR declares that the target is present, then $\mathbf{y}_{\tau,\nu} = \mathbf{y}_p = [1,0]^T$. If the CFAR declares that the target is absent, then $\mathbf{y}_{\tau,\nu} = \mathbf{y}_a = [0,1]^T$. This is also applied in the proposed CNN detector.

We expect the CNN detector to have a considerably low false alarm rate compared to CA-CFAR; therefore, most false alarms are removed in the desired outputs. Considering the fractional Doppler frequency shift, the initial detection results $\mathbf{y}_{\tau,\nu}$ located in $|\nu| \leq 1$ Hz are only allowed to have $\mathbf{y}_p = [1,0]^T$. Otherwise, $\mathbf{y}_{\tau,\nu} = \mathbf{y}_a = [0,1]^T$. Fig. 2 (first row) shows the desired output sequences corresponding to the input data. As the instantaneous bandwidth of the FM baseband signal is not a constant, we can see that the number of cells for target present in

the bistatic range domain varies with the corresponding message signal.

2. Loss Function

We used a cross-entropy (CE) loss function for CNN training. When we denote $\mathbf{y} = [y_1, \ y_2]^T$ for convenience, we omit the subscripts of τ and ν in $\mathbf{y}_{\tau,\nu}$, and the CE loss function of $L(\mathbf{w})$ with respect to kernels of the NN, $\mathbf{w}$, can be written as

$$L(\mathbf{w}) = -\sum_{n=1}^{2} \beta y_n \log \hat{y}_n + (1 - y_n) \log(1 - \hat{y}_n), \quad (5)$$

where β denotes the coefficient for weighted CE, y_n denotes the binary indicator, and $\hat{y}_n$ represents the predicted probability at the nth output node ($n = 1, 2$). As described earlier, the binary indicators for H_p (target present) and H_a (target absent) are defined by $\mathbf{y}_p = [y_1, y_2]^T = [1,0]^T$ and $\mathbf{y}_a = [y_1, y_2]^T = [0,1]^T$, respectively. A rationale for using β (generally $\beta = 1$) is to solve the class imbalance between H_p and H_a in the desired output labels. If the occurrence of H_p is much less than H_a, then β needs to be increased.

IV. SIMULATION RESULTS

1. Training Input and Label Data-Generation

For the CNN-based detector training, different CAFs are produced using message signals. The training dataset comprised 26,573 range-Doppler images. These images include several types of broadcast content, such as human voices and a wide range of music broadcasted via FM radio transmission in South Korea.

The training data can be generated using the complex envelope of an FM signal and (4). When we denote the sampled message signal as $m(k)$, the complex envelope of an FM signal $s(k)$ can be generated by

$$s(k) = \exp\left(j2\pi\Delta f \sum_{k=0}^{L-1} m(k)\Delta t\right), \quad (6)$$

where $1/\Delta t$ denotes the sampling frequency of 200 kHz, Δf denotes the frequency deviation of 75 kHz, and L is the number of observation samples of 200,000. Note that the coherent processing interval is 1 second.

The desired labels are generated from these constructed images. We used the modified result of two-dimensional CA-CFAR, where the number of guard cells is $(7, 2)$, and the reference cell size is $(0, 5)$ on either side. Because the instantaneous bandwidth of the FM radio signal fluctuates according to the message signal, we do not include cells having the same Doppler frequency as the test cell in the noise variance calculation.

We also designed the CAFs to have a size of 64×64. The output labels have a size of 50×50 because the size of a

rectangular kernel is $K = 15$.

2. Training Phase

In the layer architecture, the number of kernels is set to $R = 50$ at the first CNN layer, and then the kernels have $15^2 \times 50$ coefficients. Note that the biases in the first CNN layer are not updated and are initialized to 0. The output of the first CNN layer is a tensor with a size of $50 \times 50 \times 50$. The first fully convolutional layer takes the feature map of the first CNN layer as the input data, and therefore it has 50×250 weights and 250 biases. The second fully convolutional layer has 250×2 weights and 2 biases. The output of the softmax layer has a size of $50 \times 50 \times 2$, and this provides the final predictions using a parameter γ, which is viewed as a false alarm rate. If $y_1 > \gamma$, then the detector decides H_p. Otherwise, it decides H_a.

In the training phase, a stochastic gradient descent optimizer with the learning rate of 0.01 and the momentum term of 0.9 was used to train our CNN model. To prevent the overfitting issue, we used L^2 parameter regularization with a regularization factor of 10^{-4}. The number of epochs was 100, and the mini-batch size was 128.

Fig. 4 shows the gray-scaled 50 kernels of the first CNN layer. As shown, the trained kernels either perform signal extraction or noise variance calculation. For example, the kernels with a significantly higher value (white color) at the center than in the background can be viewed as CUT. In contrast, the kernels with a much lower value (black color) at the center than in the background can be viewed as the sum of reference cells. This result shows that the kernels act like the CFAR technique.

3. Test Phase

To derive the detection performance of the proposed method, we conducted 10,000 Monte Carlo simulations. The detection results were then averaged using a message signal, which is not included in the training sets.

We measured the false alarm rate when the target is absent. Although the false alarm rate is determined in advance, the ex-
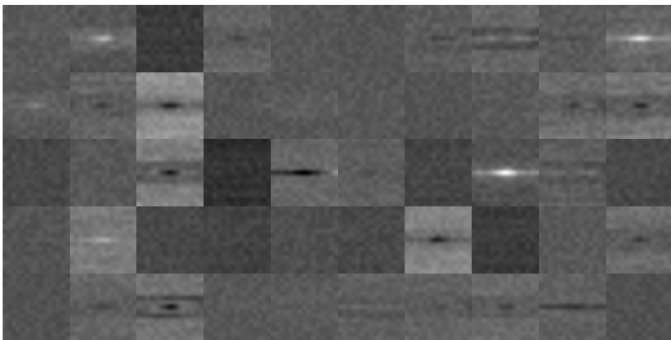
tended target cells (i.e., the correlated cells in the CAF) may change the actual false alarm rate.

We obtain the target present or target absent in two ways. First, we clustered the detected cells as a group and then determined whether the group is the target or not (see Figs. 5 and 6). If the clustered cells are in a specific region, then the detection result is determined as present. Second, we only considered a target cell placed at the (0, 0) position in the range-Doppler map (see Fig. 7). A detailed performance analysis is presented in the next subsection.

4. Performance Comparison

To compare the performance of the CNN detector and CA-CFAR, we calculated the number of false clusters when the target signal was absent. In this paper, the cluster is defined as a group of detected cells that are closely spaced with a cell dis-
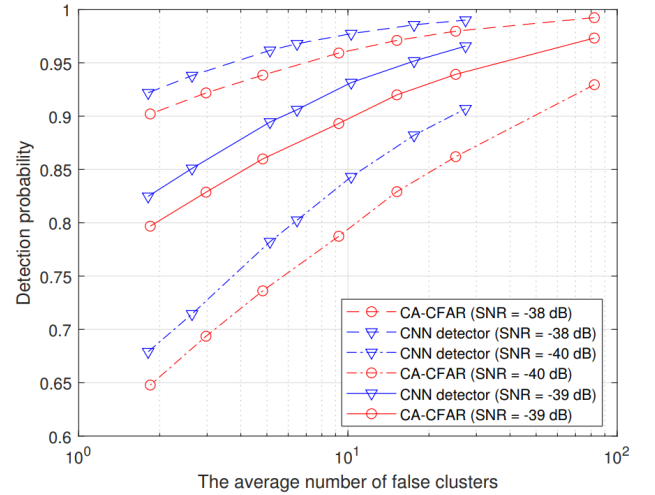

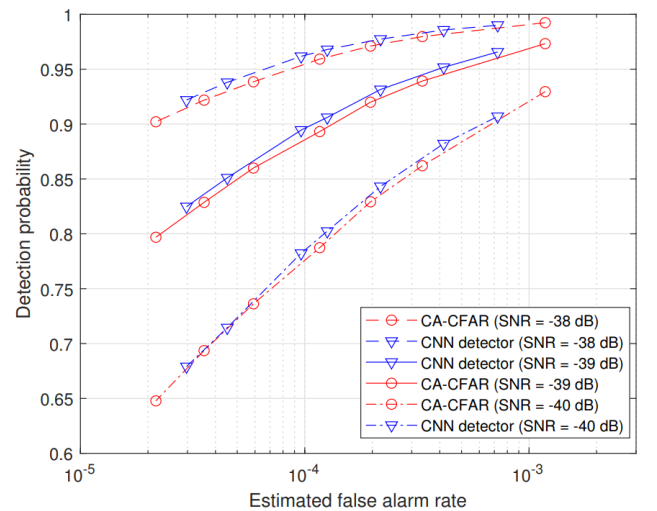Fig. 5. Detection probability of CA-CFAR and CNN detector versus the average number of false clusters.

Fig. 6. Detection probability of CA-CFAR and CNN detector versus the false alarm rate with clustering.

Fig. 4. Trained 50 kernels of first CNN layer with grayscale (white and black colors represent 1 and 0, respectively).
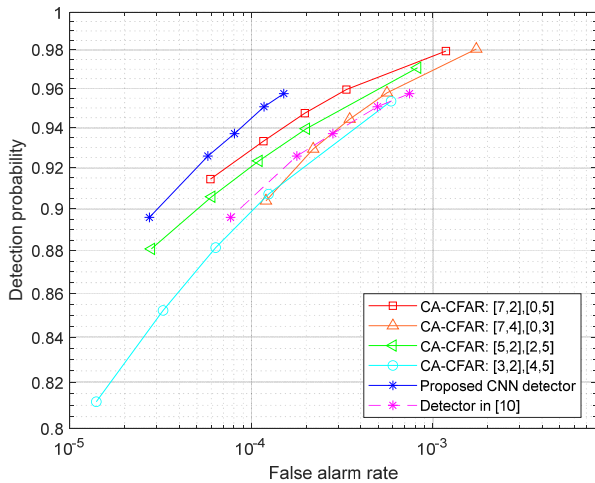
Fig. 7. Detection probability of CA-CFAR detectors and CNN detector versus the false alarm rate without clustering.

tance of 1. The targets are generally detected as the extended cells in the FM radio signal-based CAF. Therefore, we also used the number of false clusters as a performance metric. We obtained 50,000 CAFs with a size of 500×400, and the CAFs were obtained from the reference signal (SNR = 60 dB) and Gaussian noise. Subsequently, we applied the CNN detector and CA-CFAR, and the detected cells were clustered. Finally, we counted the number of clusters and averaged the results. Fig. 5 shows the number of false clusters on the x-axis with respect to $\gamma = 0.1, 0.05, 0.01, 5 \times 10^{-3}, 10^{-3},$ and 10^{-4} and a false alarm rate $p_{fa} = 5 \times 10^{-4}, 10^{-4}, 5 \times 10^{-5}, 10^{-5}, 5 \times 10^{-6},$ and 10^{-6}. The detection probability versus target SNR is also shown in Fig. 5, where we can see that the CNN detector has a higher detection probability at the same number of false clusters. Fig. 6 shows the detection probability versus the false alarm rate. In terms of the false alarm rate, the CNN detector has a slightly better detection performance than that of CA-CFAR.

Fig. 7 shows the proposed method's detection performance, CA-CFAR algorithms with various cell configurations, and the method in [10]. In this case, we considered only the detection result of the target cell. Because the method in [10] uses different size of data from ours, we slightly modified [10]; in this paper, the kernel size of the first CNN layer is 7 and the kernel size of the second CNN layer is 5. As shown in Fig. 7, the proposed method outperforms other detection methods, such as CA-CFAR detectors and the method in [10].

V. Conclusion

We designed a CNN architecture-based target detector for FM radio-based PCL. The proposed architecture includes the maxpooling layer and the Hadamard division layer, which help the detector perform the input-wise normalization. Using the input-wise normalization, we showed that the trained kernels at the first CNN layer perform the extraction for either CUT or the sum of reference cells, as in the CFAR schemes. We also showed that the proposed target detector has a better ROC than CA-CFAR in a homogeneous noise background.

The proposed CNN-based target detector performs the detection process similar to CA-CFAR; therefore, our method also has the limitations of CA-CFAR, and the detection performance of the proposed method is degraded by particular environments, such as the multi-target environment and heterogeneous noise background. Fortunately, the CNN-based layer architecture can be improved by using appropriate training data reflecting the actual receiving environment. Therefore, in our future work, we will expand the CNN-based target detector to a form that can handle the multi-target environment and heterogeneous noise background in various PCL systems.

References

[1] P. Wojaczek, F. Colone, D. Cristallini, and P. Lombardo, "Reciprocal-filter-based STAP for passive radar on moving platforms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, no. 2, pp. 967-988, 2018.

[2] J. H. Huang, J. L. Garry, and G. E. Smith, "Array-based target localisation in ATSC DTV passive radar," *IET Radar, Sonar & Navigation*, vol. 13, no. 8, pp. 1295-1305, 2019.

[3] C. Ilioudis, C. Clemente, and J. Soraghan, "GNSS-ased passive UAV monitoring: a feasibility study," *IET Radar, Sonar & Navigation*, vol. 14, no. 4, pp. 516-524, 2020.

[4] P. P. Gandhi and S. A. Kassam, "Analysis of CFAR processors in nonhomogeneous background," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 4, pp. 427-445, 1988.

[5] K. Cheikh and F. Soltani, "Application of neural networks to radar signal detection in *K*-distributed clutter," *IEE Proceedings–Radar, Sonar and Navigation*, vol. 153, no. 5, pp. 460-466, 2006.

[6] Q. Qi and W. Hu, "One efficient target detection based on neural network under homogeneous and non-homogeneous background," in *Proceedings of 2017 IEEE 17th International Conference on Communication Technology (ICCT)*, Chengdu, China, 2017, pp. 1503-1507.

[7] J. Akhtar and K. E. Olsen, "A neural network target detector with partial CA-CFAR supervised training," in *Proceedings of 2018 International Conference on Radar (RADAR)*, Brisbane, Australia, 2018, pp. 1-6.

[8] J. Akhtar and K. E. Olsen, "Go-CFAR trained neural network target detectors," in *Proceedings* of *2019 IEEE Radar Conference (RadarConf)*, Boston, MA, 2019, pp. 1-5.

[9] C. H. Lin, Y. C. Lin, Y. Bai, W. H. Chung, T. S. Lee, and H. Huttunen, "Dl-CFAR: a novel CFAR target detection method based on deep learning," in *Proceedings of 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, Honolulu, HI, 2019, pp. 1-6.

[10] L. Wang, J. Tang, and Q. Liao, "A study on radar target detection based on deep neural networks," *IEEE Sensors Letters*, vol. 3, no. 3, article no. 7000504, 2019. https://doi.org/10.1109/LSENS.2019.2896072

[11] D. Gusland, S. Rolfsjord, and B. Torvik, "Deep temporal detection: a machine learning approach to multiple-dwell target detection," in *Proceedings of 2020 IEEE International Radar Conference (RADAR)*, 2020, pp. 203-207.

[12] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, 2015, pp. 3431-3440.

[13] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2017.

Geun-Ho Park

received B.S., M.S., and Ph.D. degrees in electronic and electrical engineering from Pusan National University in Busan, South Korea in 2013, 2015, and 2020, respectively. From March 2020 to December 2020, he was a researcher with the Department of Electrical and Computer Engineering at Pusan National University. From December 2020 to November 2020, he was a researcher at the Korea Research Institute for Defense Technology Planning and Advancement (KRIT). He is currently working as a senior researcher at Hanwha Systems. His research interests include radar signal processing, array signal processing, electronic warfare systems, reinforcement learning and deep learning.

Hyoung-Nam Kim

received B.S., M.S., and Ph.D. degrees in electronic and electrical engineering from Pohang University of Science and Technology in Pohang, South Korea in 1993, 1995, and 2000, respectively. From 2000 to 2003, he was with the Electronics and Telecommunications Research Institute in Daejeon, South Korea developing advanced transmission and reception technology for terrestrial digital television. In 2003, he joined the faculty of the Department of Electronics Engineering at Pusan National University in Busan, South Korea, where he is currently a full-time professor. From 2009 to 2010, he was with the Department of Biomedical Engineering at Johns Hopkins University School of Medicine as a visiting scholar. From 2015 to 2016, he was a visiting professor with the School of Electronics and Computer Engineering at the University of Southampton in the UK. His research interests are in radar/sonar signal processing, machine learning, adaptive filtering, biomedical signal processing, digital communications, electronic warfare support systems, and brain–computer interfaces. He is a member of IEEE, IEIE, and KICS.

Ji Hun Park

received a B.S. degree in electronic engineering in 2020 from Pusan National University in Busan, South Korea, where he is currently working toward an M.S. degree in Electrical and Electronics Engineering at Pusan National University. His research interests include radar signal processing, electronic warfare systems, and deep learning.