Training of Neural Network Target Detectors Mentored by SO-CFAR

Jabran Akhtar
Norwegian Defence Research Establishment (FFI), Kjeller, Norway
Email: jabran.akhtar@ffi.no

***Abstract*—A desirable objective in radar detection theory is the ability to detect and recognize targets in intricate scenarios such as in the presence of clutter or multiple closely spaced targets. Herein we propose the use of artificial neural networks for radar target detection where Smallest Of (SO)-CFAR detector is used as the basis for neural network training. The SO-CFAR detector has exceptional good detectional capabilities, however, suffers from a very high false alarm rate and has therefore only been given limited attention in the literature. We show that by appropriately training a neural network on SO-CFAR detections it is possible to significantly lower the false alarm rate with only marginal decrease in probability of detection.**

***Index Terms*—Constant false alarm rate (CFAR), radar, detection, Swerling targets, neural network**

I. Introduction

The ability to discover targets at a great distance is one of the key features of radar systems. Disclosing potential objects in a reliable manner with a high probability of detection (P_D) and a fixed low-false alarm rate (P_{FA}), however, still remains a challenging and difficult task. A standard technique employed for target detection is through the use of constant false alarm rate (CFAR) detectors [1], [2] with many proposed variants. CFAR detectors provide an adaptive mean to calculate the detection threshold as fixed thresholds are generally inadequate in case of complex and dynamic environments. Among the more popular versions of the CFAR detectors we find the CA (Cell Averaging) and GO (Greatest Of) detectors while the SO (Smallest Of)-CFAR detector is another alternative.

Over the last years, the use of machine learning methods have been discussed in radar settings to perform target detection [3], [2], [4], [5], [6], [7], [8], [9] using assorted strategies. In [7], [8], the authors proposed how one can train feedforwarding neural networks to return the same type of detectional performance as of cell averaging (CA) or greatest of (GO)-CFAR while otherwise aiming to reduce the number of false detections. In the initial study [7] CA-CFAR was considered and the training and evaluation was carried out on a simple signal model consisting of fluctuating targets in noise only conditions. The basic idea was later expanded [8] to include clutter and GO-CFAR was employed as the main training mentor for neural network training. Both CA- and GO-CFAR are prominent detectors in radar detection theory and widely applied on practical systems, however, they do have several issues detecting targets in more complicated settings [1]. Training a neural network on these detectors also implies that the positive detectional performance is upper limited by the original detector's capability. This can be observed in [8] where the training methodology was adapted to improve the P_D but which always came with a penalty of a higher P_{FA}. The previous papers also did not give much attention to the constant false alarm property which, although present, was not verified thoroughly. In this paper, we present a further expansion of the previous works to a more wider and generic setting which is explored in greater detail. To this end, the SO-CFAR detector is utilized as the main foundation for network training with an amended training procedure. The SO-CFAR detector offers a high P_D and has the ability to detect targets in the vicinity of an one-sided clutter edge and adjacent closely spaced targets albeit with an excessive P_{FA}. The main contribution of this paper is accordingly related to the development of training techniques for neural network detectors who are able to deliver a detectional performance recreating SO-CFAR while concurrently curtailing the number of false positives stemming from SO-CFAR. An extension of the method to incorporate cells above and below the cell under test is also proposed.

II. Radar and signal model

To provide a framework for the neural network training and assessment, we briefly describe the model of a standard pulsed radar system where a waveform is emitted at regular intervals. M pulses are assumed transmitted in a coherent processing interval (CPI) and the targets are assumed to be slowly fluctuating with a Swerling 1 distribution where the values vary randomly from dwell to dwell but with a given mean signal-to-noise ratio (SNR). The radar is assumed to construct a standard range-Doppler map for each CPI represented by an $M \times R$ complex matrix $\mathbf{D}(t, \omega)$, where $t = 1, 2, ..., R$. t is the discrete fast-time parameter corresponding to different time delays and consequently range cells while $\omega = 1, ..., M$ gives the Doppler bins.

A. CFAR detector and neural network

A CFAR detector takes the square law range samples of a range-Doppler map $\mathbf{D}$, $\hat{\mathbf{D}}(t, \omega) = |\mathbf{D}(t, \omega)|^2 \ \forall \ t, \omega$, and then processes each individual cell of $\hat{\mathbf{D}}(t, \omega)$ in order to determine if conditions for declaring a detection are fulfilled or not. In this text, we follow an approach where an one dimensional SO-CFAR detector is engaged with averaging conducted across the range domain. A sliding window of size $2N + 2G + 1$ is moved across all possible cells, $\hat{t} = 1 + N + G, ..., R - N - G$ and $\hat{\omega} = 1, ..., N$ excluding potential edges. The $2N + 2G +$

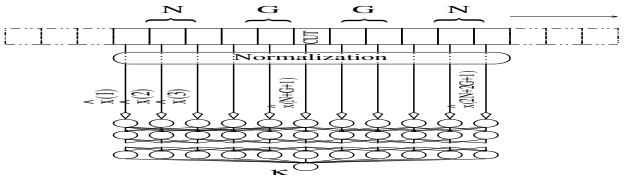

Fig. 1: Neural network detector

1 samples in range specified by the window are extracted in $x(u) = \hat{\mathbf{D}}(\hat{t} - N - G : \hat{t} + N + G, \hat{\omega}),\ u = 1, 2, ..., 2N + 2G + 1$ and the cell in the middle of the window, $x(N + G + 1)$, cell under test (CUT), is compared against a scaled average. G number of guard cells immediately to the right and left of CUT are ignored. The average, γ, in smallest of cell averaging is computed as the smallest average of either the N reference cells to the left or right of CUT,

$$\gamma = \frac{1}{N}\ \min\ (\ \textstyle\sum_{k=1}^{N} x(k), \sum_{k=N+2G+2}^{2N+2G+1} x(k)\ ).$$

A detection is declared if

$$x(u)_{|u=CUT} > \gamma\ K, \tag{1}$$

where K (dB) is a selected threshold. The CUT in SO-CFAR detector is always compared against the smallest value coming from the two reference windows which helps to reduce target masking if multiple targets are present in the CFAR window. As the detectional performance of SO-CFAR can be considered rather good, the objective of a neural network should be to preserve as much as possible while aiming to reduce the number of false detections.

For the artificial neural network, we consider a setup where the sliding CFAR window is moved across the range-Doppler map, as previously; but the selected $2N+2G+1$ window values, including guard cells, are normalized and fed directly into a fully connected feedforwarding network. The normalization of the CFAR window is carried out via standard min max normalization $\hat{x}(u) = \frac{x(u) - \min(x(u))}{\max(x(u)) - \min(x(u))}$. The output from the last layer of a neural network, $\kappa = f_{NN}(\hat{x})$, returns a single detection estimate. $f_{NN}(\hat{x})$ corresponds to the neural network modeled as a function with an input of the normalized CFAR window $\hat{x}(u)$. The process is depicted in figure 1. A threshold test is applied on κ and if exceeded then a detection is declared.

B. Training of neural network detector

The main parameter assumed fixed during training is the threshold value K. This determines the P_D and P_{FA} for the standard SO-CFAR algorithm and is the performance objective the network educates around. For training of the network we assume that a large number of independent range-Doppler maps have been gathered wherein the targets and their positions are known precisely. From each realization, representative random SO-CFAR windows are taken to construct a large training database. The training objective of the neural network will be on binary values [7] and the desired output from the network for each CFAR window of training data is either 0 (no detection) or 1 if a target is present. Categorically, the neural network's last layer trains to return:

$$\hat{\kappa}\ =\ \begin{cases} 1, & \text{SO-CFAR returns a positive detection } \textit{and} \\ & \text{a target is present at CUT.} \\ \\ 0, & \text{otherwise.} \end{cases}$$

The positive outcome in the above interacts with SO-CFAR for target detection, however, just as important is the fact that the network now learns to return a zero output in the absence of real targets regardless of the CFAR outcome. This forces the network to evolve an internal mechanism to try to distinguish between true targets and false CFAR detections. At its best, such a technique should be able to offer the same type of P_D performance as of the original CFAR method but with a lower P_{FA}. This is thus ideally suited for SO-CFAR which has a high P_{FA} and needs to be curtailed for practical purposes. For clarity, we point out that the condition "target is present at CUT" must also include neighboring cells if the known target spreads out in range and/or Doppler due to sidelobes.

The various window samples within the training database can be split into multiple categories:

- A: target is present in CUT and positive SO-CFAR
- B: clutter environment, false SO-CFAR detection
- C: noise only environment, false SO-CFAR detection
- D: random sample, no SO-CFAR detection.

For each category we denote A_N, B_N, C_N and D_N to refer to the number of samples for that particular set with the CFAR window samples denoted by $\hat{x}_{A,k},\ \ k = 1, ..., A_N$ for set A and so forth. Other categorizations and a further refinement is obviously also possible; particularly if there are certain type of detections or incorrect detections who need extra detailing. A neural network training on the above database will in principle be an optimization process with the objective to minimize the overall error which can be decomposed as,

$$\begin{aligned} \min f_{NN|\hat{\mathbf{x}}} = &\sum_{k=1}^{A_N} (|f_{NN}(\hat{x}_{A,k})| - 1)^2 + \sum_{k=1}^{B_N} |f_{NN}(\hat{x}_{B,k})|^2 \\ &+ \sum_{k=1}^{C_N} |f_{NN}(\hat{x}_{C,k})|^2 + \sum_{k=1}^{D_N} |f_{NN}(\hat{x}_{D,k})|^2. \end{aligned}$$

It is somewhat unlikely that a neural network, conditioned by SO-CFAR, can optimize all the different categorizes without negative influence on another category. If there are uneven number of samples from the different categories then some categories may also end up offering poorer performance compared to others. The number of samples for each set will thus have a large impact on what type of detections the network will prioritize to learn and by alternating these parameters different trade-offs can be achieved. The number of layers and nodes in the network also play a vital role in the training process. Very small networks may not converge well over all categories while more flexibility is achieved with greater number of layers or with a wider network at the expense of more resource intensive training. The network is still performance bounded by SO-CFAR and a bigger network also has a disadvantage in that

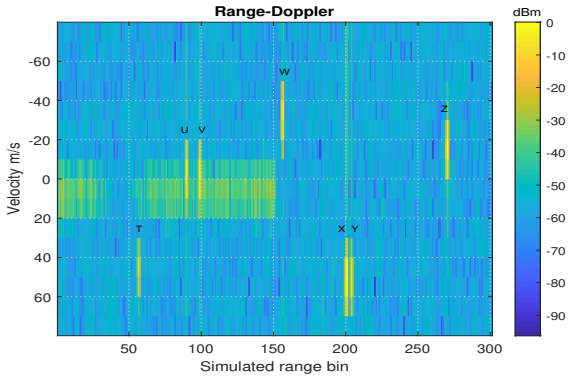

Fig. 2: Simulated range-Doppler map

it may start to optimize too much with regard to given data and not generalize well. The next section demonstrates how the various training parameters can be adjusted and how this impacts the end performance with different trade-offs.

III. Simulations and results

Parts of a pulsed radar system are simulated to train neural networks under the proposed methodology and then the performance is compared against traditional SO-CFAR and GO-CFAR detection. The radar is assumed to transmit and receive $M = 16$ pulses over 300 simulated range bins. In total, seven independently fluctuating targets are modeled being placed at various range bins. The targets' reflectivity is assumed to follow a standard Swerling 1 model where the mean is varied randomly during training to mirror different power levels. The clutter is modeled using a K-distribution function [6] and covers the first half of range bins split in two patches. The clutter shape parameter is randomly selected uniformly for each dwell to be in the range between $v = 0.05$ (spiky) and $v = 10$ (Rayleigh distributed) and the clutter values are then arbitrary generated. A random process additionally up or down scales the clutter to implement more variation in signal-to-clutter ratio from dwell-to-dwell. To construct the range-Doppler map Hamming window is put to use. We refer to figure 2 for an example where all seven targets stand out with high SNR and are designated from T to Z. Targets T to W are considered to be in a region with clutter and the target velocity is randomly set between -45 m/s to 45 m/s. The range placement of T and W is randomly determined to be between 0 to 9 bins to the left or right of range bins, respectively, 60 or 160. Positive detection samples will thus experience cases of clutter edge with different distance from CUT from both sides. Targets X, Y and Z are in the noise only environment with a random velocity between -60 m/s to 60 m/s. Targets U,V and X,Y are closely spaced targets with identical velocities and the distance between U and V or X and Y is randomly set between 3 to 10 range cells on either side of U or X. With a probability of 0.5 these targets have correspondingly equal power levels. Closely spaced targets both with matching and dissimilar energies are thus taken into account. All targets, with a probability of 0.5 spread across two

Case	A_N	B_N	C_N	D_N	Total training samples
1	200000	0	0	200000	400000
2	200000	20000	20000	200000	440000
3	200000	300000	300000	200000	1000000

TABLE I: Neural network training parameters

range-cells and have a single sidelobe in range of -20dB. If the target spreads out in range then a neighboring cell follows an independent Swerling 1 value from the same distribution before a sidelobe is encountered. The noise floor is also not kept fixed, rather ranges between -80 dB to -115 dB following a uniform distribution between CPIs. The variation in the above setup captures the essence of the wide types of CFAR windows which should be suitable for training a generic type of detector aimed toward targets and closely spaced targets in noise and clutter.

After formation of a range-Doppler map, CFAR tests are performed and the blocks extracted for the training database, the SO-CFAR parameters being set as $G = 3$ guard cells and $N = 9$ averaging cells on each side and thresholding level of $K = 14$dB. True and false detections of type A, B and C were taken into the training database as encountered sequentially while random samples of type D were only taken with a probability of 5%. New random range-Doppler maps were generated for as long as required to fulfill conditions for the training database under different numbers of A_N, B_N, C_N and D_N as defined in table I.

We remark that the vast majority of false detections in the presented setup are likely to come from the clutter region in contrast to noise only arenas. To force the networks to have interchangeable false alarm rate performance, false detections from clutter and non-clutter regions are taken in equal measure, i.e. $B_N = C_N$ was required. Exceptionally, training case 1 only trains on category A and D type results and such a network should ideally converge to the base case of SO-CFAR. As false detections are introduced, the network should start learning to identify incorrect SO-CFAR detections and be able to improve upon the false alarm rate of SO-CFAR, at possible expense of a lower probability of detection. To train, a fully connected feedforwarding networks with 2 hidden layers and 75 nodes in each layer were utilized, each node using the hyperbolic tangent as the activation function, for up to 1 million epochs. Wider networks generally offer better detectional performance [8], [9] and the number of nodes in each layer was set to three times the number of inputs.

With the initial training process completed, a new larger set of 6400 range-Doppler maps was generated using the same described principles but now with a set mean power value for the targets. Each resulting map was in full evaluated through SO-CFAR, GO-CFAR and the trained neural network to build up performance statistics. The neural network detection thresholds utilized were $\kappa > 0.5$, $\kappa > 0.8$ and $\kappa > 0.95$. This process was repeated with varying average target power levels to obtain probability of detection and false alarm rate curves with respect to mean target signal-to-clutter plus noise ratio

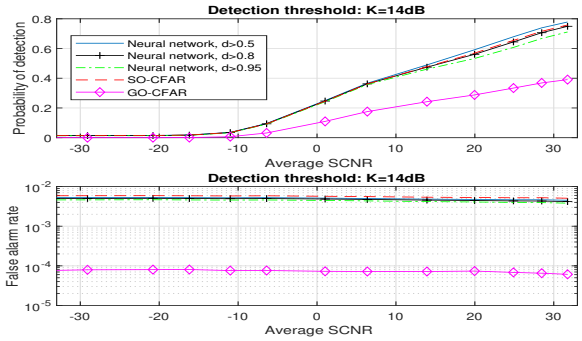

Fig. 3: Case 1: P_D and P_{FA}

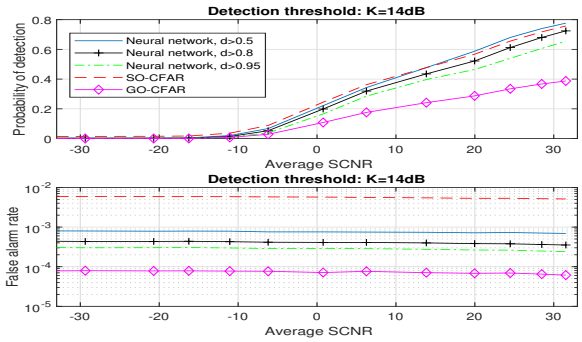

Fig. 4: Case 2: P_D and P_{FA}

(SCNR). P_D was calculated as the number of correctly detected targets relative to the total number of simulated targets while P_{FA} as the number of incorrectly detected targets in relation to total number of tests (27.5 millions per SCNR).

Figures 3 to 5 depict the P_D and P_{FA} results from the three trained networks. As can be seen in figure 3, training with $B_N = C_N = 0$ results in a very close match against standard SO-CFAR shown in red. Setting a low threshold of $\kappa = 0.5$ (blue curve) can give a slight improvement in P_D but otherwise the network is acting similar as to the original SO-CFAR detector. Although the P_D is good, the P_{FA} is much

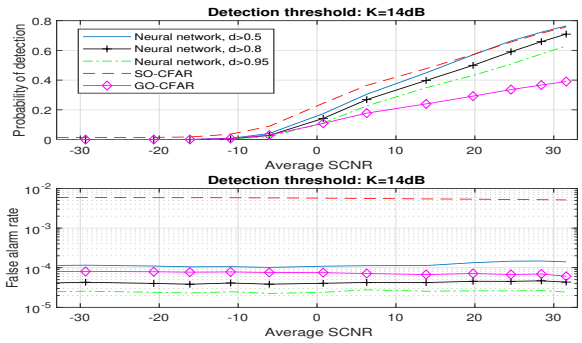

Fig. 5: Case 3: P_D and P_{FA}

degraded compared to GO-CFAR (magenta).

By training the network mutually on 10% incorrect detections, split equally across clutter and noise regions, the curves deviate moderately from SO-CFAR (figure 4). The P_D curve is now a bit lower than SO-CFAR with the thresholds of $\kappa = 0.95$ (green) or $\kappa = 0.8$ (black) while with $\kappa = 0.5$ it is still comparable to the SO-CFAR performance. The main benefit, nevertheless, stems from the P_{FA} curves which are now situated in-between SO-CFAR and GO-CFAR. They enhancement in P_{FA} is relatively much greater than the marginal decrease in P_D. We remark that the P_{FA} curves are also all relatively flat. The third training case with greater emphasis on false detections returns figure 5 on evaluated data. The P_D curves are now further shifted downwards, though the performance is still favorable, and is for all the three threshold values still above GO-CFAR. Impressively, the P_{FA} from the neural networks is now more on par with, or lower than, GO-CFAR. The trained network can thus roughly approximate SO-CFAR detectional performance with GO-CFAR false detection rate.

To further evaluate the performance of case 3 trained neural network, detailed simulations were carried out with different target and clutter settings. Only a few selected targets were included in the simulations to assess specific detectional capabilities. For these simulations, the SCNR was randomly varied, as in the described model; and the P_D and P_{FA} results are provided in tables II and III. The top three simulation settings do not include any clutter and thus evaluate the neural network performance in noise only setup while the remaining cases also incorporate clutter. The false alarm rate performance from both SO and GO detectors is quite satisfactory as long as no clutter is considered at which it starts to suffer considerably either in P_D or P_{FA} or both. The trained neural network exhibits a more consistent performance throughout but also obtains the highest P_D rate either with a single target or closely spaced targets in noise only environment (targets T,W,Z or targets U,V and X,Y). When only a single target (U) is present in the clutter region then the lowest P_D results is obtained. Evidently, the

Targets	NN $\kappa>0.5$	NN $\kappa>0.8$	NN $\kappa>0.95$	SO	GO
All, no clutter	0.83	0.78	0.69	0.76	0.40
T, W, Z, no c.	0.82	0.79	0.74	0.81	0.78
U, V, X, Y, no c.	0.83	0.77	0.66	0.81	0.16
All w/clutter	0.79	0.73	0.65	0.78	0.41
U w/clutter	0.70	0.67	0.62	0.70	0.66
U, V w/clutter	0.74	0.68	0.57	0.72	0.15
T w/clutter	0.76	0.72	0.66	0.78	0.65

TABLE II: Performance comparison, P_D

Targets	NN$\kappa>0.5$	NN$\kappa>0.8$	NN$\kappa>0.95$	SO	GO
All, no clut.	$1.13{\cdot}10^{-4}$	$3.25{\cdot}10^{-5}$	$1.79{\cdot}10^{-5}$	$9.39{\cdot}10^{-6}$	$7.69{\cdot}10^{-8}$
T, W, Z, "	$2.85{\cdot}10^{-4}$	$1.11{\cdot}10^{-5}$	$7.42{\cdot}10^{-6}$	$1.12{\cdot}10^{-5}$	$3.71{\cdot}10^{-8}$
U, V, X, Y,"	$8.75{\cdot}10^{-5}$	$2.89{\cdot}10^{-5}$	$1.70{\cdot}10^{-5}$	$1.19{\cdot}10^{-5}$	$3.74{\cdot}10^{-8}$
All w/clut.	$1.90{\cdot}10^{-4}$	$6.95{\cdot}10^{-5}$	$3.72{\cdot}10^{-5}$	$5.08{\cdot}10^{-3}$	$6.39{\cdot}10^{-5}$
U, "	$1.34{\cdot}10^{-4}$	$6.49{\cdot}10^{-5}$	$3.82{\cdot}10^{-5}$	$5.96{\cdot}10^{-3}$	$6.98{\cdot}10^{-5}$
U, V, "	$1.55{\cdot}10^{-4}$	$6.90{\cdot}10^{-5}$	$4.17{\cdot}10^{-5}$	$6.00{\cdot}10^{-3}$	$6.29{\cdot}10^{-5}$
T, "	$1.00{\cdot}10^{-4}$	$4.95{\cdot}10^{-5}$	$2.92{\cdot}10^{-5}$	$4.97{\cdot}10^{-3}$	$6.88{\cdot}10^{-5}$

TABLE III: Performance comparison, P_{FA}

last situation is only indirectly simulated during training when it is assumed that the closely spaced targets have independent fading with a probability of 0.5. Explicit training on a single target in clutter could therefore be carried out if there is a desire for further improvement. The same two cases with highest P_D also return the lowest P_{FA} where the values are otherwise very equivalent. The main exception here is the GO-CFAR detector, which performs competently for a single target in noise and generally has a low P_{FA} but is, notably, not able to detect many closely spaced targets.

A. Doppler 2D extension

In a range-Doppler map, targets are not necessarily located at a single point but will often exhibit some spread in Doppler. The extend of this will depend on several factors such as the applied window function. It can nevertheless be beneficial to also provide the Doppler sidelobe bins to the neural network for training and detection purposes even though the CFAR test and training criteria remains restricted to the range domain. To investigate this aspect, the 2 Doppler cells above and below the CUT were taken account of, in a circular manner, and fed alongside the standard CFAR window into the neural network. Keeping the other parameters fixed with $G = 3$ guard cells and $N = 9$ averaging cells this increases the number of samples for each test to 29. For a fair comparison, the same neural network structure of 2 hidden layers and 75 nodes in each layer was utilized. The training process was otherwise identical.

The final results on the same evaluation data are shown in figures 6 and 7. Comparing figure 3 to figure 6 and figure 4 to figure 7, the limited extension to two dimensions is not able to improve much upon the P_D. This can be expected, as the SO-CFAR test for positive detection is unchanged but the extra information can still aid in recognizing cases where a false SO-CFAR detection may occur. This can be observed in the plots where the false alarm rate curves show a clear improvement for the same SCNR compared to the previous plots. The enhancement is noticeable but small for case 1 when no training is carried out on false detections but is more apparent for case 2. Particularly with $\kappa = 0.8$, the P_D loss compared to SO-CFAR is only 1-2dB while the P_{FA} is just slightly worst off compared to GO-CFAR. A neural network can easily be extended to consider such additional factors requiring few extra resources while these aspect would be rather difficult to administer via conventional means.

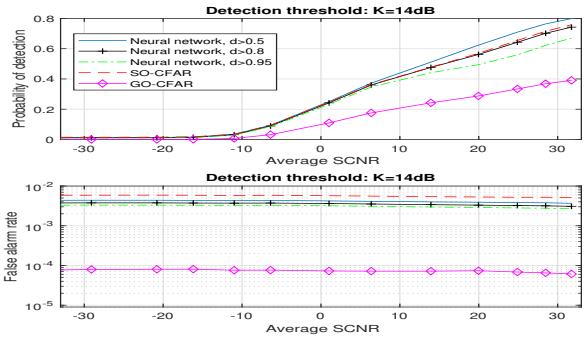

Fig. 6: Case 1: P_D and P_{FA}, 2D extension

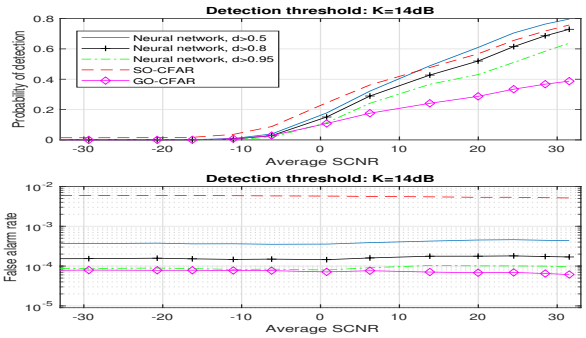

Fig. 7: Case 2: P_D and P_{FA}, 2D extension

IV. Conclusion

In this paper, artificial neural networks were considered as replacement for traditional CFAR detectors. A new training scheme based on SO-CFAR was proposed on how to efficiently train such networks. The scheme can be used to obtain a desired trade-off between probability of detection and false alarm rate. The neural network improvement is most noticeably linked to a lower false alarm rate which can be attributed to the networks' ability to recognize cases where SO-CFAR can erroneously return a false detection. This proposed approach allows the network to preserve basic features of the CFAR detection process being able to identify closely spaced targets and take account of clutter edges while concurrently providing an easy to implement training methodology.

References

[1] H. Rohling, "Radar CFAR thresholding in clutter and multiple target situations," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 19, no. 4, pp. 608–621, 1983.

[2] P. P. Gandhi and V. Ramamurti, "Neural networks for signal detection in non-gaussian noise," *IEEE Trans. Signal Processing*, vol. 45, no. 11, pp. 2846–2851, Nov. 1997.

[3] F. Amoozegar and M. Sundareshan, "A robust neural network scheme for constant false alarm rate processing for target detection in clutter environment," in *Proc. American Control Conference*, 1994.

[4] G. López-Risueño, J. Grajal, S. Haykin, and R. Díaz-Oliver, "Convolutional neural networks for radar detection," in *Internation Conference on Artificial Neural Networks*, 2002, pp. 1150–1155.

[5] N. Galvez, J. Pasciaroni, O. Agamennoni, and J. Cousseau, "Radar signal detector implemented with artificial neural networks," in *Proc. XIX Congreso Argentino de Control Automatico*, 2004.

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

[7] J. Akhtar and K. E. Olsen, "A neural network target detector with partial CA-CFAR supervised training," in *Proc. of International Conference on Radar*, 2018.

[8] ——, "GO-CFAR trained neural network target detectors," in *Proc. of IEEE Radar Conference*, 2019.

[9] M. Carretero, R. Harmanny, and R. Trommel, "Smart-CFAR, a machine learning approach to floating level detection in radar," in *Proc. 16th European Radar Conference*, 2019.

[10] M. M. Horst, F. B. Dyer, and M. Tuley, "Radar sea clutter model," in *Proc. of the Intl. IEEE AP/S URSI Symposium*, 1978, pp. 6–10.