Improved Detection by Peak Shape Recognition Using Artificial Neural Networks

Stefan Wunsch, Johannes Fink, Friedrich K. Jondral
Communications Engineering Lab, Karlsruhe Institute of Technology
Stefan.Wunsch@student.kit.edu, Johannes.Fink@kit.edu, Friedrich.Jondral@kit.edu

***Abstract*—Conventional peak detection algorithms are not designed to include information on the expected peak shape. Therefore, commonly used detectors discard this valuable information and do not perform optimally in regard to the given possibilities. Designed and evaluated is a detector based on an artificial neural network, which is employed for pattern recognition in order to exploit the peak shape information. The detector outperforms the best detector using no peak shape information with a detection rate increase of up to 10% at a constant false alarm rate. The proposed detection method is compared with a threshold detector and an ordered statistics constant false alarm rate (OS-CFAR) detector commonly used in radar. The introduced detector provides useful information on the reliability of the peak detection. Furthermore, it is shown that the neural network based detection mechanism is easily employable on hardware because no knowledge about the signal-to-noise ratio of the training data is needed.**

***Index Terms*—Peak Shape, Detection, Neural Network, Radar, DSP**

I. Introduction

Peak detection is a fundamental part of radar signal processing. It is implemented in every radar system to discriminate targets against clutter and noise. The target detection performance is expressed in terms of detection probability and false alarm probability, which are the main figures of merit of a radar. Thus, every improvement in this field should be examined carefully.

In this paper, a peak detector based on a neural network is designed and evaluated. The basic idea of the detector is including conventionally unused information in order to differ between signal and noise peaks in the frequency domain by identifying known signal peak shapes.

II. Signal and Noise Model

Various popular radar continuous wave techniques such as mono-frequency CW, FSK or FMCW radar [1] require the detection of single spectral lines in the frequency domain. Hence, given the presence of I targets, the complex baseband signal can be modelled as

$$\tilde{s}(n) = \sum_{i=1}^{I} a_i \exp\left(j 2\pi f_i T_{\mathrm{s}} n\right) \tag{1}$$

with amplitudes a_i, frequencies f_i and the sampling rate

$$f_{\mathrm{s}} = \frac{1}{T_{\mathrm{s}}}. \tag{2}$$

E.g., in a single target scenario with CW or FSK modulation the baseband signal is described with $I = 1$ and f_1 represents the Doppler frequency of the target. Noise is added using a complex additive white Gaussian noise (AWGN) model, yielding

$$s(n) = \tilde{s}(n) + \mathrm{AWGN}(n, \mu, \sigma). \tag{3}$$

We assume the noise to be zero-mean ($\mu = 0$) with a constant standard deviation σ. The signal-to-noise ratios (SNRs) of each frequency f_i in (1) is given by

$$\mathrm{SNR}_i = \frac{a_i^2}{\sigma^2}. \tag{4}$$

The respresentation of (3) in the frequency domain is generated using a normalized N-point Fast Fourier Transformation (FFT)

$$\tilde{S}(k) = \frac{1}{N} \mathrm{FFT}_{\tilde{N}} \left\{ s(n) w(n) \right\} \tag{5}$$

with the window function $w(n)$. Zero-padding allows interpolating in the frequency domain in order to increase the spectral resolution by attaching zeros to the original N signal samples and calculate the transformation (5) over an enlarged signal with $\tilde{N}$ samples. The zero-padding factor shall be defined as

$$a_{\mathrm{zp}} = \frac{\tilde{N}}{N}. \tag{6}$$

III. Non-Coherent Peak Detection

Two peak detection techniques are chosen to compare the neural network based detector, which is proposed in section IV, with existent detection algorithms. The threshold detector [2] and OS-CFAR detector [3] are commonly used methods for peak detection but do not exploit peak shape information.

Samples for peak detection are taken from the amplitude spectrum

$$S(k) = \mathrm{abs}(\tilde{S}(k)). \tag{7}$$

The subset $G(k) \subset S(k)$ with an odd width of M samples is obtained by applying a sliding window to $S(k)$. The indices of $G(k)$ range from $-\left(\frac{M}{2} - 1\right)$ to $\frac{M}{2} - 1$ and $G(0)$ is called the cell-under-test (CUT). Fig. 1 gives an example spectrum.

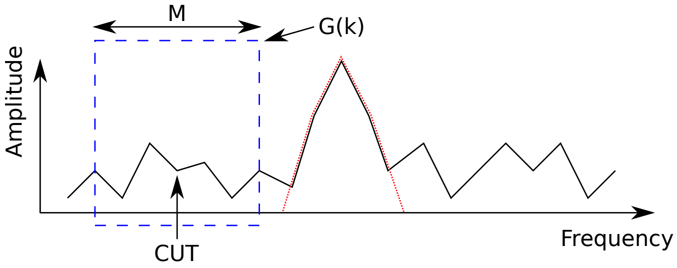

Fig. 1. Depicted is an example noisy amplitude spectrum with the sliding window (blue, dashed) and the peak with the expected peak shape (red, dotted) based on the window function used in the FFT. The detector gives a decision for the CUT using only samples in the sliding window.

A. Threshold Detector

The detection rule of the threshold detector can be written as

$$G(0) \underset{H_1}{\overset{H_0}{\lessgtr}} \gamma_{\mathrm{T}}. \tag{8}$$

H_1 denotes the hypothesis that a peak is existent and H_0 is the Nullhypothesis, which declares that only noise is present in the CUT.

The Neyman-Pearson lemma [2] declares that this detector is the most powerful one using only information from the CUT if a priori information of the noise power is given. Therefore the threshold detector is used as the main reference detector.

Obviously for this detector the sliding window width $M_{\mathrm{T}} = 1$.

B. OS-CFAR Detector

The OS-CFAR detector takes M_{O} samples from $G(k)$ and sorts them in ascending order:

$$\begin{aligned} &G_{\mathrm{O}}(\tilde{k}) = \mathrm{sort}\,\{G(k)\} \text{ with} \\ &k \in [-M_{\mathrm{O},1}, -M_{\mathrm{O},2} - 1] \wedge [M_{\mathrm{O},2} + 1, M_{\mathrm{O},1}] \end{aligned} \tag{9}$$

Fig. 2 explains the sample selection from $G(k)$ with $M_{\mathrm{O},1}$ and $M_{\mathrm{O},2}$ in detail. The detection rule is given by

$$\frac{G(0)}{G_{\mathrm{O}}(\Lambda)} \underset{H_1}{\overset{H_0}{\lessgtr}} \gamma_{\mathrm{O}}. \tag{10}$$

$\Lambda = \mathrm{round}\,\{3M_{\mathrm{O}}/4\}$ is a well suited choice for practical applications and a typical value for M_{O} is given with $24 \ldots 32$ or greater [3]. M_{O} scales linearly with the used zero-padding factor a_{zp}.

This detection algorithm is used widely in radar because the noise power has not to be known, which is a great advantage in practice. Furthermore, the algorithm is robust against time-varying noise floors and clutter.

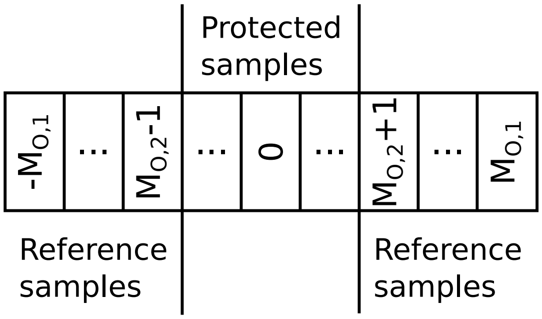

Fig. 2. To select M_O samples in $G_O(k)$, $M_{\mathrm{O},1} = \frac{M_{\mathrm{O}}}{2} + M_{\mathrm{O},2}$ has to hold. Samples within $k \in [-M_{O,2}, M_{O,2}]$ are called protected samples, which should exclude the peak shape around $G(0)$.

IV. Peak Detection with a Neural Network

The used neural network implementation is called NeuroBayes, which is developed by Blue Yonder [4] (formerly phi-t). Originally this software was developed for analyses in the field of high energy physics and is widely used at CERN and other physics institutions. The following sections describe the internal processing of the employed NeuroBayes configuration as described in the documentation [5]. This software is chosen because of the known algorithmic robustness and easy usability, e.g., it is not possible to over-train the network.

A. Peak Detection

The peak detection with the neural network is done by feeding M_{N} samples from the sliding window $G(k)$ to the network. The selected samples from $G(k)$ cover the entire peak shape around the CUT. The peak shape is defined by the used window function $w(n)$ in Eq. (5). The network output O_{N} is given by a real scalar value in the range of $[-1, 1]$. The result of the transformation

$$P_{H_1} = \frac{O_{\mathrm{N}} + 1}{2} \tag{11}$$

represents the probability that a peak is shown to the network and therefore directly provides useful information, which can be used as a figure for the reliability of the detection. The detection rule is a threshold decision and can be written as

$$O_{\mathrm{N}} \underset{H_1}{\overset{H_0}{\lessgtr}} \gamma_{\mathrm{N}}. \tag{12}$$

B. Neural Network

The used neural network provided by NeuroBayes is a three layer feed-forward network. Fig. 3 depicts the structure of the network. The number of input nodes is set to the sliding window size M_{N} but can be reduced if the training shows no significance for specific input bins. The output layer consists only of a single node because we want to perform a binary classification. The number of hidden nodes is set to $M_{\mathrm{N}} + 1$. Although the number of hidden nodes is optimizable, the actual number is not very performance-critical because

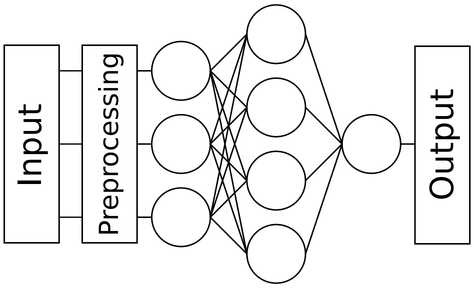

Fig. 3. The first layer of input nodes (circles) is fed with the preprocessed input data. These are followed by the so-called hidden layer. The network output is taken from the output layer, which is connected to the hidden layer.

the training algorithm is capable of pruning. This means that unused connections between the nodes are removed if they are not significant during the training. Nevertheless, a too small number makes pruning impossible whereas a too large number slows down the training.

C. Preprocessing

The employed default NeuroBayes preprocessing for binary classification implies that all input variables are normalised, decorrelated and then transformed to a Gaussian distribution before they are given to the neural network.

The decorrelation of the input variables performs a transformation similar to creating an orthogonal basis from arbitrary chosen vectors. This yields a transformation rule for the input to a variable space in which no linear dependencies are existent.

Hence, the preprocessing transforms the given input to input variables which are more suitable for training due to an equalized value range and distribution and the elimination of linear dependencies.

D. Training

The used training procedure is a so-called supervised training and is based on a Broyden-Fletcher-Goldfarb-Shanno [6] back-propagation algorithm. A weight is assigned to each connection between nodes in Fig. 3, which makes them more or less important to the network output. Given an input of a known type, the back-propagation algorithm compares the output of the network with the desired output and adjusts the weights of the connections iteratively.

Although the training has to be done on samples with the later expected noise power, it is a great advantage in practice that the noise power of the shown samples has not to be known because the training is doing implicitly a noise power estimation by adjusting the network on the shown value range.

V. Simulation Analysis

A. Analysis Procedure

The different detectors are evaluated by running the algorithms N times on a noisy spectrum with a single target peak in the middle. Fig. 4 shows an example for the count of positive detections over the amplitude spectrum bins and explains the calculation of the detection rate and the false alarm rate.

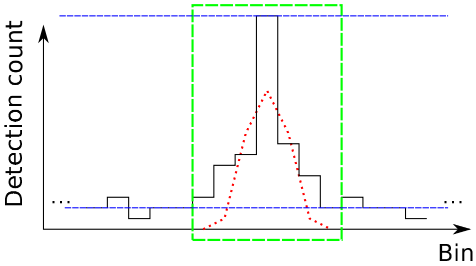

Fig. 4. The evaluation of N spectra with the red dotted target peak in the middle provides a count of positive detections (solid black line) by a specific detector for each bin. The detection rate is given by the maximum count in the green dashed box normalized to N. This is indicated by the upper blue dashed line. The false alarm rate is calculated using the mean of all detections outside this box normalized to N as shown by the lower blue dashed line. The size of the green dashed box is chosen in regard to the width of the specific peak shape and the sliding window so that no information of the target peak is included in the false alarm rate calculation.

To model the leakage of the signal energy across adjacent frequency bins, a jitter is applied to the frequencies f_i in Eq. (1). This means that whenever a signal is generated the frequency is randomly chosen within the range $\left[f_i - \frac{f_\mathrm{S}}{2\tilde{N}}, f_i + \frac{f_\mathrm{S}}{2\tilde{N}}\right]$, which covers all positions between the bins.

The analysis refers to the receiver operating characteristics (ROC) as the plot of the detection rate over the false alarm rate at a given SNR. A ROC diagram is created by evaluating N spectra for various thresholds γ.

B. Signal and Detector Parameters

The threshold detector is set up as described in section III because no free parameters are to be determined. The noise power is supposed to be known. The OS-CFAR algorithm is implemented with 20 protected samples and 129 reference samples. The relative threshold is set to $G(\Lambda)$ with $\Lambda = \mathrm{round}\left\{3M_\mathrm{O}/4\right\} = 97$. For the neural network based detector, a sliding window width of $M_\mathrm{O} = 21$ is used. Fig. 5 shows the averaged spectrum following Eq. (5) with a Hamming window [7], $f_\mathrm{s} = 1024$, $N = 1024$, and $a_\mathrm{zp} = 4$. It illustrates the suitability of the parameter choices for signal and detectors. These parameters apply to all presented simulations if not stated otherwise.

C. Training Analysis

NeuroBayes provides a training analysis toolchain using ROOT [8]. The separation of signal and background (noise only) events by the neural network can be evaluated and is shown in Fig. 6. It provides all information about detection rate and false alarm rate at a given neural network output threshold γ_N and can be transformed into a ROC plot.

The detection rate and false alarm rate estimations gathered from the neural network training analysis in Fig. 6 match well with the simulations based on the trained network, which are depicted in Fig. 7. Hence, any false alarm rate and

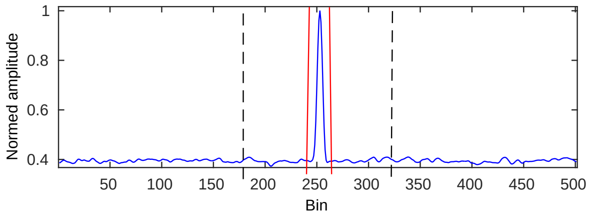

Fig. 5. This spectrum is averaged over 10^3 realizations. The length of the window in-between the vertical red lines is 21 samples and equals the neural network sliding window width M_N and the number of protected samples of the OS-CFAR detector. The black dotted vertical lines indicates the OS-CFAR sliding window size with 149 samples.

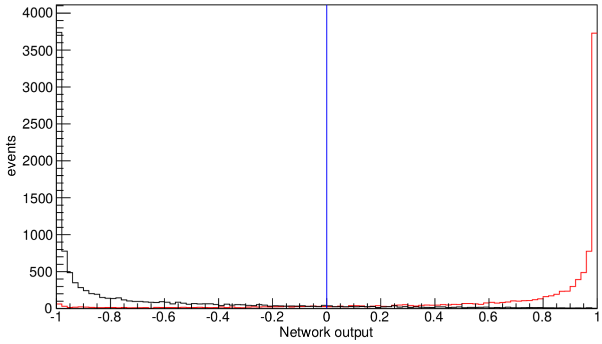

Fig. 6. Shown is the histogram of the signal (red) and background (black) events used in the training over the neural network output. This data equals a receiver operating characteristic if an arbitrary number of thresholds $\gamma_\mathrm{N} \in [-1, 1]$ is evaluated.

the associated threshold γ_N are given directly by the neural network training analysis.

The NeuroBayes training analysis provides also the evaluation of the significance of the input bins. Fig. 8 shows exactly which bins are significant due the correlation to the output and that the sliding window width $M_\mathrm{N} = 21$ can be optimized to $\tilde{M}_\mathrm{N} = 13$ without a performance loss.

Fig. 9 evaluates the influence of the SNR of the training data on the performance on data with a constant SNR. It shows that no knowledge of the training data SNR is needed, which simplifies training on data from hardware significantly. Because of this, the SNR of the training data is not specified in following evaluations.

D. Detection Performance

Fig. 10 compares the discussed detectors in a ROC diagram. It can be seen that the neural network based detector outperforms the reference detectors and reveals a highly improved detection rate especially at low false alarm rates. The detection rate in regard to a variable SNR and a constant false alarm rate as depicted in Fig. 11 shows an improved performance at all SNRs for the neural network based detector. Furthermore, this detector performs superiorly due to a better peak localisation because of the slimmer detection probability function shown in Fig. 12. Fig. 10 to Fig. 12 show the expected performance loss of the OS-CFAR detector in regard to the threshold detector [3].

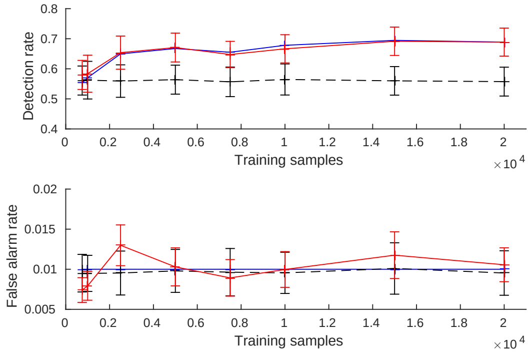

Fig. 7. Plotted are detection rates and false alarm rates estimated by the NeuroBayes training analysis (blue line without errorbars), the test of these values (continuous red line with errorbars) and the reference threshold detector performance (dashed black line with errorbars) at a constant false alarm rate of 10^{-2} and $8\,\mathrm{dB}$ SNR. The test data is evaluated with $M_\mathrm{N} = 21$ and 10^2 iterations per value. Plotted are the mean and standard deviation of 100 tests.

VI. Conclusion

The proposed neural network based peak detector exploits peak shape information successfully with a superior performance in all evaluated SNR scenarios. Furthermore, it is shown that the detector has an improved peak localization due to a slim detection probability distribution around the target peak and using a minimal number of samples from the amplitude spectrum including only the width of the peak shape. The training of the used neural network provided by NeuroBayes is simple because it is shown that no knowledge about the SNR and the constant noise power of the training data is needed. At a false alarm rate of 10^{-2} the neural network based detector provides a performance increase of up to $10\,\%$ in regard to the threshold detector and up to $16\,\%$ compared with the OS-CFAR detector. An additional benefit of the neural network based detector is that we directly obtain information on the reliability of the detection.

Moreover, the detection method using neural networks is easily extendible by giving additional input parameters to the network, e.g., abstract properties such as mean or variance of a subset of samples. Whether including such information improves the performance has to be shown by further studies.

References

[1] Kronauge, Matthias "Waveform Design for Continuous Wave Radars" Cuvillier, 2014

[2] Richards, Mark A. Fundamentals of radar signal processing. Tata McGraw-Hill Education, 2005.

[3] Rohling, Hermann. "Radar CFAR thresholding in clutter and multiple target situations." Aerospace and Electronic Systems, IEEE Transactions on 4 (1983): 608-621.

[4] Blue Yonder, www.blue-yonder.de

[5] NeuroBayes User's Guide, neurobayes.phi-t.de

[6] Byrd, Richard H., et al. "A limited memory algorithm for bound constrained optimization." SIAM Journal on Scientific Computing 16.5 (1995): 1190-1208.

[7] Harris, Fredric J. "On the use of windows for harmonic analysis with the discrete Fourier transform." Proceedings of the IEEE 66.1 (1978): 51-83.

[8] ROOT Data Analysis Framework, root.cern.ch

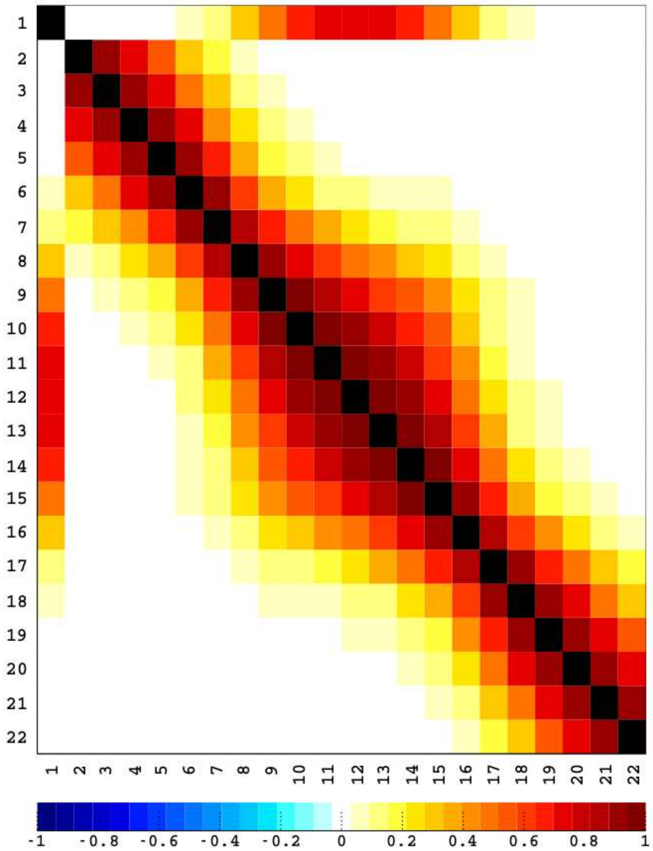

Fig. 8. The matrix shows the correlation of the output of the neural network (bin 1) and the neural network input (bins $2 \ldots 22$). Evaluated is a training with 10^4 training samples on data with 8 dB SNR and a Hamming window. The plot shows that only input variables 6 to 18 are significant and the sliding window width $M_\mathrm{N} = 21$ can be reduced.

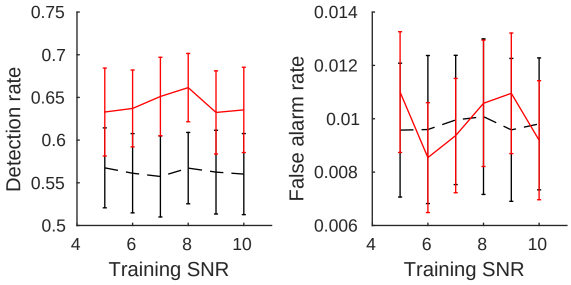

Fig. 9. The detection rates are evaluated at 8 dB SNR and a false alarm rate of 10^{-2} with a training at different SNRs. A single detection rate or false alarm rate is measured over 10^3 iterations. Shown are the mean and standard deviation of 100 values. The continuous red line represents the neural network based detector and the dashed black line is the reference threshold detector.

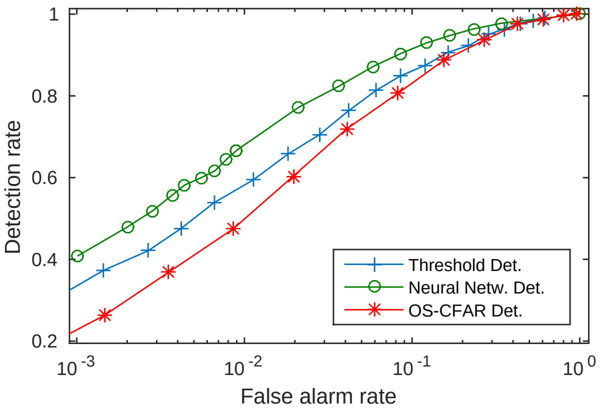

Fig. 10. The ROC diagram is generated at 8 dB SNR, 10^3 iterations per point and a neural network training on 10^4 samples.

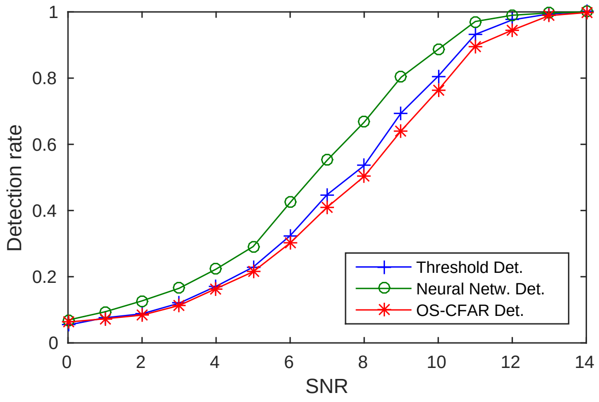

Fig. 11. The detection rates are evaluated with 10^3 iterations per point and the detector thresholds for a false alarm rate of $(0.98 \pm 0.09) \cdot 10^{-2}$ for the threshold detector, $(1.04 \pm 0.06) \cdot 10^{-2}$ for the neural network based detector and $(1.29 \pm 0.09) \cdot 10^{-2}$ for the OS-CFAR detector. The neural network is trained on 10^4 samples.

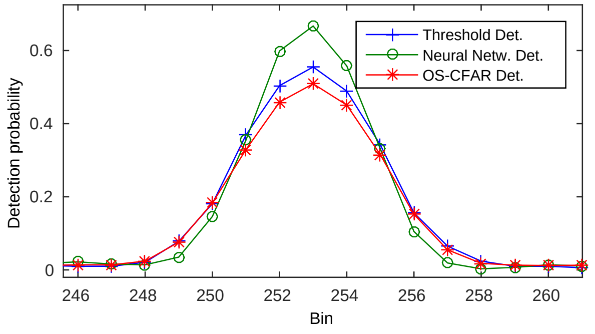

Fig. 12. Shown is the detection count normalized to $N = 10^3$ iterations with a false alarm rate of 10^{-2} at 8 dB SNR and a neural network trained on 10^4 samples.