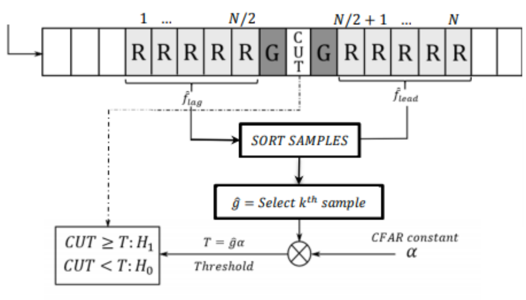
2D OS CFAR Initial results.

2D OS CFAR was developed.

The following presents the initial simulation results of the developed 2D OS(ordered-statistic) CFAR.

2D OS CFAR Theory

As elaborated in the previous report, Rohling[1] proposed the Order Statistic CFAR detector (OS-CFAR), which was designed to overcome the issues, such as clutter transition, self-masking target, and mutual masking targets. This OS-CFAR detector processes in a way that is quite different than the CFAR detectors. It estimates the average background noise by a signal sample from the reference cells using order statistics processing. The OS-CFAR detector ranks the samples in the reference cells according to increasing power. It then selects the k-th sample based on the CFAR statistics to estimate the average threshold of the environment. The architecture of the 1D OS-CFAR is shown below:



Ordered Statistics CFAR processor

Figure 1

To reach near to the minimum of the CFAR loss of the OS-CFAR, as a rule of thumb, the $k\ th$ sample should be chose at $3N/4$ order which is well fit for practical applications [1].
In OS-CFAR, the guard cells are not necessary since ranking the samples push the high values to the end of the reference cells. This means that the self-masking target is not an issue in OS-CFAR detector. Therefore, it is possible to apply the OS-CFAR without guard cells.

OS CFAR Threshold Factor

As in any CFARs, the expression of the Threshold Factor has to be derived.

The exact derivation can be seen in Appendix A, or in the book – “Fundamentals of Radar Signal Processing by Mark_A._Richards”[2] – Chapter 7.

For a required **Pfa**, **N** and **k**, the Threshold Factor(**alpha_os**) is to be calculated using the expression below:

$$\overline{P}_{FA} = k\frac{N!}{k!(N-k)!}\frac{(k-1)!(\alpha_{OS}+N-k)!}{(\alpha_{OS}+N)!} = \frac{N!(\alpha_{OS}+N-k)!}{(N-k)!(\alpha_{OS}+N)!} \qquad (\alpha_{OS}\ \text{integer})$$

Equation 1

Matlab function was written for the calculation purposes(f(Pfa,N,k)) and compared to the theory(left), as can be seen below:

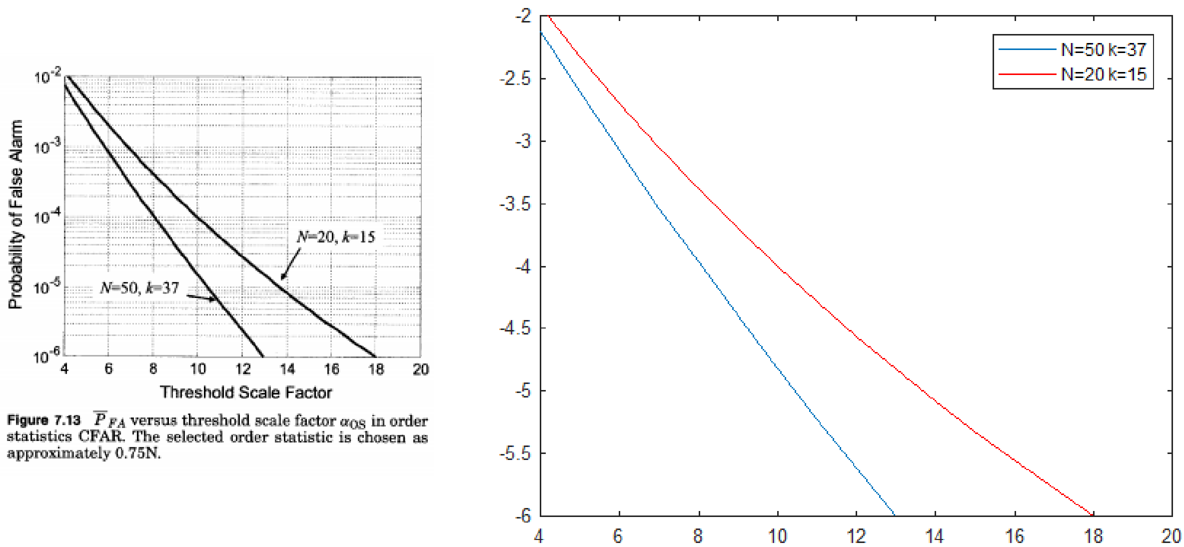


Figure 2

Simulation with Noise only.

The Matlab code of the 2D OS CFAR was developed and tested.

The code can be found in[3]

First, the parameters of the CFAR are set:

```
N = 141;%Raws
M = 151;%Colomn
detector.ProbabilityFalseAlarm = 10e-4;%Probablity of false alarm;
detector.GuardBandSize = [0 0];
detector.TrainingBandSize = [6 4];
```

Then a N-by-M image containing random complex data is being created.

Next, squaring the data to simulate a square-law detector.

Then, the indexes of all the CUT indexes skipping the edges, are being generated(done once).

Both, can be seen below:

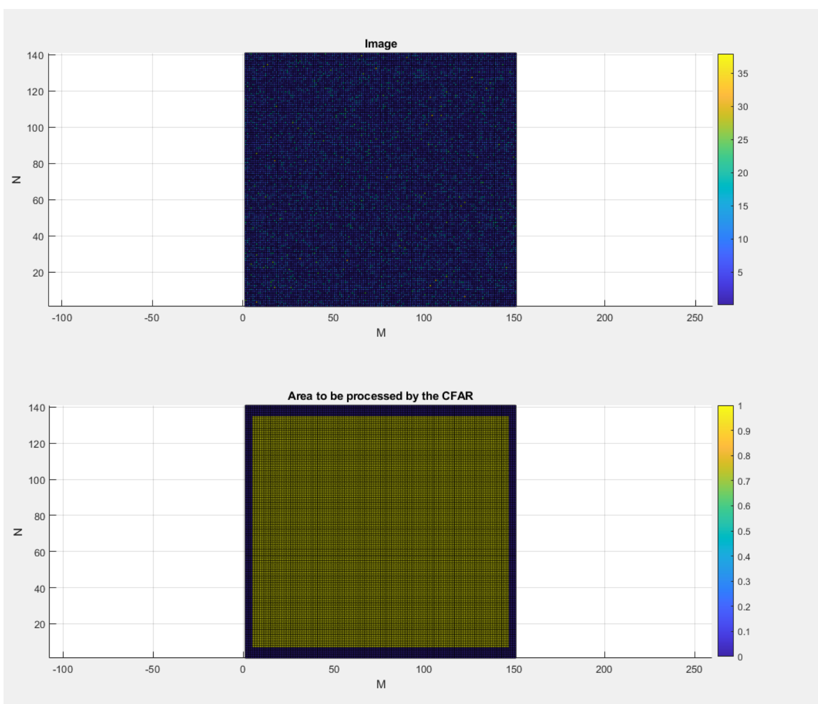


Figure 3

First, noise matrix was generated, as can be seen below:

For the TrainingBandSize = [6 4] and Pfa = 10^-4, the Threshold factor was calculated and found to be `ThFac` = 7:

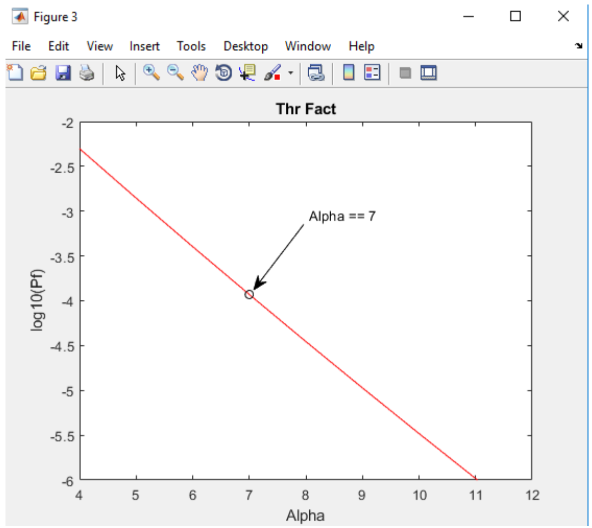


Figure 4

The main loop, calculates the indexes of the CUT(marked red below), sorts the values, choses the one at index Rank(calculated once), calculates the threshold according to:

```
th(k) = noisePowEst * ThFac;
noise(k) = noisePowEst;
```

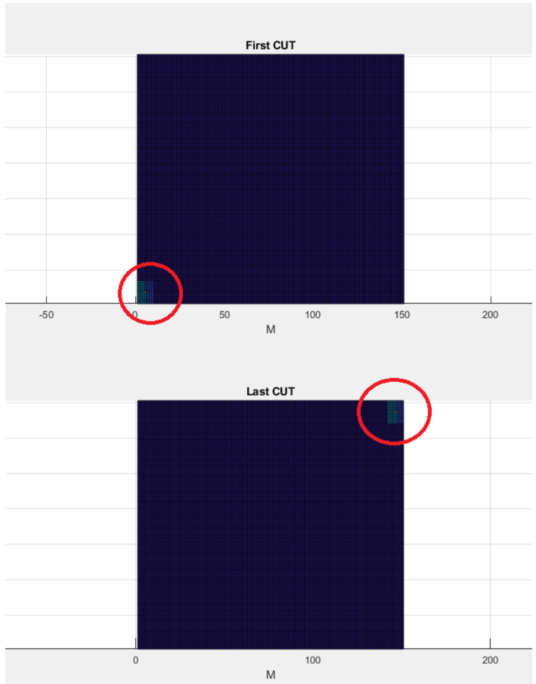


Figure 5

Applying the OS CFAR with the parameters as listed above, yielded the following:

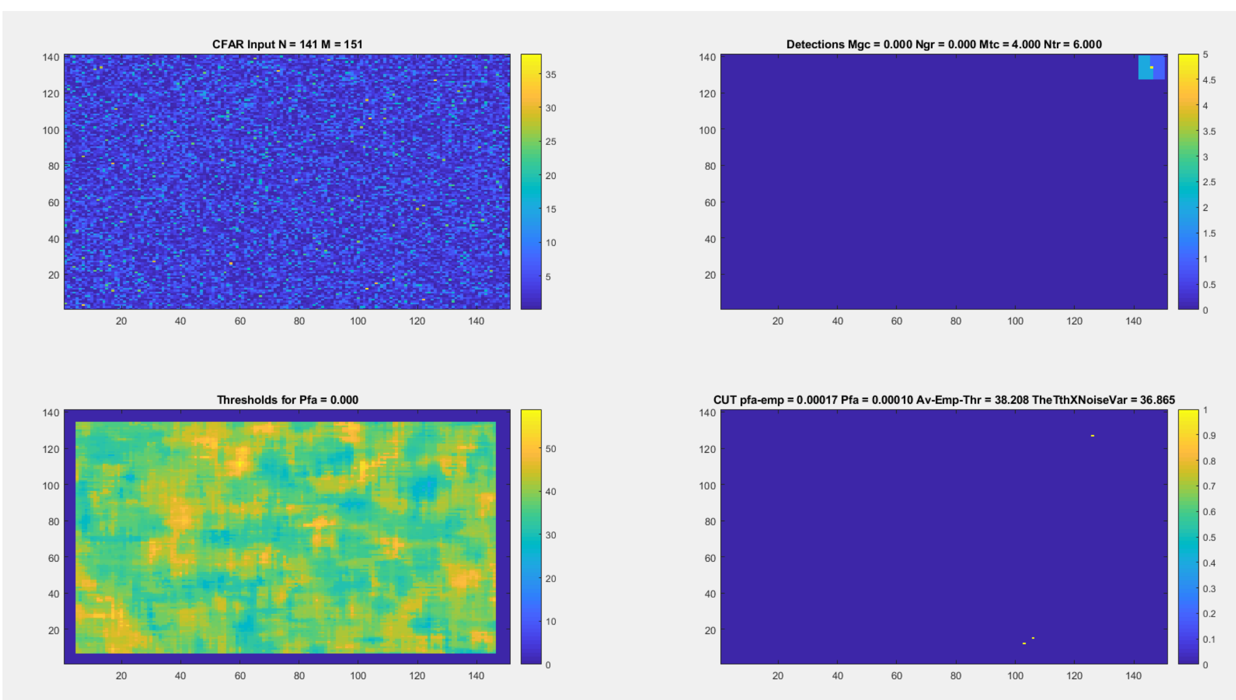


Figure 6

The theoretical and the empirical parameters were found to be similar(Pfa(0.00017~=0.00010) and Thr(38.208~=36.865)).

Simulation for Noise+Targets

CFAR input(200X1024) with 3 targets was generate using Matlab function – **helperRangeDoppler**.

The image can be seen below.

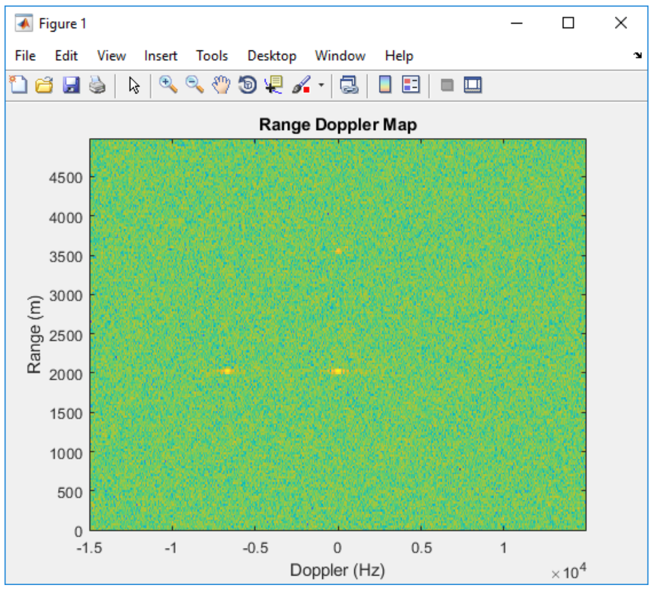


Figure 7

3D representation below:

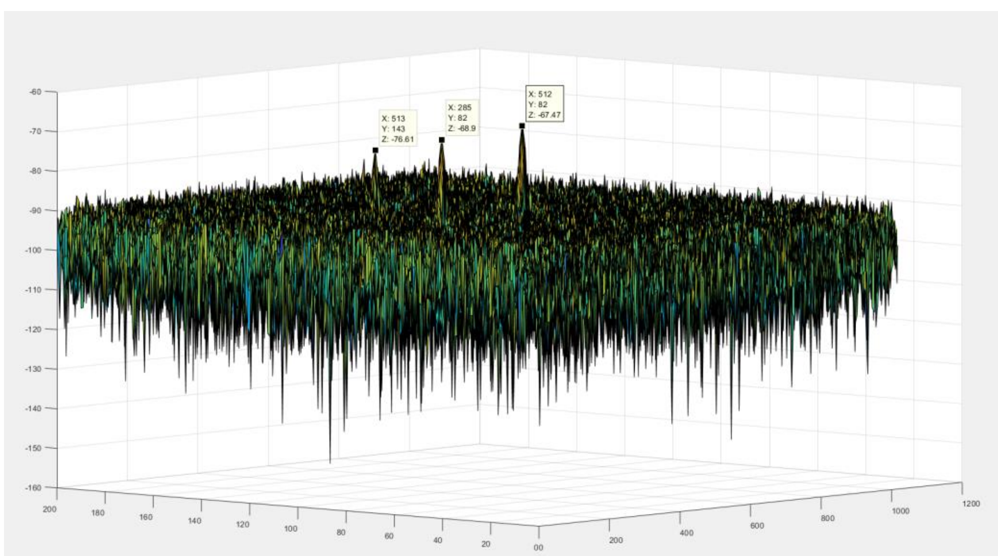


Figure 8

The detector parameters set to:

TrainingBandSize: [6 4]

Method: 'OS'

ProbabilityFalseAlarm: 1.0000e-05

The threshold factor calculated(alpha == 9), and the detector applied, yielding the following:

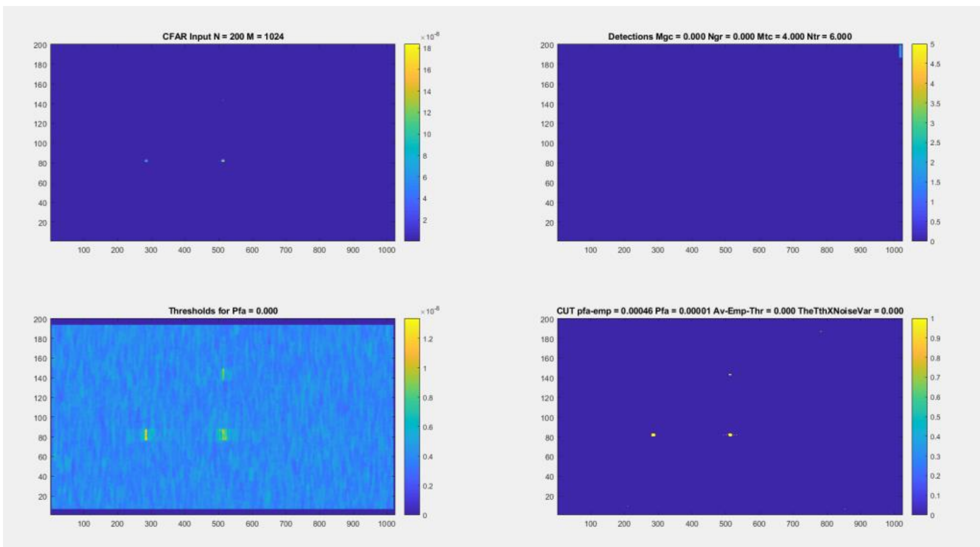


Figure 9

The three detected targets can be spotted on the right bottom plot.

Further tuning is required.

Code can be found in[3].

Next:

1. Clustering Methods overview.

References:

[1] H. Rohling, “Radar CFAR thresholding in clutter and multiple target situations”, IEEE Trans. on Aerospace and electronic systems Vol. AES-19, NO. 4, pp. 608- 621, July 1983

[2] https://drive.google.com/file/d/1sl0I8DDBF30IwNpBbaiVEb85tb3AwvhQ/view?usp=sharing

[3] https://drive.google.com/file/d/14Q4ZIcNCmppMLPikXeepotqvGWtS4E0L/view?usp=sharing

Appendix A:

An alternative to cell-averaging CFAR is the class of *rank-based* or *order statistic* CFARs (OS CFAR). Proposed primarily for combating masking degradations, OS CFAR retains the one-dimensional or two-dimensional sliding window structure of CA CFAR, including guard cells if desired, but does away entirely with averaging of the reference window contents to explicitly estimate the interference level. Instead, OS CFAR rank orders the reference window data samples $\{x_1, x_2, \ldots, x_N\}$ to form a new sequence in ascending numerical order, denoted by $\{x_{(1)}, x_{(2)}, \ldots, x_{(N)}\}$. The kth element of the ordered list is called the kth *order statistic*. For example, the first order statistic is the minimum, the Nth order statistic is the maximum, and the $(N/2)$th order statistic is the median of the data $\{x_1, x_2, \ldots, x_N\}$. In OS CFAR, the kth order statistic is selected as representative of the interference level and a threshold is set as a multiple of this value

$$\hat{T} = \alpha_{\mathrm{OS}} x_{(k)} \tag{7.40}$$

Note that the interference is thus estimated from only one actual data sample, instead of an average of all of the data samples. Nonetheless, the threshold in fact depends on all of the data, since all of the samples are required to determine which will be the kth largest.

It will be shown that this algorithm is in fact CFAR (i.e., does not depend on the interference power β^2), and the threshold multiplier required to achieve a specified $\overline{P}_{FA}$ will be determined. The analysis follows (Levanon 1988). To simplify the notation, consider the square-law detected output x_i normalized to its mean, $y_i = x_i/\beta^2$; this will have an exponential pdf with unit mean. The rank-ordered set of reference samples $\{y_i\}$ are denoted by $\{y_{(i)}\}$. For a given threshold T, the probability of false alarm will be

$$P_{FA}(T) = \int_T^{+\infty} e^{-y}\, dy = e^{-T} \tag{7.41}$$

The average P_{FA} will be computed as

$$\overline{P}_{FA} = \int_0^{+\infty} P_{FA}(T)\, p_T(T)\, dT \tag{7.42}$$

where $p_T(T)$ is the pdf of the threshold. Because T is proportional to the kth-ranked reference sample $y_{(k)}$, it is necessary to find the pdf of $y_{(k)}$. This tedious derivation can be found in the book by Levanon (1988); in terms of the probability density function $p_{y_i}(y)$ and corresponding *cumulative distribution function* (CDF) $P_{y_i}(y)$, the result is

$$p_{y_{(k)}}(y) = k \binom{N}{k} P_{y_i}^{k-1}(y) \left[1 - P_{y_i}(y)\right]^{N-k} p_{y_i}(y) \tag{7.43}$$

For i.i.d. Gaussian I/Q noise, the pdf and CDF of a single normalized reference sample y_i are

$$p_{y_{(k)}}(y) = e^{-y}$$

$$P_{y_{(k)}}(y) = \int_0^y p_{y_{(k)}}(y')\,dy' = 1 - e^{-y} \tag{7.44}$$

Using Eq. (7.44) in Eq. (7.43) gives the pdf of the kth ranked sample

$$p_{y_{(k)}}(y) = k\binom{N}{k}[e^{-y}]^{N-k+1}[1-e^{-y}]^{k-1} \tag{7.45}$$

The threshold is $\hat{T} = \alpha_{\mathrm{OS}}\, y_{(k)}$, so the pdf of $\hat{T}$ is $p_{\hat{T}}(\hat{T}) = (1/\alpha_{\mathrm{OS}})p_{y_{(k)}}(\hat{T}/\alpha_{\mathrm{OS}})$

$$p_{\hat{T}}(\hat{T}) = \frac{k}{\alpha_{\mathrm{OS}}}\binom{N}{k}\left[e^{-\hat{T}/\alpha_{\mathrm{OS}}}\right]^{N-k+1}\left[1-e^{-\hat{T}/\alpha_{\mathrm{OS}}}\right]^{k-1} \tag{7.46}$$

Inserting this result into Eq. (7.42) gives

$$\begin{aligned}\overline{P}_{FA} &= \int_0^{+\infty} e^{-\hat{T}}\,\frac{k}{\alpha_{\mathrm{OS}}}\binom{N}{k}\left[e^{-\hat{T}/\alpha_{\mathrm{OS}}}\right]^{N-k+1}\left[1-e^{-\hat{T}/\alpha_{\mathrm{OS}}}\right]^{k-1} d\hat{T} \\ &= \frac{k}{\alpha_{\mathrm{OS}}}\binom{N}{k}\int_0^{+\infty} e^{-(\alpha_{\mathrm{OS}}+N-k+1)\,\hat{T}/\alpha_{\mathrm{OS}}}\left[1-e^{-\hat{T}/\alpha_{\mathrm{OS}}}\right]^{k-1} d\hat{T}\end{aligned} \tag{7.47}$$

With the change of variable $T' = \hat{T}/\alpha_{\mathrm{OS}}$ this becomes the slightly more convenient form

$$\overline{P}_{FA} = k\binom{N}{k}\int_0^{+\infty} e^{-(\alpha_{\mathrm{OS}}+N-k+1)T'}\left[1-e^{-T'}\right]^{k-1} dT' \tag{7.48}$$

Utilizing integral 3.312(1) of the book by Gradshteyn and Ryzhik (1980) gives

$$\begin{aligned}\overline{P}_{FA} &= k\binom{N}{k}\mathrm{B}(\alpha_{\mathrm{OS}}+N-k+1, k) \\ &= k\binom{N}{k}\frac{\Gamma(\alpha_{\mathrm{OS}}+N-k+1)\Gamma(k)}{\Gamma(\alpha_{\mathrm{OS}}+N+1)}\end{aligned} \tag{7.49}$$

where $\mathrm{B}(\cdot,\cdot)$ is the beta function and in turn can be expressed in terms of the gamma function $\Gamma(\cdot)$ as shown. For integer arguments, $\Gamma(n) = (n-1)!$ and Eq. (7.49) therefore reduces for integer α_{OS} to

$$\begin{aligned}\overline{P}_{FA} &= k\frac{N!}{k!(N-k)!}\frac{(k-1)!(\alpha_{\mathrm{OS}}+N-k)!}{(\alpha_{\mathrm{OS}}+N)!} \\ &= \frac{N!(\alpha_{\mathrm{OS}}+N-k)!}{(N-k)!(\alpha_{\mathrm{OS}}+N)!} \qquad (\alpha_{\mathrm{OS}}\text{ integer})\end{aligned} \tag{7.50}$$

Figure 7.13 plots $\overline{P}_{FA}$ as a function of α_{OS} for two choices of OS windows, one with $N = 20$ and one with $N = 50$. In the first case, the $k = 15$th order statistic is chosen to set the threshold, while in the second the $k = 37$th order statistic

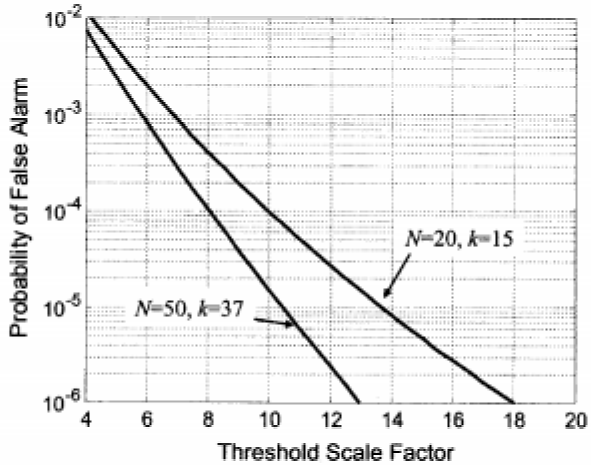


Figure 7.13 $\overline{P}_{FA}$ versus threshold scale factor α_{OS} in order statistics CFAR. The selected order statistic is chosen as approximately 0.75N.