JOHANNES KEPLER
UNIVERSITY LINZ

Submitted by
Christina Katzlberger

Submitted at
Institute of Signal Processing

Supervisor
DI Michael Gerstmair

September 2018

Object Detection with Automotive Radar Sensors using CFAR-Algorithms

Bachelor Thesis
to obtain the academic degree of
Bachelor of Science
in the Bachelor's Program
Elektronik und Informationstechnik

**JOHANNES KEPLER
UNIVERSITY LINZ**
Altenberger Straße 69
4040 Linz, Österreich
www.jku.at
DVR 0093696

Abstract

One of the most important tasks in radar signal processing is to reliably detect objects in the surrounding of the radar sensor. This can be done by comparing the frequency spectrum of the measured signal to a specific detection threshold. Using a constant threshold value may cause a large number of wrong object detections. Thus, so-called constant false alarm rate (CFAR) algorithms are used, which are able to calculate an adaptive threshold value due to the estimated noise floor. Two methods widely used in frequency modulated continuous wave radar (FMCW) systems will be presented, which differ in how the noise estimation is achieved. In the first part of this thesis the principles of both methods are explained and compared in the means of Matlab simulations. Therefore, the signal of interest is directly calculated in the intermediate frequency (IF)-Domain and distorted by additive white Gaussian noise (AWGN). Since the CFAR algorithms are adapted to a specific noise model, the detection performance should become independent of the Gaussian background noise power. In the second part the presented CFAR algorithms are tested based on measured FMCW radar signals, with the aim to detect non-moving objects.

Kurzfassung

Eine der wichtigsten Aufgaben bei der Radarsignalverarbeitung ist die zuverlässige Erkennung von Objekten in der Umgebung des Radarsensors. Dies kann durch Vergleichen des Frequenzspektrums des gemessenen Signals mit einer bestimmten Detektionsschwelle erfolgen. Die Verwendung eines konstanten Schwellenwerts kann zu mehreren falschen Objekterkennungen führen, die aufgrund von Hintergrundrauschen verursacht werden können. Daher werden sogenannte CFAR-Algorithmen (CFAR = Constant False Alarm Rate) verwendet, die aufgrund des geschätzten Rauschens einen adaptiven Schwellenwert berechnen. Zwei weitverbreitete Methoden in der Dauerstrichradartechnik werden in dieser Arbeit vorgestellt, welche die Hintergrundrauschleistung auf unterschiedliche Weise schätzen. Im ersten Teil dieser Arbeit werden die Prinzipien beider Methoden erklärt und mit Hilfe von Matlab Simulationen verglichen. Dafür wurde, das für die Objektdektierung interessante Signal direkt im Zwischenfrequenz (IF=Intermediate frequency)-Bereich berechnet und mit additivem weißem Gaußschem Rauschen (AWGN = additive white Gaussian noise) beaufschlagt. Da die CFAR-Algorithmen an ein bestimmtes Rauschmodell angepasst sind, sollte die Stärke des Hintergrundrauschens keinen Einfluss auf die Detektionsfähigkeit nehmen. Im zweiten Teil werden die vorgestellten CFAR-Algorithmen mit gemessenen FMCW-Radarsignalen getestet, um ruhende Objekte zu erkennen.

Contents

1 Introduction

In automotive applications typically FMCW radars, operating in frequency ranges from 76 GHz to 81 GHz, are used. Advanced Driver Assistance Systems (ADAS) like autonomous emergency breaking, identify imminent collisions and hit the brakes before the driver even starts to react. This technology requires the simultaneous measurement of range, velocity and angle of objects in the surrounding of the radar sensor. Most important is that the intervention of the car is reasonable. The driver would not be satisfied with the innovative technology, if the car stops for no reason. This situation may occur if the radar waves are reflected multiply, for example at corners in a parking garage or at crash barriers next to the streets. These so-called clutter environment should be ignored by the object detection algorithm. On the other hand, the ADAS should be activated when an object is located in front of the FMCW radar sensor. A phenomenon which often occurs, is the so-called object masking. In this case, an object is not detected due to a closely located second object. The issues of object masking or clutter, additionally to AWGN, are stated in this thesis. If an object is detected in a pure noise scenario, it is called a false alarm. Constant false alarm rate (CFAR) algorithms are forced to reach a specified false alarm rate, meaning the number of false alarms that are accepted in a certain data range. The threshold holds the value, which a certain range cell has to exceed to be identified as an object. In CFAR algorithms this threshold is adapted according to background noises.

In the first part of this thesis, the FMCW radar principle is explained. In addition, it is shown how the measured data has to be processed, in order to obtain the distances of objects in the vicinity of the radar sensor. Based on a suitable noise model for the radar channel, two algorithms are presented which estimate the noise floor. The idea of these algorithms is to specify a threshold according to neighbouring range cells of the cell of interest, the so-called reference window. With the use of the *sliding window*-technique, it is possible to calculate an adaptive threshold for each range cell based on the present noise. As a result, the decision procedure becomes independent of the surrounding noise power for the underlying noise model. The first algorithm, referred to as cell averaging (CA)-CFAR algorithm in the literature, estimates the surrounding noise power by averaging all reference cells. Therefore, the changes in probability density functions due to the averaging process are examined, in order to obtain a valid threshold calculation for object detection. The second algorithm is termed ordered-statistic (OS)-CFAR, and uses the value out of a certain cell out of the reference window. For a proper noise estimation, the values in the reference window have to be sorted in ascending order according to their magnitude.

Finally, the mentioned algorithms are tested on measured radar data and compared to MATLAB simulations. Conclusively, advantages and disadvantages of the presented algorithms in automotive applications are pointed out.

2 Range estimation using FMCW Radar

In automotive applications typically frequency modulated continuous wave (FMCW) radars are used. As shown in Figure 2.1, the frequency of the transmitted signal is changed linearly over the defined bandwidth B during the measurement time T_{CH}. This signal is radiated into the environment via the transmitting antenna and reflected by the objects in the surroundings of the sensor. The received signal can therefore be modeled as a damped and delayed version of the transmit signal. In the next step, the transmitted and received signals are mixed and low-pass filtered, leading to the intermediate frequency (IF) signal. This signal contains the frequency difference of the multiplied signals, which is directly proportional to the distance of the object and is referred to as beat frequency f_{B} in radar literature. In further consequence, f_{B} has to be estimated in order to determine the object distance.

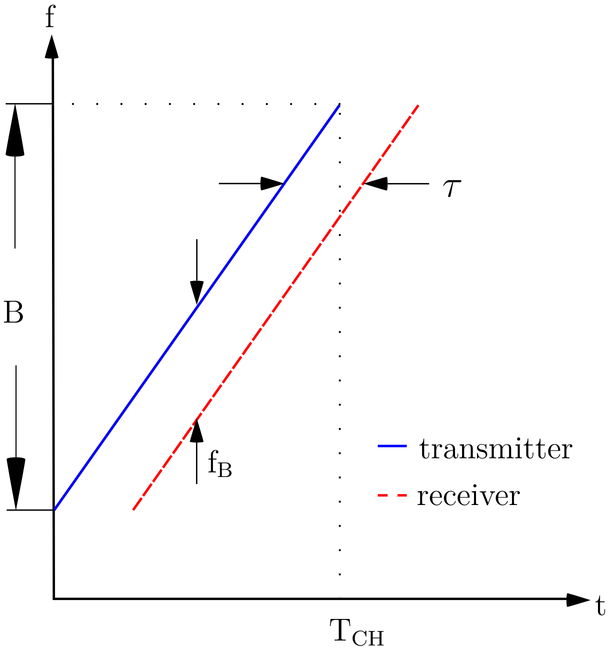


Figure 2.1: Principle of the FMCW radar

The transmit signal (TX) is defined as chirp. An up-chirp is a harmonic signal which increases frequency linearly with time and is defined by

$$s_{TX}(t) = A \cdot cos(2\pi f_0 t + \pi k t^2 + \phi_0), \tag{2.1}$$

where k describes the slope of the chirp and is calculated by $k = B/T_{\mathrm{CH}}$. The parameters f_0 and ϕ_0 are the starting frequency and a constant phase term, respectively. Considering, that only one object is present, the received signal (RX)

$$s_{RX}(t) = A\alpha \cdot cos(2\pi f_0(t-\tau) + \pi k(t-\tau)^2 + \phi_0), \tag{2.2}$$

is also an up-chirp signal with the time delay τ, typically denoted as round trip delay time (RTDT). Furthermore, α is a damping factor due to path losses. The time delay is caused by the distance to the object d and is determined by

$$\tau = \frac{2d}{c_0}, \tag{2.3}$$

where c_0 equals the speed of light. The IF signal is a harmonic signal which carries the information of the object distance and is the result of mixing the TX with the RX signal

$$s_{\mathrm{IF}}{}'(t) = A \cdot cos(\underbrace{2\pi f_0 t + \pi k t^2 + \phi_0}_{\mathrm{x}}) \cdot A\alpha \cdot cos(\underbrace{2\pi f_0(t-\tau) + \pi k(t-\tau)^2 + \phi_0}_{\mathrm{y}}). \tag{2.4}$$

Using the trigonometric identity

$$cos(x) \cdot cos(y) = \frac{1}{2}\left(cos(x-y) + cos(x+y)\right), \tag{2.5}$$

(2.4) results in

$$\begin{aligned} s_{\mathrm{IF}}{}'(t) = & \frac{A^2\alpha}{2}\Big(cos(2\pi f_0\tau + 2\pi k t\tau - \pi k\tau^2) \\ & + cos(4\pi f_0 t + 2\pi k t^2 + 2\phi_0 - 2\pi f_0\tau - 2\pi k\tau + \pi k\tau^2)\Big). \end{aligned} \tag{2.6}$$

As it can be seen from (2.6), the IF signal contains the frequency difference as well as the sum of the involved signal. In order to get rid of the undesired high frequency components $s_{\mathrm{IF}}{}'(t)$ is low-pass filtered which leads to

$$s_{\mathrm{IF}}(t) = s_{\mathrm{IF}}{}'(t) * h_{LP} = \frac{A^2\alpha}{2} cos(2\pi f_0\tau + 2\pi \underbrace{k\tau}_{f_B} t - \pi k\tau^2). \tag{2.7}$$

The term

$$f_{\mathrm{B}} = k\tau = \frac{B}{T_{\mathrm{CH}}}\frac{2d}{c_0} \tag{2.8}$$

is called beat frequency and is directly proportional to the distance between the object and the radar sensor. For a single object reflection, f_B is the frequency of the harmonic IF signal.

A frequency spectrum shows the amplitudes of each frequency, logarithmically scaled against frequency and is obtained by the Fourier transformation. The peak in the frequency spectrum, shown in Figure 2.2, has to be detected in order to get f_B and the distance

$$d = f_{\mathrm{B}} \frac{c_0}{2k} \tag{2.9}$$

of a non-moving object.

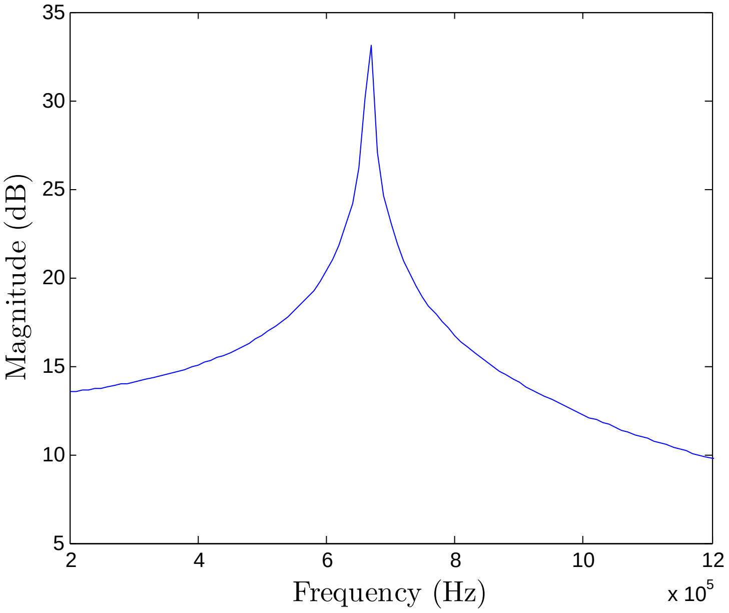


Figure 2.2: Magnitude spectrum S_{IF} of one object in frequency domain

Considering multiple objects scenarios, the transmitted signal will be reflected on each object in the surrounding of the radar. Therefore, the resulting RX signal is the superposition of all RX components which refers to the respective objects. Thus, the unfiltered IF signal results in

$${s_{\mathrm{IF}}}'(t) = s_{TX} \cdot \sum_{i=1}^{N} s_{RX,i} \tag{2.10}$$

for all $i = 1, ..., N$ objects in the radar channel. Figure 2.3 shows the magnitude spectrum of the IF signal for the case of various objects in frequency domain. With the use of (2.10), the data was generated in order to implement and test the CFAR procedure. The transmission of the data was simulated over an AWGN channel.

Again, the beat frequencies and further the distances of the detected objects refer to the peaks in the frequency spectrum. Their heights are defined by the power the receiving antenna

$$P_{\mathrm{r}} = \frac{P_{\mathrm{t}} G_{\mathrm{t}} G_{\mathrm{r}} \lambda^2 \sigma}{(4\pi)^3 d^4} \tag{2.11}$$

which inherently depends on the transmitted power P_{t}, the distance between object and radar d, the transmitter/receiver gain G and the radar operating frequency wavelength λ. Furthermore, the received power is significantly influenced by the object's radar cross section σ which is a measure of an object's detectability. The second and third peak in Figure 2.3 are at the same height, because of the higher σ of the third object. Factors such as the material, the size or the polarisation of radiation of an object determines the radar cross section.

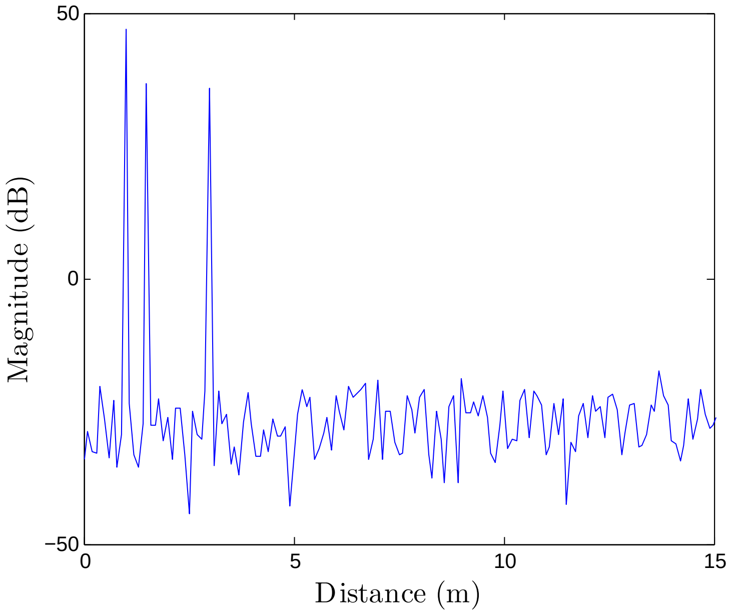


Figure 2.3: Magnitude spectrum S_{IF} of several objects in frequency domain

3 CFAR Methods

The major challenge in object detection is to decide if a peak in the spectrum corresponds to a potential object or not. Comparing the frequency spectrum to a fixed threshold value could work perfectly for an ideal spectrum as shown in Figure 2.2. In a real measurement the presence of noise with unknown power may cause many false alarms if the threshold value was chosen too low. Conversely, if it is set too high, fewer objects will be detected. The CFAR procedure should provide an output which is adapted to the noise floor and ensures that the number of false alarms does not depend on the noise power. The assumed noise model is a zero-mean complex-valued Gaussian random variable, which is independently and identically distributed (iid). To find an adaptive threshold for the given noise model, the noise power has to be estimated. In the following the two most common CFAR algorithms in literature are presented.

3.1 Square law detection

Before delving into specific implementation details of CFAR algorithms, the input of those algorithms has to be defined. According to [1] the best results will be achieved if a square law detector is used, before the CFAR algorithm is applied to the radar data, which is distorted by Gaussian noise. Using square law detection allows to consider noise as the real-valued square of its power rather than using its real and imaginary part. As a consequence, the squared noise data consists exclusively out of positive values. Therefore, the probability density function of the noise model changes.

Assuming that real and imaginary part of the complex noise values follow the Gaussian probability density function, the joint probability density function of the bivariate normally distributed random variables x_1 and x_2 can be written as

$$f_{\mathrm{x}}(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left(\frac{-1}{2(1-\rho^2)}\left(\frac{(x_1-\mu_1)^2}{\sigma_1^2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2} + \frac{2\rho x_1 x_2}{\sigma_1\sigma_2}\right)\right), \tag{3.1}$$

where μ_1 or μ_2 are the expected values, σ_1 or σ_2 describes the variances and ρ is called the correlation coefficient between the two random variables [2, ch. 13]. Considering a zero-mean noise source ($\mu_1 = \mu_2 = 0$) with independent ($\rho = 0$) and identically distributed ($\sigma_1 = \sigma_2 = \sigma$) real and imaginary part, (3.1) can be simplified to

$$f_{\mathrm{x}}(x_1, x_2) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x_1^2 + x_2^2}{2\sigma^2}\right). \tag{3.2}$$

The square law detector sums the squares of the real and imaginary part. This can graphically be represented by a circle. So the probability function of $x_1^2 + x_2^2$ equals an integral over a circle with radius $\sqrt{w}$ of the joint probability function from (3.2). To simplify the integration over the circle, the coordinates are transformed into polar coordinates, resulting in

$$F_{\mathrm{w}}(w) = Pr[x_1^2 + x_2^2 \leq w] = \int_0^{2\pi} \int_0^{\sqrt{w}} \frac{1}{2\pi\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) r dr d\phi. \tag{3.3}$$

Solving the integrals of (3.3) leads to

$$F_{\mathrm{w}}(w) = 1 - \exp\left(-\frac{w}{2\sigma^2}\right), \tag{3.4}$$

which equals the cumulative distribution function of an exponential probability. The related probability density function is defined as [3]

$$f_{\mathrm{w}}(w) = \frac{1}{2\sigma^2} \exp\left(-\frac{w}{2\sigma^2}\right). \tag{3.5}$$

This shows that the Gaussian noise source equals an exponential distribution with double its variance after square law detection. According to (2.9) every frequency bin of the spectrum of the IF signal corresponds to a specific range. Each of this so-called range cells should be analysed whether an object is located in this cell or not. Therefore, two algorithms are presented in the following. They both determine an adaptive threshold value, according to the noise floor, which is compared with each range cell in order to detect objects.

3.2 Cell-averaging CFAR

The first algorithm is called cell-averaging CFAR (CA)-CFAR where the threshold for each value under test Y is evaluated by averaging over the neighbouring cells X_1 to X_{N}. The principle of the CA-CFAR algorithm is shown in Figure 3.1. Averaging over the reference windows consisting out of N values, presents the background noise estimation of this CFAR algorithm. Because of the fact, that peaks are not located at one cell but rather extends across some range cells, the reference window is not directly placed nearby the test cell. Those left out cells, which are placed directly next to Y are called guard cells. The resultant mean of the values in the reference window is defined by

$$Z = \frac{1}{N} \sum_{i=1}^{N} X_{\mathrm{i}}, \tag{3.6}$$

which forms the basis for the adaptive threshold calculation. Multiplying Z by a scaling factor T leads to the threshold value. T is constant for each cell and evaluated depending on the window size N as well as the given probability of a false-alarm P_{FA}. If the value Y exceeds the threshold value $T \times Z$ the comparator declares that a object is located in cell Y.

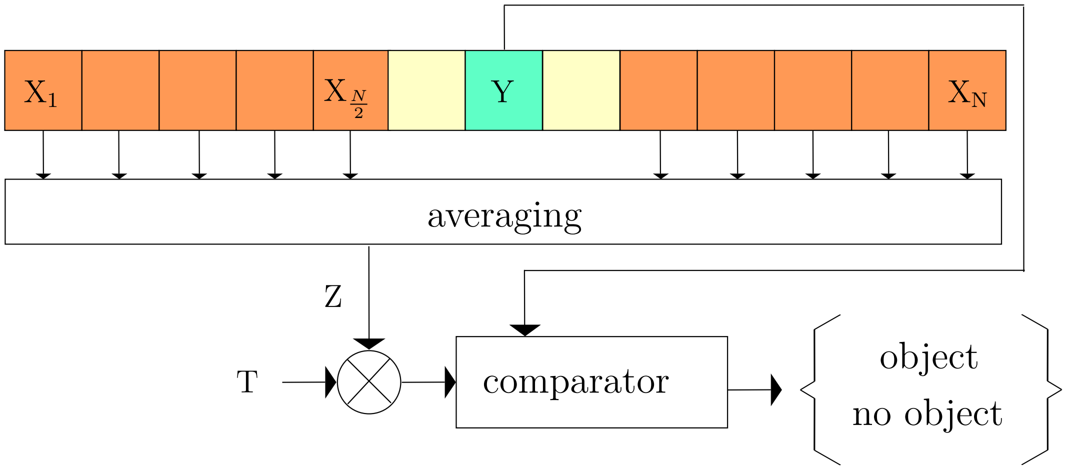


Figure 3.1: Principle of CA-CFAR algorithm

Calculation of the scaling factor T

To get an adaptive threshold value for comparison, a scaling factor has to be found in order to properly scale the result of the averaging process Z. For a specified probability of a false alarm P_{FA} and a fixed window size N, a constant T is determined. Therefore, the changes in the probability density function of Gaussian noise due to the averaging process have to be evaluated. As discussed, the input of the CFAR algorithms is exponentially distributed if a square law detection is used. The first step in the averaging process is the calculation of the sum of N values. If two independently distributed random variables X_1 and X_2 with their probability density functions $f_{\mathrm{X}_1}(x_1)$ and $f_{\mathrm{X}_2}(x_2)$ are summed up, the probability density function of the sum f_{Z} is written as the convolution of their density functions

$$f_{\mathrm{Z}}(z) = \int_{-\infty}^{\infty} f_{X_1}(x_1) f_{X_2}(z - x_1) dx_1. \tag{3.7}$$

As shown in (3.5), the values in the reference window are exponentially distributed with equal expected value $\mu = 1/2\sigma$. Thus, the probability density function

$$f_{\mathrm{Z}}(z) = f_{X_1} * f_{X_2} = \mu^2 z e^{-z\mu} \tag{3.8}$$

describes the distribution for the sum of two exponentially distributed random variables. In fact, the process can be extended to a sum of N random variables, where the resulting probability density function

$$f_{\mathrm{Z_N}}(z) = \frac{\mu^N z^{N-1} e^{-z\mu}}{(N-1)!}, \tag{3.9}$$

follows the Erlang distribution, which is a special case of the Gamma distribution. The second step in the averaging process is the division by N. Dividing by a constant does not change the probability density function, since the size of the reference window is fixed.

To get an expression for the scaling factor T, the definition of the probability of a false alarm is written as

$$P_{\mathrm{FA}} = E[Pr[Y \geq T \cdot Z | H_0]]. \tag{3.10}$$

The hypothesis H_0 implies the presence of noise only. In contrast H_1 assumes the presence of an object. The probability of a false alarm equals the possibility that the cell under test exceeds the threshold value under the condition H_0, which means only noise is present. So the probability density function of Y is integrated over all values greater than the threshold in order to get an expression for the probability of a false alarm

$$P_{\mathrm{FA}} = \frac{1}{N} \int_0^\infty \int_{TZ}^\infty \frac{1}{2\sigma^2} exp\left(\frac{-y}{2\sigma^2}\right) dy \frac{1}{(N-1)!} \left(\frac{1}{2\sigma}\right)^N z^{N-1} exp\left(\frac{-z}{2\sigma}\right) dz. \tag{3.11}$$

As discussed, Y is exponentially distributed and Z is Gamma distributed. Therefore, (3.11) simplifies to

$$P_{\mathrm{FA}} = \frac{1}{(1+\frac{T}{N})^N}, \tag{3.12}$$

which is no longer dependent on statistical properties of the Gaussian noise model. Thus it can be transformed into an expression for the scaling factor

$$T = N({P_{\mathrm{FA}}}^{-\frac{1}{N}} - 1). \tag{3.13}$$

This scaling factor T is constant for fixed values of P_{FA} as well as N, and it can therefore be calculated offline. With the multiplication of T, the determined average Z evolves into an adaptive threshold for object detection, which satisfies the specified probability of a false alarm P_{FA} for an AWGN channel. The CA-CFAR algorithm returns a Boolean array, stating if a range cell has exceed the adaptive threshold or not.

The detection procedure in CA-CFAR algorithms is not designed for multiple object detection. Other objects inside the reference window distort the noise estimation and increase the threshold value as a consequence. Therefore, an algorithm that should overcome this problem is presented in the next Section.

3.3 Ordered-statistic CFAR

In contrast to the CA-CFAR procedure, which uses all signal amplitudes in the reference window to determine a threshold, the OS-CFAR algorithm only selects a single amplitude. As shown in Figure 3.2, the basic design is similar to the CA-CFAR. Again, the sliding reference window is surrounding the cell under test Y and its guard cells. The general idea of an ordered-statistic is that the noise estimation is based on the k^{th} values of reference values sorted in ascending order

$$X_1 \leq X_2 \leq ... \leq X_k \leq ... \leq X_{N-1} \leq X_N. \tag{3.14}$$

Accordingly, if there is another object present in the reference window its value is not affecting the peak detection in the cell under test Y. The arithmetic mean used in CA-CFAR algorithms is replaced by a single rank of the ordered-statistic X_{k}. Identically to the CA-CFAR, the estimation of the noise floor has to be multiplied by a scaling factor T_{OS} to obtain the threshold value. Since the OS-CFAR uses only a single value to determine the threshold value, the choice of N is less important compared to the CA-CFAR.

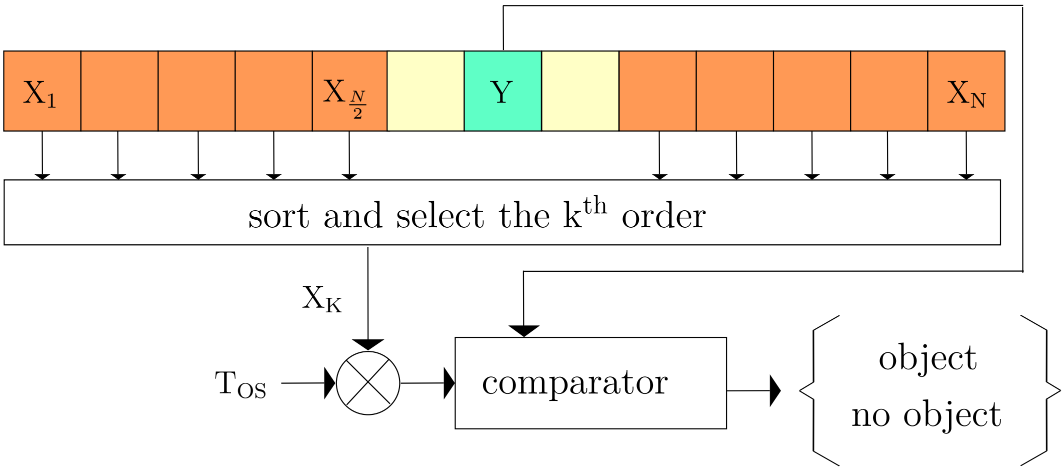


Figure 3.2: Principle of OS-CFAR algorithm

Calculation of the scaling factor T_{OS}

At first, a suitable value for k must be found, though it decides which value out of the reference window is used for the background noise estimation. According to [4] three quarters of the window size N is a reasonable choice for k. Furthermore, k has to be multiplied by a scaling factor in order to achieve a given constant probability of false alarm. The scaling factor T_{OS} can be calculated by solving

$$P_{\mathrm{FA}} = k\binom{N}{k}\frac{(k-1)!(T_{\mathrm{OS}}+N-k)!}{(T_{\mathrm{OS}}+N)!}, \tag{3.15}$$

for a given probability of false alarm and the k^{th} value of an ordered-statistic array of exponentially distributed values [5, ch. 6.5.6]. In contrast to the CA-CFAR, (3.15) can not be transformed into a closed formula for the scaling factor T_{OS}. Therefore, T_{OS} was found iteratively, which results in a longer run time. Plots such as Figure 3.3 can be used to determine the scaling factor T_{OS} to achive a specified P_{FA} for a given reference window size N. The selected order-statistic is chosen as approximately $k = 3/4N$.

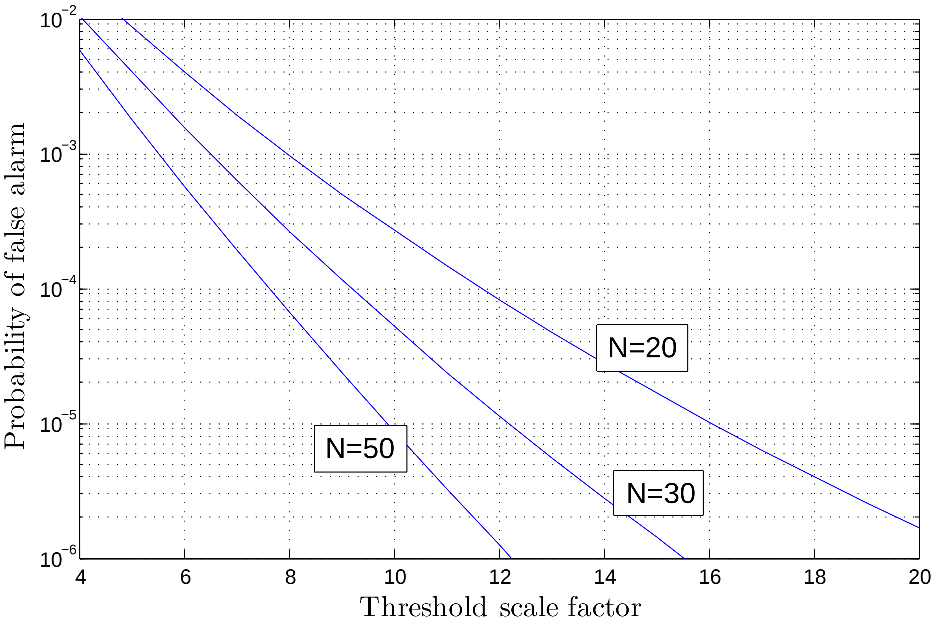


Figure 3.3: $\mathrm{P_{FA}}$ versus threshold scale factor $\mathrm{T_{OS}}$

4 The effect of windowing

Fourier analysis converts the signal from time domain into the respective frequency spectrum. Therefore, the fast Fourier transform (FFT) takes the amount of data and repeats it with the assumption that this signal continues for all time. As a consequence, there may occur sudden transitions at the borders of each replica. These sharp transitions mean a broad frequency response. This high-frequency components, which are not in the original time signal, are called spectral leakage and should be reduced by windowing. Thus, a finite function, the so-called window function, is applied to the set of data in time domain, which reduces the spectral leakage. A window function is mainly characterized by the width of the main-lobe and the height of the side-lobes. In this chapter three window functions, shown in Figure 4.1, are analysed and specifically the effect of windowing on the CFAR procedure is investigated. Furthermore the effect on the CFAR procedure of the rectangular window, the Hann window as well as the Dolph-Chebyshev window are pointed out. With regard to the similar outcomes between the OS- and the CA-CFAR for windowed data, this section only shows the results of the OS-CFAR algorithm.

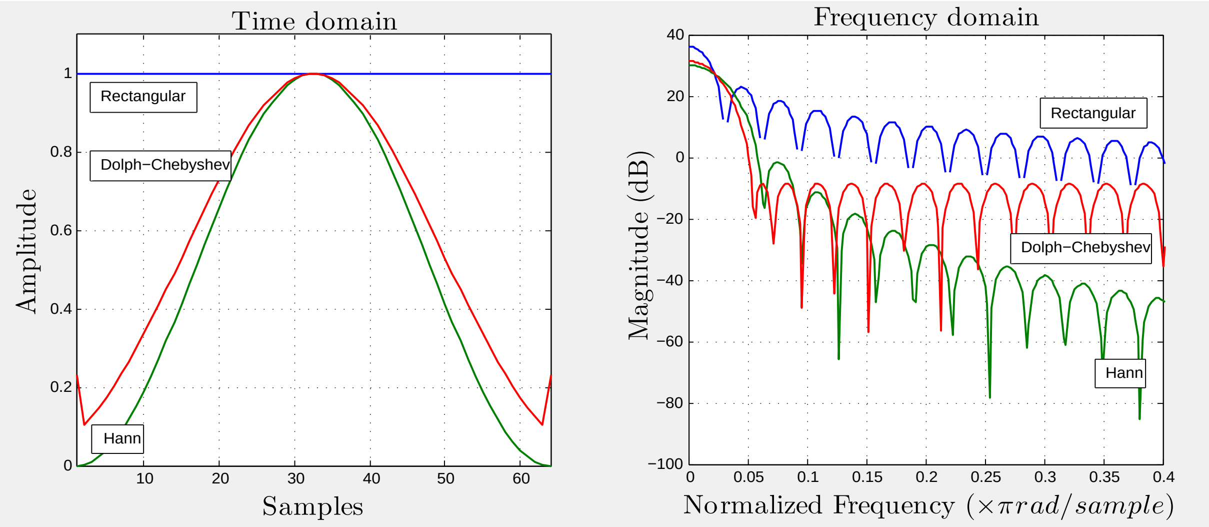


Figure 4.1: Rectangular window, Hann window and Dolph-Chebyshev window in time and frequency domain

Rectangular Window

The rectangular window equals the results if no window is applied, meaning the window equals a multiplication with one. This leads to sharp transitions at the signal borders, resulting in a very high side-lobe level, as presented in the frequency response of Figure 4.1. Since the side-lobe level is high in comparision to the main-lobe, an object peak in the range spectrum differentiates less from the surroundings. On the other hand, the width of the main-lobe is comparatively slim, this helps to locate an object precisely. For the application of CFAR algorithms as represented in this thesis, the use of a rectangular window is recommended. Due to this window function, the correlation between the samples in the reference window is not changed (they follow the presumed noise model and are iid). As illustrated in Figure 4.2, the CFAR algorithm is able to detect all three simulated objects in the channel.

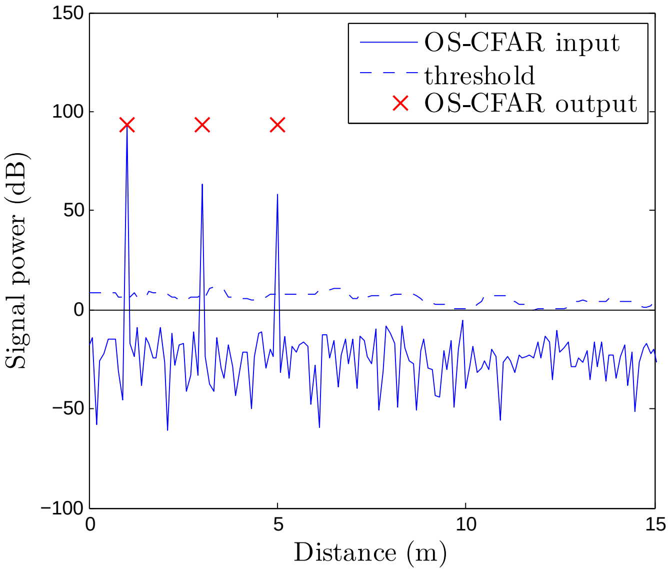


Figure 4.2: OS-CFAR applied to rectangular windowed data

Hann Window

The Hann window is frequently used. Significantly this window function is equal to zero at the beginning and at the end. Hence, the window function is applied to, is multiplied with zero at the transition areas, as in the time domain in Figure 4.1 shown. Therefore, no discontinuities occur when applying the FFT. In consequence of the Hann windowing, the side-lobe level is significantly reduced, but the main-lobe becomes wider. This causes problems for the presented CFAR algorithms. Figure 4.3 shows that the CFAR performance suffers from the windowing procedure and the distance is indicated inaccurately. The CFAR algorithm force the input data to be iid, which is not satisfied when a windowing function is applied. According to [6], the CFAR procedure has to be modified due to the correlation of the samples.

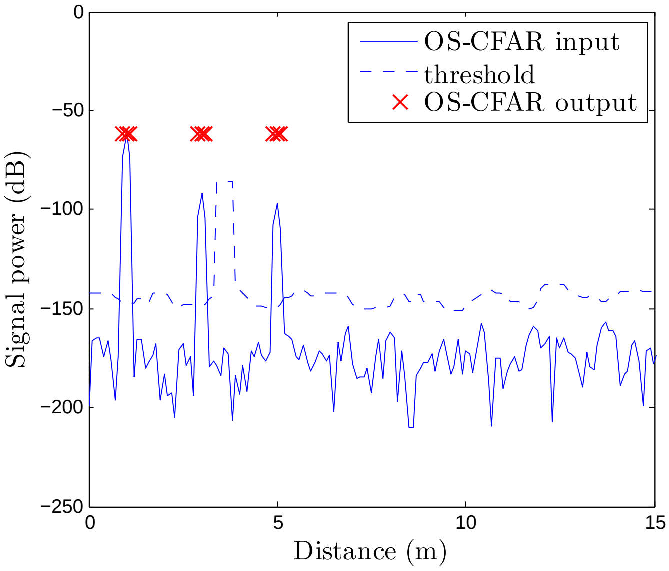


Figure 4.3: OS-CFAR applied to Hann windowed data

Dolph-Chebyshev Window

Most window function, including the Rectangular window and the Hann window, are defined in the time domain. The Dolph-Chebyshev Window is specified in the frequency domain, therefore, the side-lobe level can be specified precisely. Figure 4.1 shows, that the side-lobes in the frequency domain are at equal height, the so-called *equal ripple* property. Compared to other window functions, the lowest P_{FA} error occurs when using a Dolph-Chebyshev window with a 40dB side-lobe level [6]. The main-lobe differentiates from the nearly constant side-lobe level. It can be seen from Figure 4.4 that the detection performance is improved compared to the Hann window. Still, not every object was detected perfectly as it was the case for the rectangular window.

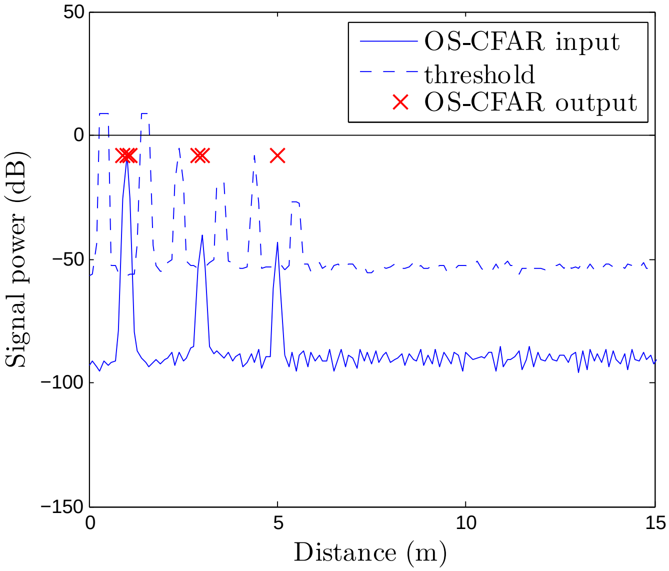


Figure 4.4: OS-CFAR applied to Dolph-Chebyshev windowed data

5 Comparision of CFAR algorithms

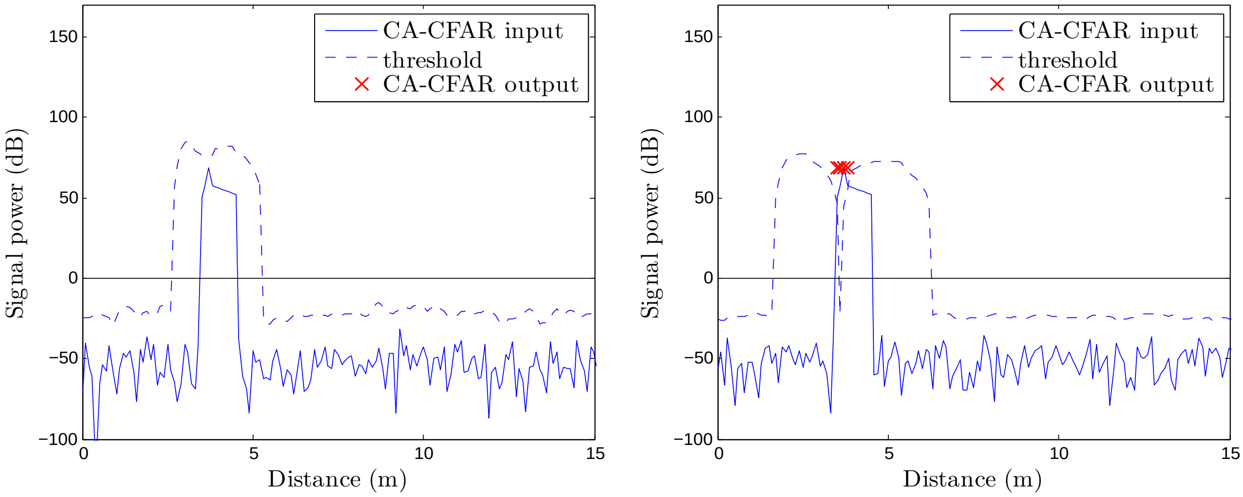
For comparison, the presented algorithms will be tested in defined scenarios containing clutter, multi object scenarios and an increased noise floor. As a performance measure it will be checked if all potential objects can be detected or not. In the following, the scenarios will be explained and compared using Matlab simulations. There are a few problems when it comes to object detection using radar measurements which either modify the specified false alarm rate or minimize the detection probability. Both of them will be discussed in the following chapter and considered in the comparison of the algorithms.

5.1 Clutter

According to [7], clutter denotes all undesired background signals from the standpoint of object detection. In automotive application multipath reflections, for example from crashing barriers, can cause unwanted echos in the measuring environment. The CFAR algorithms are forced to identify this clutter as irrelevant. Therefore, the threshold value should be increased in clutter regions. The specified false alarm rate is often not satisfied for environments containing clutter. This is due to the fact, that the edges will be detected as objects when the threshold is not rising high enough. If the sliding window is chosen much larger than the clutter area, the clutter samples have less influence on the average. In Figure 5.1a the clutter region was interpreted correctly for a window size of N=12. In contrast, for N=16 the CA-CFAR algorithm detects objects at the left clutter edge, as illustrated in Figure 5.1b. The OS-CFAR algorithm does not overcome the problem either, the simulation results are similar to the CA-CFAR, as shown in Figure 5.2. The clutter in the simulated environment extends over seven range cells. If the size of the reference window is more than twice the clutter area, the clutter samples affect the threshold result less and it will be detected as several objects. As a result, the specified false alarm rate is not satisfied for both algorithm with N=16.

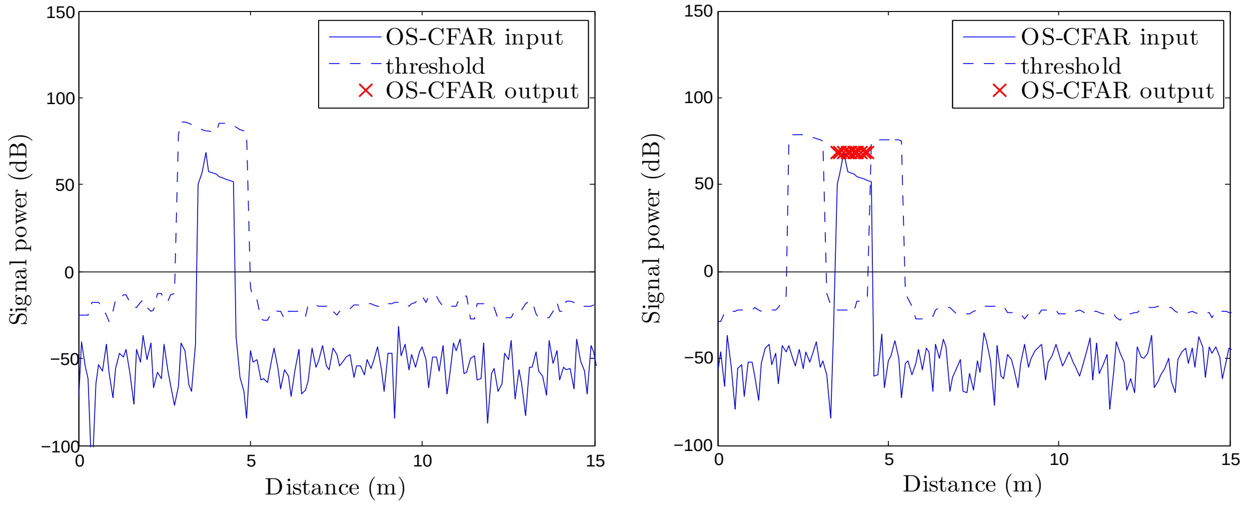
5.2 Object masking

As outlined in chapter 3, the basis of the CFAR procedure forms the noise model in the reference window. In situations where many objects are present, masking effects can occur. In that case, the threshold is too high to detect an object, since the other object peaks are increasing the threshold. From an estimation point of view, a larger reference window provides a more reliable



a) N=12 b) N=16

Figure 5.1: Simulation results of the CA-CFAR algorithm in clutter environment



a) N=12 b) N=16

Figure 5.2: Simulation results of the OS-CFAR algorithm in clutter environment

noise estimation, as the following equations show [3, ch. 3.2],

$$Z = \frac{1}{N}\sum_{i=1}^{N} X_i$$

$$\lim_{N\to\infty} Pr[|Z-\mu| \geq \epsilon] = 0 \tag{5.1}$$

for all $\epsilon > 0$. In (5.1), μ describes the expected value of the background noise and Z represents its estimation. Note that the law of (5.1) holds only for real valued numbers, which is satisfied due to the square-law detection as described in chapter 3. Since the noise estimation is distorted by all the other objects located in the reference window, the CA-CFAR algorithm fails in multiple object situations, as illustrated in Figure 5.3. That means using a bigger reference window could worsen the estimation due to the masking effect. Therefore, the impact on the reference window size N is much higher compared to the OS-CFAR algorithm where the k^{th} value is used for the threshold calculation. As a result, the peaks in the reference window have little influence on the noise estimation. As shown in Figure 5.4, the OS-CFAR suffers less from those masking effects, since the objects are detected correctly.

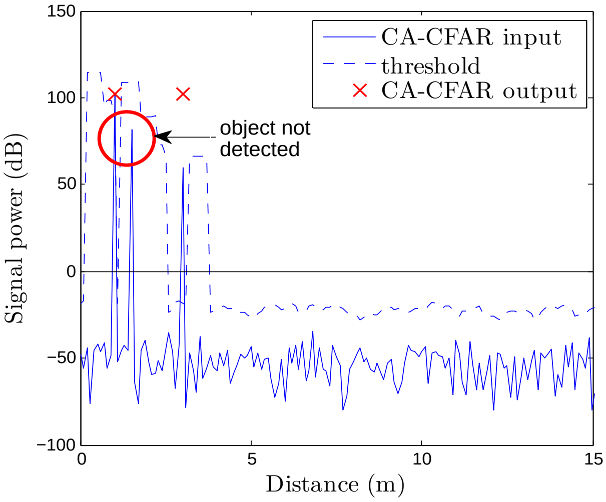


Figure 5.3: CA-CFAR in a multiple object scenario

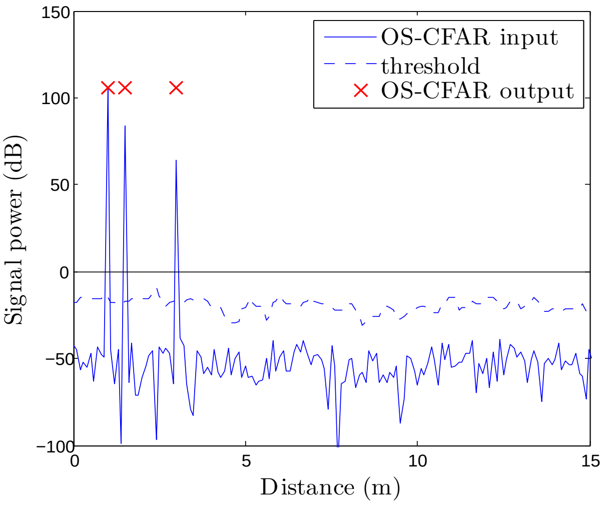


Figure 5.4: OS-CFAR in a multiple object scenario

Due to the unacceptable performance of CA-CFAR in terms of masking effects, there exist some extensions. The basic idea of the cell-average-smallest-of (CASO) CFAR is that only the side of the reference window is regarded which holds the smaller averaging value. But it is obvious that the better performance in object masking of the CASO-CFAR algorithm comes with the problem that the clutter edges often will be detected as objects. This happens because the part of the reference window which contains the clutter area holds the greater average value and therefore will be completely ignored by the CASO-CFAR procedure. Implementations and calculations details of the extended CA-CFAR algorithms are available in [5, ch. 6.5.5].

5.3 Increased noise power

A constant threshold, which is not adapted to the noise, would be very difficult to find. On the one hand, it would cause a high false alarm rate if it is set too low in noisy environments. On the other hand, it worsen the detection probability drastically if it is set too high. Thus, the specification of CFAR algorithms is, that the object detection gets independent of AWGN through an adaptive threshold. In order to identify if the algorithms fulfil this requirement, they are tested in a noisy environment without any objects. The results in Figure 5.5 and Figure 5.6

show, that both algorithms rise the threshold properly when the noise power is increased.

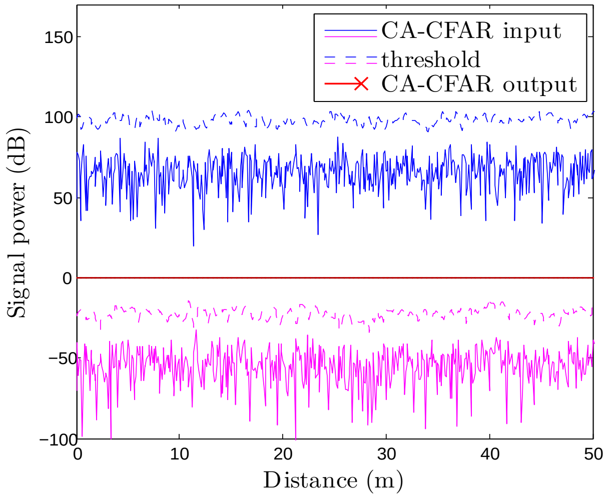


Figure 5.5: CA-CFAR on noise only data

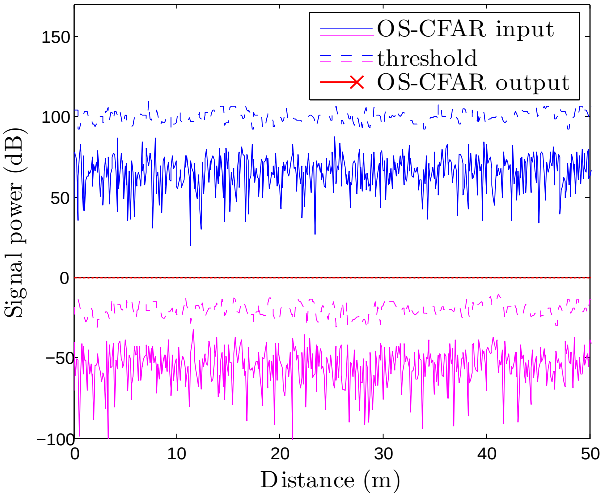


Figure 5.6: OS-CFAR on noise only data

This raises the question, if the object detection works in noisy area. In the previous section the peaks in the range cells were clearly visible, but now the signal-to-noise-ratio (SNR) of the data is decreased. In the following scenario the objects are located in distances one, three and five meters to the radar. The distances were chosen in order to prevent the algorithms from masking effects. As shown in Figure 5.7 and 5.8, the CA-CFAR and the OS-CFAR algorithms are still able to detect the objects, when noise power is increased. The simulations show that noisy data can still be analysed, since the noise source follows the statistical properties of AWGN.

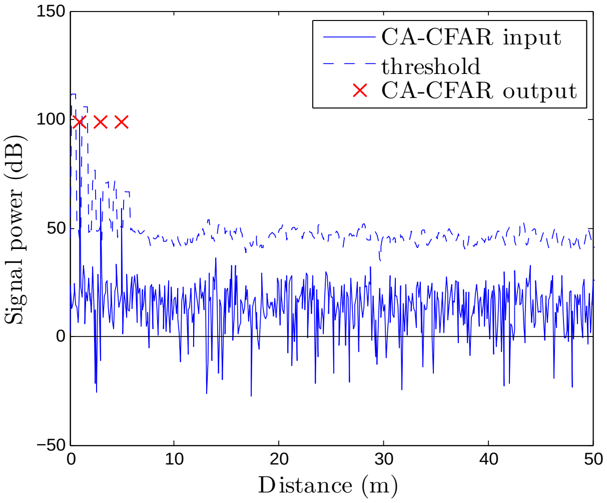


Figure 5.7: CA-CFAR in noisy background

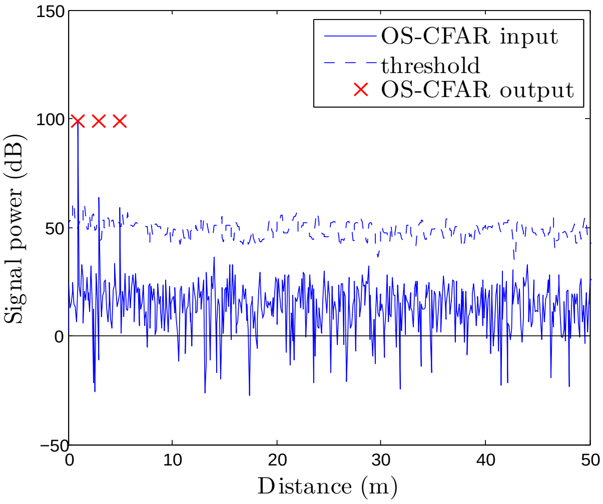


Figure 5.8: OS-CFAR in noisy background

6 Measurement results

In the first measurement data set, the false alarm rate is determined in a noise only environment. In contrast to the simulation results in Section 5.3, both CFAR algorithms declare that an object is located directly nearby the radar antenna. This phenomena is called DC-offset and appears due to reflections of transmitting and receiving antenna, resulting in crosstalk. The object detection based on the DC Offset in Figure 6.1 and Figure 6.2 is eliminated by using a high-pass (HP) filter, as Figure 6.3 and Figure 6.4 reveal. Apart from that, the threshold calculation of both algorithms work properly, hence no false alarms occur in the data series.

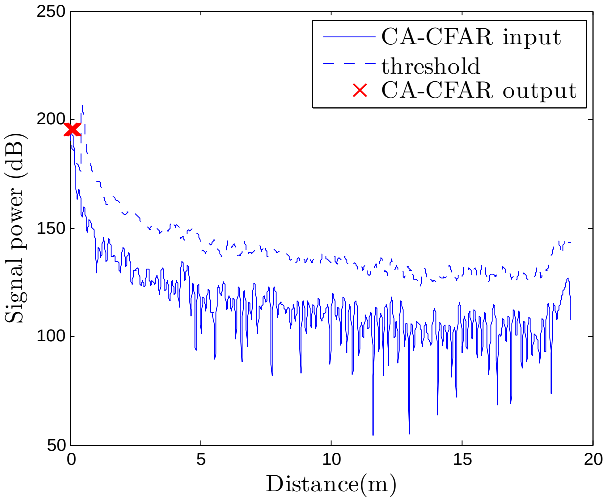


Figure 6.1: CA-CFAR on noise only data with DC-offset

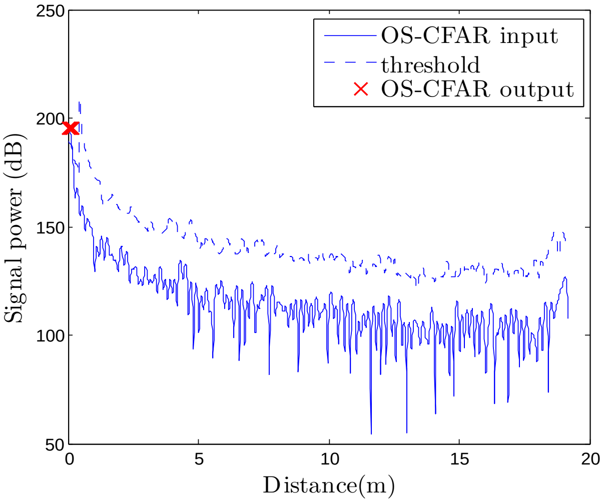


Figure 6.2: OS-CFAR on noise only data with DC-offset

In order to test the implemented algorithms according to the object masking problems, stated in Section 5.2, there are two objects placed within two meters inside the measurement range. The CA-CFAR is expected to perform worse than the OS-CFAR, since the CA-CFAR procedure uses every value of the reference window, which make it more vulnerable to this effect. Figure 6.5 confirms this expectations, since the CA-CFAR misses the second object due to the distortion of the noise estimation by the first object. Using the same parameters for the OS-CFAR procedure, leads to the results in Figure 6.6. Significantly, the two objects are detected by the OS-CFAR in noisy environment.

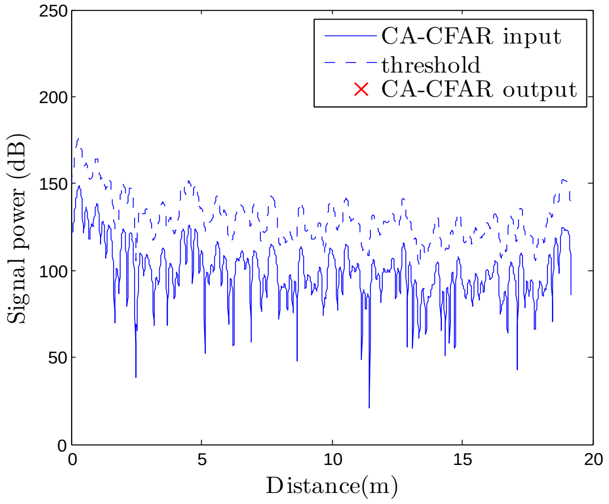


Figure 6.3: CA-CFAR on noise only data after HP filtering

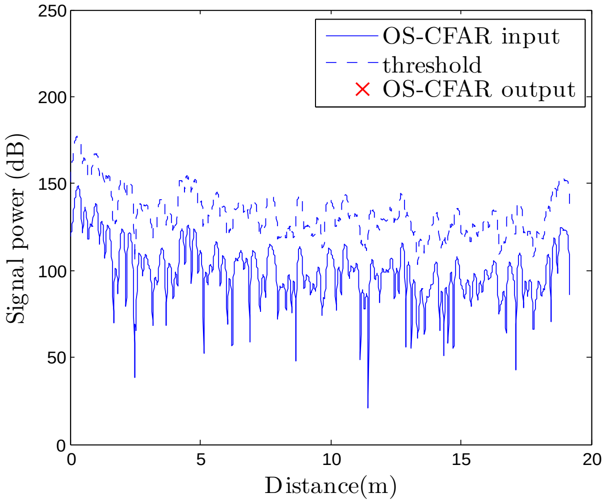


Figure 6.4: OS-CFAR on noise only data after HP filtering

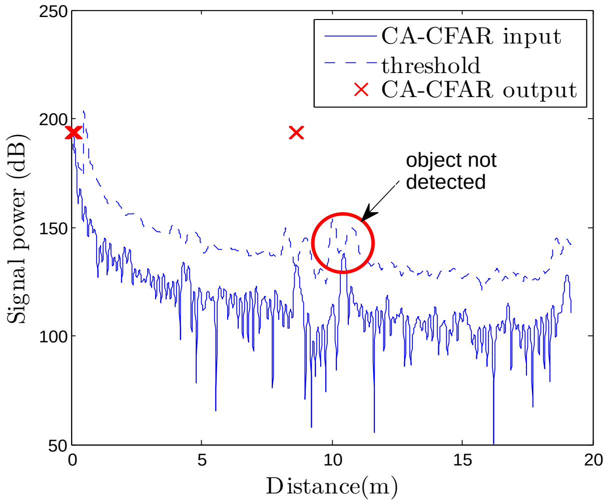


Figure 6.5: CA-CFAR in multiple object scenario with DC-offset

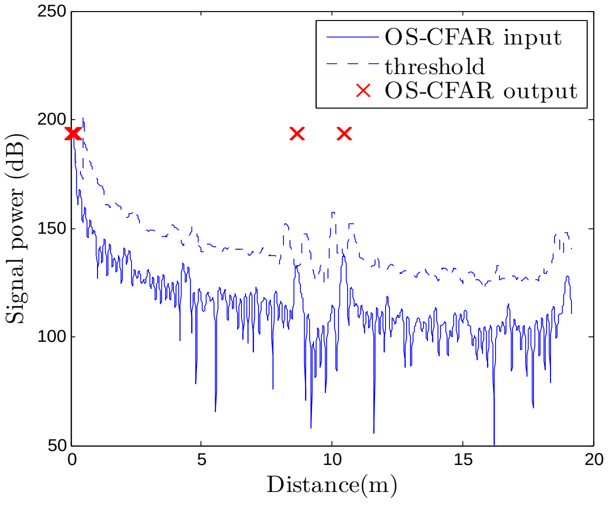


Figure 6.6: OS-CFAR in multiple object scenario with DC-offset

7 Conclusion

Both presented algorithms are compared by the means of the simulation and measuring results. The most significant disadvantage of the CA-CFAR is, that the scaling factor T is examined based on the assumption that white Gaussian noise samples are contained in the reference window. Consequently, every peak or clutter sample distorts the noise estimation. In contrast, the OS-CFAR only uses a specified value, namely the k^{th} order of the sorted reference window. Therefore, potential peaks or clutter samples are mostly ignored for the threshold calculation in OS-CFAR. Simulation and measurement results clearly demonstrates the resultant masking effects of the CA-CFAR algorithm.

In terms of threshold calculation the CA-CFAR algorithm is easier to handle. The scaling factor used in CA-CFAR is a closed formula, depending on the specified parameters, in contrast, the OS-CFAR determines the threshold scaling factor by means of iterative calculation. Accordingly, the OS-CFAR suffers from high computation costs, as an alternative, the threshold scaling factor for fixed values of N, k and P_{FA} can be written in look-up tables. Therefore, the iteration process is not necessarily required.

Another point to consider is the inaccuracy which CFAR algorithms show, when a windowing function is applied beforehand. A workaround is to scale the window size N to an effective number of the reference samples due to the correlation of the formerly iid samples [6]. Certainly, the size of the reference window affects the performance of the CFAR algorithms. In a direct comparison of CA and OS algorithm, the size of the reference window and the number of guard cells are always defined equally. The results of the CA-CFAR algorithm vary strongly, while the results of the OS-CFAR algorithm are less affected with changing N. On the other hand, the value of k has a high influence on the OS-CFAR procedure. Anyhow, the authors of [4] recommend to choose k as three quarters of N and thus suitable results are obtained. Therefore, the OS-CFAR provides results that are robust towards parameter variations.

Having considered all these arguments, the OS-CFAR is definitely the better choice between the two presented algorithms for automotive applications with the focus on proper object detection. In this application, the problems of CA-CFAR algorithms with masking effects are serious. Although the CA-CFAR algorithm provides a closed formula for the scaling factor T, it should be noted that the iteration results of the OS-CFAR algorithm can be provided offline, due to the fact that, the scaling factor T is constant for fixed N, k and P_{FA}. Thus, the OS-CFAR algorithm is suitable in automotive applications for real-time signal processing. This thesis only refers to the fundamental OS-CFAR and CA-CFAR concepts, therefore, no statements of the extended versions of these algorithms are made.

References

[1] A. Melebari, A. K. Mishra, and M. Y. A. Gaffar, “Comparison of square law, linear and bessel detectors for CA and OS-CFAR algorithms,” in *Proc. of the 2015 IEEE Radar Conference*, Oct. 2015, pp. 383–388.

[2] J. Groß, *Grundlegende Statistik mit R.* Wiesbaden: Vieweg+Teubner Publishing Company, 2010, ISBN: 978-3-8348-1039-7.

[3] G. Bourier, *Wahrscheinlichkeitsrechnung und schließende Statistik.* Wiesbaden: Gabler Verlag | Springer Fachmedien, 2011, ISBN: 978-3-8349-2762-0.

[4] M. Habib, M. Barkat, B. Aïssa, and T. Denidni, “CA-CFAR detection performance of radar targets embedded in non centered chi-2 gamma clutter,” English, *Progress in Electromagnetics Research*, pp. 135–148, 2008.

[5] M. A. Richards, *Fundamentals of radar signal processing.* New York: McGraw-Hill, 2014, ISBN: 978-0-0717-9832-7.

[6] A. Melebari, A. Melebari, W. Alomar, M. Y. A. Gaffar, R. D. Wind, and J. Cilliers, “The effect of windowing on the performance of the CA-CFAR and OS-CFAR algorithms,” in *2015 IEEE Radar Conference*, Oct. 2015, pp. 249–254.

[7] H. Rohling, “Ordered statistic CFAR technique - an overview,” in *Proc. of the 12th International Radar Symposium (IRS)*, Sep. 2011, pp. 631–638.