IEICE Electronics Express, Vol.16, No.21, 1–6

LETTER

Reconfigurable hardware architecture for Mean Level and log-t CFAR detectors in FPGA implementations

Jiafei Zhao[1], **Rongkun Jiang**[1a)], **Hao Yang**[1], **Xuetian Wang**[1], **and Hongmin Gao**[1]

Abstract For radar target detection, the selection of the optimal constant false alarm rate (CFAR) detector usually relies on clutter distribution types. By integrating two types of Mean Level and log-t CFAR detectors, a reconfigurable hardware architecture is proposed and implemented on field programmable gate array (FPGA). It allows to switch a suitable detector for specific clutter distribution and configure the parameters including the number of reference and guard cells, the threshold factor, and the desired false alarm probability. Synthesis results reveal its advantages of occupying 18% less hardware resources than the architecture that naively integrates two types of detectors. According to the experimental results, the proposed architecture can perform a processing speed of 100 MHz and require only 83 microseconds for a clutter of 8192 samples.
Keywords: radar, target detection, CFAR, Mean Level, log-t, FPGA implementation
Classification: Integrated circuits

1. Introduction

Constant false alarm rate (CFAR) processing plays an important role in radar target detection [1, 2, 3]. Facing the varying electromagnetic environment, the purpose of CFAR technology is to enable the radar system to maximize the detection probability while maintaining a fixed false alarm probability. In order to achieve this, it is necessary to estimate the power of the interference clutter in real time and adjust the detection threshold adaptively and dynamically to retain the specified false alarm probability [4, 5, 6].

Historically, a number of CFAR detection algorithms have been published in the literature. However, most of them are optimal detection for a particular type of clutter distribution. For instance, for the Rayleigh distribution, there are the type of Mean Level (ML) CFAR detectors including cell averaging (CA) [7], smallest-of (SO) [8], and greatest-of (GO) [9], the type of ordered statistics (OS) CFAR detectors [10], and the type of generalized ordered statistic (GOS) CFAR detectors using automatic screening techniques [11], etc. For a non-Rayleigh clutter, such as Weibull, Log-normal, and K distribution, there are log-t CFAR [4, 12, 13], maximum likelihood (MLH) CFAR [14], best linear unbiased estimation (BLUE) CFAR [15] detectors, and so on. In practice, if the actual clutter environment is not consistent with the predetermined clutter type, the false alarm probability of CFAR processor will be greatly increased. Therefore, radar systems usually need to select the optimal CFAR detectors according to the different types of clutter distribution statistics. Owing to the simplicity and practicability, the ML CFAR and log-t CFAR detectors are widely used in practical scenarios [16, 17, 18, 19]. In this letter, the attention is focused on the implementation of these two types of CFAR detectors.

With the development of digital circuit technology, field programmable gate array (FPGA) has emerged as an attractive integrated circuit for hardware implementation to support high speed algorithms and intensive computation applications. It comprises a large number of configurable logic elements, which are connected by a programmable structure [20]. FPGA can effectively reduce the difficulty of hardware design and shorten the development cycle. In radar signal processing field, [21] and [22] proposed configurable FPGA-based hardware architectures for three versions of CFAR algorithms, including the CA, GO, and SO. [23] presented the FPGA-based implementations of CA-CFAR and OS-CFAR detectors, focusing on the adaptive pseudo noise (PN) code acquisition in multipath spread spectrum communications. [24] investigated a specialized architecture of an FPGA-based CA-CFAR processor with Xilinx integrated circuit chip XC9600. [25] proposed an FPGA-based hardware architecture for Trimmed Mean (TM) CFAR processor for radar target detection. In [26], the hardware architecture of an energy-CFAR processor was described for adaptive filtering based on energy analysis of radar echoes. [27] reported an FPGA-based CFAR target detector for Log-normal clutter based on forward and backward automatic censored cell algorithms. [28] realized a real-time adaptive OS-CFAR processor for homing application in marine environments using KINTEX-Ultrascale FPGA.

However, the hardware architectures in these papers are generally implemented for a particular clutter distribution and not applicable to different clutter distributions. If designing a hardware for each type of clutter distribution separately, it will cause serious resource consumption and complicated logic structure. Therefore, an FPGA-based reconfigurable hardware architecture is proposed to ameliorate the issue in this work. It integrates two types of ML and log-t CFAR detectors into one multiplexing architecture for improving the utilization of hardware resources. Through different control signals, a suitable detector from two types can be switched to adapt to different clutter

[1]School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China
a) jiangrongkun@bit.edu.cn

DOI: 10.1587/elex.16.20190584
Received September 14, 2019
Accepted September 20, 2019
Publicized October 9, 2019
Copyedited November 12, 2019

distribution. Additionally, the detector parameters can be also configured, including the number of reference cells, guard cells, the threshold factor, and the desired false alarm probability. The proposed architecture is implemented on the Xilinx Kintex-7 XC7K325T FPGA board. It saves approximately 18% hardware resources compared with the architecture which naively integrates two types of detectors. Experimental results demonstrate its effectiveness and applicability in typical Rayleigh and non-Rayleigh distributions. With 32 reference cells and 8 guard cells, the processing speed of the proposed architecture can be achieved to 100 MHz and only 83 microseconds are needed for a clutter of 8,192 samples.

The letter is organized as follows. Section 2 explains the fundamental concepts of clutter distribution and CFAR detector. Section 3 presents the proposed CFAR detector architecture. Section 4 describes the FPGA-based implementations and experimental validation results. Finally, Section 5 concludes the letter.

2. Preliminaries

2.1 Clutter distribution

Generally, Rayleigh distribution is the most classical model for describing the amplitude distribution of clutter [29]. Assuming the clutter amplitude as x, the probability density function (PDF) of Rayleigh distribution clutter is given by:

$$f_R(x) = \frac{x}{b^2}\exp\left(-\frac{x^2}{2b^2}\right), \quad x \geq 0 \tag{1}$$

where b denotes the Rayleigh coefficient related by the clutter power.

However, with the development of high resolution radar, a large amount of experimental data present that the occurrence probability of larger clutter increases and results to a longer tail in the actual clutter amplitude distribution than Rayleigh distribution. As a remedy, several distribution models have been proposed to describe statistical properties of non-Rayleigh clutter, consisting of Log-normal, Weibull, and K distribution [12, 30, 31], etc. Specifically, the PDFs of these three non-Rayleigh distributions can be expressed as follows:

$$f_L(x) = \frac{1}{\sqrt{2\pi}cx}\exp\left[-\frac{(\ln x - v)^2}{2c^2}\right], \quad x \geq 0 \tag{2}$$

$$f_W(x) = \frac{c}{v}\left(\frac{x}{v}\right)^{c-1}\exp\left[-\left(\frac{x}{v}\right)^c\right], \quad x \geq 0 \tag{3}$$

$$f_K(x) = \frac{2c}{\Gamma(v)}\left(\frac{cx}{2}\right)^v K_{v-1}(cx), \quad x \geq 0 \tag{4}$$

where v and c represent the shape parameter and scale parameter respectively. $\Gamma(\cdot)$ is the Gamma function and $K_{v-1}(\cdot)$ is the modified Bessel function of the second kind with order $v-1$. Comparing Eqs. (1) to (4), it is evident that the Weibull, Log-normal, and K distributions involve two parameters of v and c, whereas only one parameter b in the Rayleigh distribution. In light of this essential differentiation, it is necessary to adopt various CFAR detection methods for Rayleigh and non-Rayleigh distributions separately.

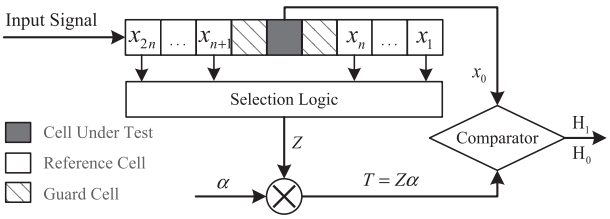


Fig. 1. Schematic diagram of a generic CFAR algorithm.

2.2 CFAR detector

Referring to a Rayleigh distribution, ML CFAR detector is the most typical and simplest type, encompassing the CA, GO, and SO. As shown in Fig. 1, assume x_i presents the continuous clutter signal in the reference window and x_0 denotes the sample of the cell under test (CUT), the decision criterion is given as:

$$x_0 \underset{H_0}{\overset{H_1}{\gtrless}} T \tag{5}$$

where $T = \alpha Z$ refers to the adaptive detection threshold, which is determined by the threshold factor α and the arithmetic mean Z of reference cells surrounding the CUT. For three CFAR algorithms of CA, GO, and SO, the calculation of Z is based upon different schemes [7, 8, 9] as:

$$\begin{cases} Z_{CA} = \dfrac{1}{N}\displaystyle\sum_{i=1}^{2n} x_i & (6) \\ Z_{GO} = \max\left(\dfrac{2}{N}\displaystyle\sum_{i=1}^{n} x_i,\ \dfrac{2}{N}\displaystyle\sum_{i=n+1}^{2n} x_i\right) & (7) \\ Z_{SO} = \min\left(\dfrac{2}{N}\displaystyle\sum_{i=1}^{n} x_i,\ \dfrac{2}{N}\displaystyle\sum_{i=n+1}^{2n} x_i\right) & (8) \end{cases}$$

where $N = 2n$ stands for the number of reference cells in the reference window. Besides, the hypothesis H_1 indicates the presence of a target in the CUT, while H_0 indicates the absence.

For non-Rayleigh distributions, many studies have confirmed that the log-t detector can maintain constant false alarm characteristics in Weibull and Log-normal distributions with unknown parameters, as well as K distribution [12, 32, 33]. The principle of a log-t CFAR detector is to compress the clutter signal logarithmically and form a two-parameter test statistic. Assuming $y_i = \ln x_i$, the test statistic of log-t detector can be constructed as:

$$\frac{y_0 - \dfrac{1}{N}\displaystyle\sum_{i=1}^{N} y_i}{\sqrt{\left[\dfrac{1}{N}\displaystyle\sum_{i=1}^{N}\left(y_i - \sum_{j=1}^{N} y_j\right)^2\right]}} \underset{H_0}{\overset{H_1}{\gtrless}} T \tag{9}$$

where y_0 is the clutter sample of CUT.

By comparing the detection statistics of ML and log-t detectors, it can be seen that the constant false alarm detection with non-Rayleigh distributions is much more complicated than that with the Rayleigh distribution. Therefore, different detectors are needed for various clutter

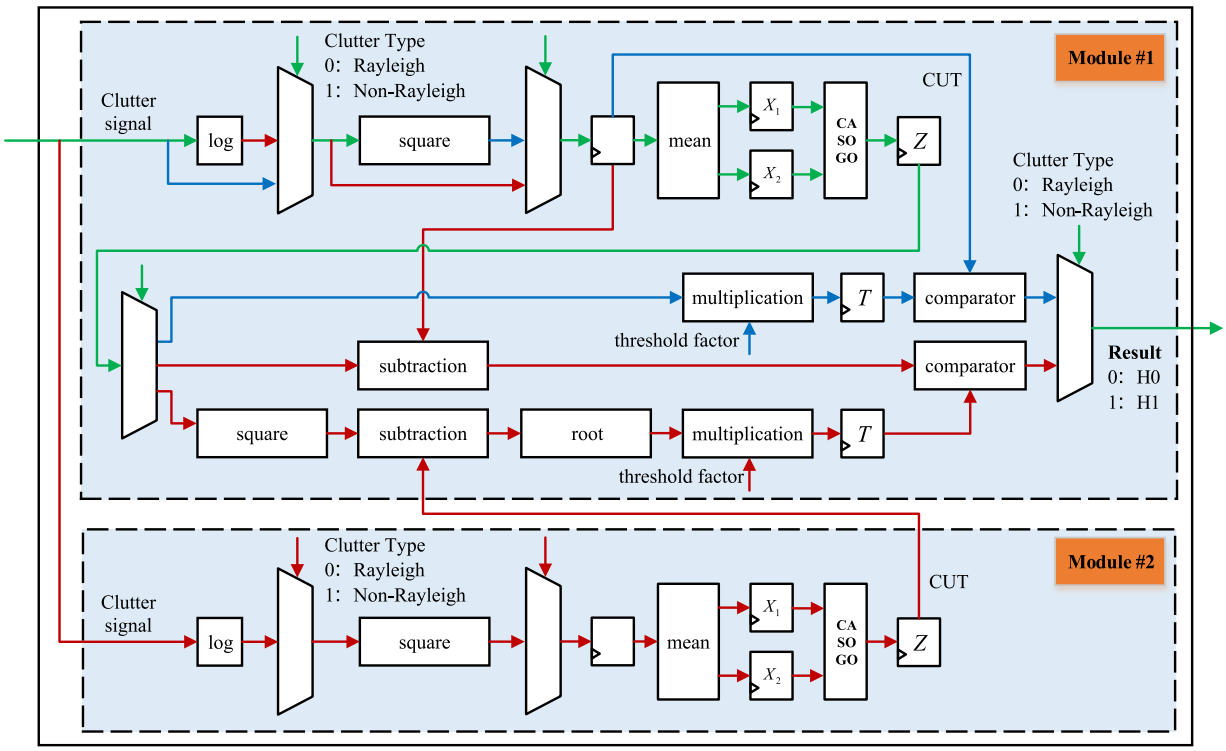


Fig. 2. Block diagram of the proposed reconfigurable architecture for two types of ML and log-t CFAR detectors. As a part of the sharing, the green paths cooperate with other color paths to implement different detectors. The proposed architecture will be configured as an ML detector when the blue paths work, whereas it will be a log-t detector when the magenta paths work.

distributions to obtain the optimal constant false alarm performance.

3. Proposed CFAR detector architecture

In many studies, when integrating several CFAR detectors in a single architecture, it is common to implement each detector separately on the FPGA board. Such a naive-integration structure is not conducive to make full use of hardware resources, and the complicated layout will cost more area overhead. From the above analysis in Section 2.2, although the strategies of constructing the test statistics in ML and log-t CFAR detectors are significant different, there are still partial operations that are superimposed. In this letter, a multiplexing and reconfigurable architecture combining these two types of CFAR detectors is proposed for the purpose of saving hardware resources and raising hardware versatility.

As illustrated in Fig. 2, the block diagram of the proposed CFAR detector architecture contains two modules, i.e. Module #1 and Module #2. In this architecture, the main components consist of shift register, mean calculation module, parallel addition, subtraction, multiplication, logarithm, comparator, and controlling unit, etc. Through corresponding parameter configurations, the proposed architecture is available to achieve both the ML and log-t CFAR detectors by different colour paths.

For a Rayleigh distributed clutter, a conventional ML CFAR detector can be implemented only by the Module #1 from the proposed architecture. As shown in the Module #1, the input data stream of the configured ML detector will be transmitted along the green and blue paths, whereas the magenta paths are set to be dormant states. To be specific, the clutter signal is preprocessed by the square law and then passed into the reference window continuously. Arranged on the both sides of CUT and guard cells as in Fig. 1, the reference window is divided into two halves, termed as a leading window and a lagging window, with the length of $N/2$. By calculating the reference cells in the leading and lagging windows, the corresponding mean value of each window can be obtained as X_1 and X_2. Referring to an ML detector from CA, GO and SO algorithms, the estimated clutter background level is computed by the mathematical expressions in Eqs. (6) to (8) respectively. After that, the mean value Z will be multiplied by a configured threshold factor and compared with a threshold value to complete the detection process in Eq. (5).

For a non-Rayleigh clutter obeying Weibull, Log-normal, or K distribution, two modules of Module #1 and Module #2 can be associated to achieve a log-t CFAR detector. At this moment, the input data stream will pass along the green and magenta paths and the blue paths will be invalid. As shown in Fig. 2, the process of the mean calculation is basically similar to the ML detector. Differently, the log-t CFAR detector requires two different branches to compute the arithmetic mean and standard deviation of the reference cells respectively according to Eq. (9). Specifically, as the denominator on the left side of Eq. (9), the statistic of the standard deviation can be acquired with the Module #2 and the magenta paths at the bottom row of the Module #1. Here, the output of the Module #2 will be linked as an input to the subtraction block in the Module #1. The other magenta paths in the Module #1 are utilized to calculate the statistic of the

arithmetic mean for the log-t CFAR detector, which is a part of the numerator on the left side of Eq. (9). Finally, passing through the comparator, the detection result can be obtained as a binary value of H_1 or H_0.

Stated thus, the proposed architecture integrates two types of ML and log-t CFAR detectors for different clutter distributions, motivating by the features of the test statistics. Moreover, it is also parameterizable and reconfigurable with regard to the number of reference cells and guard cells, the value of threshold factors, and the probability of desired false alarms. In the application process, it is practicable and convenient to examine the CFAR performance of ML and log-t detectors through real-time configuration of the parameters.

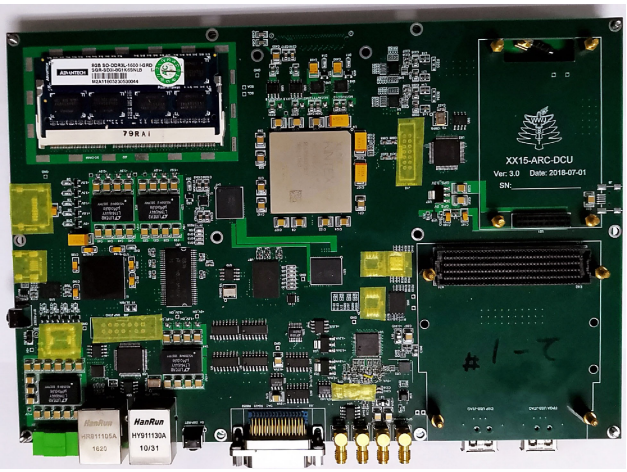


Fig. 3. Photograph of the used prototyping FPGA board.

4. FPGA-based implementations and results

High-performance FPGA facilitates the design and hardware implementation of the proposed CFAR detector architecture. To validate the effectiveness and applicability, the proposed architecture is developed by Verilog hardware description language (HDL) and synthesized on an Xilinx Kintex-7 XC7K325T FPGA device as Fig. 3.

As shown in Fig. 2, an ML CFAR detector can be realized by instantiating the Module #1, while a log-t CFAR detector can be accomplished by instantiating the Module #1 and Module #2 simultaneously. For both types of detectors, 32 reference cells and 8 guard cells are configured by default for the proposed architecture, which is a common configuration for the most radar applications with a good trade-off between the performance and accuracy. Moreover, the input data are defined to 32-bit unsigned values and the precision of the internal temporal data during operations is 40 bits with 8-bit decimal parts.

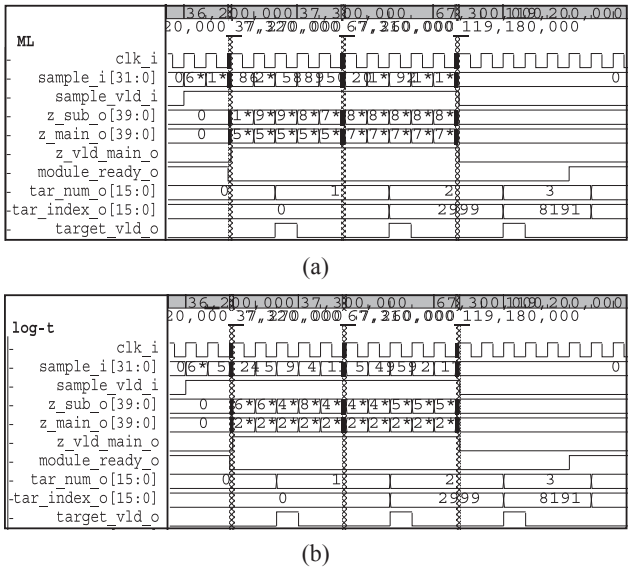
To validate the effectiveness of the proposed CFAR detector architecture, its synthesis result is compared with several approaches, including the single ML detector, single log-t detector, and ML&log-t detector which naively integrates two types of CFAR processors without multiplexing structures. Table I recapitulates the resource utilization performance of the FPGA hardware for these approaches. As shown, the single ML detector takes 18% of look up tables (LUTs), 2.3% of flip-flops and 0.67% of block random access memory (BRAM), whereas the values of the single log-t detector are 47%, 5.6%, and 1.1% respectively. For the naive-integration approach of ML&log-t detector, it consumes 66% of LUTs, 8% of flip-flops and 2.0% of BRAM, which are almost the sum of two single detectors in resource consumption. However, our proposed architecture requires only 48% of LUTs, 5.9% of flip-flops and 1.1% of BRAM. If the attention is paid to the dominant LUTs, it can be inferred that although it expends about 30% more than the single ML detector and 1% more than the single log-t detector, the proposed architecture saves approximately 18% than the ML&log-t detector. This saving is appreciable for the limited hardware resource of XC7K325T FPGA board. Moreover, all these approaches can achieve a maximum operating frequency of 100 MHz for 32 reference cells and 8 guard cells.

Table I. Synthesis results of the CFAR architectures

Resources	Utilization	Ratio
single ML detector		
Number of LUTs	37005	18%
Number of flip-flops	10727	2.3%
Number of BRAM	3	0.67%
single log-t detector		
Number of LUTs	96249	47%
Number of flip-flops	25834	5.6%
Number of BRAM	5	1.1%
ML&log-t detector		
Number of LUTs	134924	66%
Number of flip-flops	36610	8%
Number of BRAM	9	2.0%
Proposed architecture		
Number of LUTs	99650	48%
Number of flip-flops	27082	5.9%
Number of BRAM	5	1.1%

To confirm the detection performance of the proposed CFAR architecture, MATLAB is employed to generate the experimental clutter that obeys Rayleigh or non-Rayleigh distributions. Suppose the size of the clutter data is 1×8192, and the average power is 20 dB. Three Swerling II targets are embedded in the clutter with the same signal to clutter ratio (SCR) of 35 dB, located in the range units of (1,1), (1,3000), and (1,8192) separately. After the input clutter data are produced as floating-point form by MATLAB, they must first be converted to fixed-point values to accommodate the hardware processing. When performing the hardware simulations, the previously generated data are read and subjected to the proposed architecture for CFAR detection. The desired false alarm probability is prescribed to 10^{-6}, and the number of reference cells and guard cells are configured according to the default values of 32 and 8, respectively.

Fig. 4 depicts the emulation timing results of the two types of ML and log-t CFAR detectors. As seen in each subfigure, clk_i is the input clock signal, sample_i is the original input clutter signal, z_sub_o and z_main_o are interior temporary variables, sample_vld_i, z_vld_main_o,



(a)

(b)

Fig. 4. Timing result graphs of the configured detectors from the proposed architecture. (a) ML detector. (b) Log-t detector.

module_ready_o and target_vld_o are controlling signals. Besides, tar_num_o and tar_index_o declare the number and range coordinate of the detected targets by each CFAR detector, respectively. Specifically in Fig. 4(a), since the three ML CFAR detectors (CA, GO, and SO) share the same implementation paths in Fig. 2 except for the slightly different selection strategies in Eqs. (6) to (8), the detection results of such three detectors are consistent in uniform background. From the signals of tar_num_o and tar_index_o in Figs. 4(a) and 4(b), it can be observed that both types of the configured ML and log-t detectors can accurately scout the number and location of the three targets without additional false alarms. This illustrates the correctness of the proposed CFAR detector architecture for the different clutter distributions on FPGA.

Table II. Comparison of operation cycle and time for each detector

Detector	Cycle	Runtime (μs)
CA-CFAR	8303	83
GO-CFAR	8303	83
SO-CFAR	8303	83
log-t CFAR	8303	83

Furthermore, Table II records the number of clock cycles and the operation time for each configured detector with different parameters. Using the inherent parallelism and pipelining techniques, the proposed architecture requires only 8303 cycles to process a radar clutter of 8192 samples. At the maximum operating frequency of 100 MHz, it implies that the processing time to perform CFAR detections is approximately 83 μs within 32 reference cells and 8 guard cells, which satisfies the real-time requirements of practical engineering well.

5. Conclusion

In this letter, a novel hardware architecture for radar target CFAR detection is presented to reduce resource consumption and area overhead. The proposed architecture is compatible with two types of ML and log-t detectors, which are optimal to the Rayleigh and non-Rayleigh distributions respectively. It is also reconfigurable in terms of the number of reference cells and guard cells, the value of threshold factors, as well as the desired false alarm probability. After programmed by Verilog HDL and implemented on the XC7K325T FPGA board, the synthesis results reveal that the proposed architecture can occupy 18% less hardware resources than the naive-integration approach and operate up to 100 MHz by exploiting parallel computing and pipeline design. By virtue of the clutter data generated by MATLAB, the proposed architecture is examined and verified for detection performance in different clutter distributions. From the experimental results, it is evident that both the ML and log-t CFAR detectors require only 83 μs to deal with 8192 samples for detecting the number and location of targets successfully. As a future work, the proposed architecture will be integrated into a complete radar system for automatic target detection.

References

[1] W. Zhao, *et al.*: "Maximum eigenvalue matrix CFAR detection using pre-processing in sea clutter," IEEE Access **7** (2019) 91414 (DOI: 10.1109/ACCESS.2019.2928003).
[2] Y. Liu, *et al.*: "Research on a new comprehensive CFAR (Comp-CFAR) processing method," IEEE Access **7** (2019) 19401 (DOI: 10.1109/ACCESS.2019.2897358).
[3] Z. Z. Wang, *et al.*: "Adaptive GLR-, Rao- and Wald-based CFAR detectors for a subspace signal embedded in structured Gaussian interference," Digit. Signal Prog. **92** (2019) 139 (DOI: 10.1016/j.dsp.2019.05.010).
[4] A. Pourmottaghi, *et al.*: "A CFAR detector in a nonhomogenous Weibull clutter," IEEE Trans. Aerosp. Electron. Syst. **48** (2012) 1747 (DOI: 10.1109/TAES.2012.6178094).
[5] G. V. Weinberg: "Formulation of a generalised switching CFAR with application to X-band maritime surveillance radar," SpringerPlus **4** (2015) 574 (DOI: 10.1186/s40064-015-1347-2).
[6] G. V. Weinberg, *et al.*: "Development of non-coherent CFAR detection processes in Weibull background," Digit. Signal Prog. **75** (2018) 96 (DOI: 10.1016/j.dsp.2018.01.002).
[7] M. Weiss: "Analysis of some modified cell-averaging CFAR processors in multiple-target situations," IEEE Trans. Aerosp. Electron. Syst. **AES-18** (1982) 102 (DOI: 10.1109/TAES.1982.309210).
[8] G. V. Trunk: "Range resolution of targets using automatic detectors," IEEE Trans. Aerosp. Electron. Syst. **AES-14** (1978) 750 (DOI: 10.1109/TAES.1978.308625).
[9] V. G. Hansen and J. H. Sawyers: "Detectability loss due to Greatest Of selection in a cell-averaging CFAR," IEEE Trans. Aerosp. Electron. Syst. **AES-16** (1980) 115 (DOI: 10.1109/TAES.1980.308885).
[10] H. Rohling: "Radar CFAR thresholding in clutter and multiple target situations," IEEE Trans. Aerosp. Electron. Syst. **AES-19** (1983) 608 (DOI: 10.1109/TAES.1983.309350).
[11] Y. He: "Performance of some generalised modified order statistics CFAR detectors with automatic censoring technique in multiple target situations," IEE Proc., Radar Sonar Navig. **141** (1994) 205 (DOI: 10.1049/ip-rsn:19941159).
[12] G. B. Goldstein: "False-alarm regulation in log-normal and Weibull clutter," IEEE Trans. Aerosp. Electron. Syst. **AES-9** (1973) 84 (DOI: 10.1109/TAES.1973.309705).
[13] G. V. Weinberg, *et al.*: "Enhancing Goldstein's log-t detector in pareto-distributed clutter," IEEE Trans. Aerosp. Electron. Syst. **53** (2017) 1035 (DOI: 10.1109/TAES.2017.2665144).
[14] R. Ravid, *et al.*: "Maximum-likelihood CFAR for Weibull

background," IEE Proc., F, Radar Signal Process. **139** (1992) 256 (DOI: 10.1049/ip-f-2.1992.0033).
[15] M. Guida, *et al.*: "Biparametric linear estimation for CFAR against Weibull clutter," IEEE Trans. Aerosp. Electron. Syst. **28** (1992) 138 (DOI: 10.1109/7.135440).
[16] R. Djemal, *et al.*: "A novel hardware/software embedded system based on automatic censored target detection for radar systems," AEU Int. J. Electron. Commun. **67** (2013) 301 (DOI: 10.1016/j.aeue.2012.09.001).
[17] W. Zhou, *et al.*: "Robust CFAR detector with weighted amplitude iteration in nonhomogeneous sea clutter," IEEE Trans. Aerosp. Electron. Syst. **53** (2017) 1520 (DOI: 10.1109/TAES.2017.2671798).
[18] V. A. Aalo, *et al.*: "Performance of CA-CFAR detectors in nonhomogeneous positive alpha-stable clutter," IEEE Trans. Aerosp. Electron. Syst. **51** (2015) 2027 (DOI: 10.1109/TAES.2015.140043).
[19] S. Chen, *et al.*: "A new CFAR algorithm based on variable window for ship target detection in SAR images," Signal Image Video Process. **13** (2019) 779 (DOI: 10.1007/s11760-018-1408-4).
[20] S. Simic, *et al.*: "An FPGA based implementation of a CFAR processor applied to a pulse-compression radar system," Radio-engineering **23** (2014) 73.
[21] R. Cumplido, *et al.*: "A configurable FPGA-based hardware architecture for adaptive processing of noisy signals for target detection based on constant false alarm rate (CFAR) algorithms," GlobalSIP (2004) 214.
[22] C. Torres-Huitzil, *et al.*: "Design and implementation of a CFAR processor for target detection," FPL **3203** (2004) 943 (DOI: 10.1007/978-3-540-30117-2_104).
[23] B. Wei, *et al.*: "Adaptive PN code acquisition in multi-path spread spectrum communications using FPGA," ISSCS **2** (2007) 1 (DOI: 10.1109/ISSCS.2007.4292790).
[24] J. K. Ali, *et al.*: "An FPGA-based implementation of CA-CFAR processor," Asian J. Inf. Technol. **6** (2007) 511.
[25] A. M. Alsuwailem, *et al.*: "Design and implementation of a configurable real-time FPGA-based TM-CFAR processor for radar target detection," J. Active Passive Electr. Devices **3** (2008) 241.
[26] S. Lopez-Estrada, *et al.*: "Hardware architecture for adaptive filtering based on energy-CFAR processor for radar target detection," IEICE Electron. Express **7** (2010) 628 (DOI: 10.1587/elex.7.628).
[27] R. Djemal, *et al.*: "An adaptive CFAR embedded system architecture for target detection," Des. Autom. Embed. Syst. **17** (2013) 109 (DOI: 10.1007/s10617-013-9121-6).
[28] V. K. Bharti and V. Patel: "Realization of real time adaptive CFAR processor for homing application in marine environment," SPACES (2018) 185 (DOI: 10.1109/SPACES.2018.8316342).
[29] C. H. Gowda, *et al.*: "Performance of distributed CFAR test under various clutter amplitudes," IEEE Trans. Aerosp. Electron. Syst. **35** (1999) 1410 (DOI: 10.1109/7.805457).
[30] V. Anastassopoulos, *et al.*: "Optimal CFAR detection in Weibull clutter," IEEE Trans. Aerosp. Electron. Syst. **31** (1995) 52 (DOI: 10.1109/7.366292).
[31] P. Shui, *et al.*: "Shape-parameter-dependent coherent radar target detection in K-distributed clutter," IEEE Trans. Aerosp. Electron. Syst. **52** (2016) 451 (DOI: 10.1109/TAES.2015.140109).
[32] A. Jakubiak: "False-alarm probabilities for a log-t detector in K-distributed clutter," Electron. Lett. **19** (1983) 725 (DOI: 10.1049/el:19830493).
[33] N. Detouche and T. Laroussi: "Extensive Monte Carlo simulations for performance comparison of three non-coherent integrations using log-t-CFAR detection against Weibull clutter," SETIT (2012) 726 (DOI: 10.1109/SETIT.2012.6482004).