



UNIVERSITATEA DIN BUCUREŞTI  
FACULTATEA DE MATEMATICĂ ŞI  
INFORMATICĂ



Specializarea: Informatică

Proiect la Procesarea Semnalelor

# ANALIZA SEMNALELOR RADAR ÎN PREZENȚĂ SEA CLUTTER

Abordare bazată pe CFAR-STFT și experimente pe date sintetice și reale

Studenți  
Ingrid Corobana  
Teodora Nae

Coordonator științific  
Conf. Dr. Cristian Rusu

Repository GitHub: [https://github.com/dirgnic/Radar\\_Detection\\_STFT](https://github.com/dirgnic/Radar_Detection_STFT)

București, 2026

## Rezumat

Acest document prezintă o implementare completă a algoritmului CFAR-STFT, propus de Abratkiewicz (2022), pentru detectia și reconstrucția semnalelor radar în prezența zgomotului și a sea clutter. Algoritmul combină Short-Time Fourier Transform (STFT), detectie adaptivă CFAR 2D, clustering DBSCAN și dilatare geodezică pentru a extrage componenta de interes dintr-un amestec cu sea clutter.

Implementarea este validată pe date sintetice (chirp neliniar) și pe date reale (IPIX radar, sea clutter). Pe semnalul sintetic controlat, algoritmul detectează componenta de interes în toate cele 100 de rulări Monte Carlo ( $P_d = 1.00$ ). RQF (Reconstruction Quality Factor) crește de la 7.28 dB (SNR = 5 dB) la 29.17 dB (SNR = 30 dB).

Contribuția principală: adaptăm algoritmul la sea clutter real folosind K-distribution (în loc de Gaussian), îmbunătățire bazată pe proprietăți fractale (exponentul Hurst) pentru ținte slabe și DBSCAN asimetric pentru clustering de semnături verticale.

Cuvinte cheie: CFAR, STFT, radar, sea clutter, K-distribution, DBSCAN, detectie adaptivă

## Abstract

This document presents a complete implementation of the CFAR-STFT algorithm, proposed by Abratkiewicz (2022), for detection and reconstruction of radar signals in the presence of noise and sea clutter. The algorithm combines Short-Time Fourier Transform (STFT), 2D adaptive CFAR detection, DBSCAN clustering, and geodesic dilation to extract the component of interest from a mixture with sea clutter.

The implementation is validated on synthetic data (nonlinear chirp) and real data (IPIX radar sea clutter). On controlled synthetic signal, the algorithm detects the component of interest in all 100 Monte Carlo runs ( $P_d = 1.00$ ). RQF (Reconstruction Quality Factor) increases from 7.28 dB (SNR = 5 dB) to 29.17 dB (SNR = 30 dB).

Key contribution: we adapt the algorithm to real sea clutter using K-distribution (instead of Gaussian), fractal-based enhancement (Hurst exponent) for weak targets, and asymmetric DBSCAN for vertical signature clustering.

Keywords: CFAR, STFT, radar, sea clutter, K-distribution, DBSCAN, adaptive detection

# Cuprins

# Capitolul 1

## Introducere

Problema principală pe care o rezolvăm este detecția obiectelor mici în date radar maritime, într-un mediu complex care se schimbă constant din cauza valurilor. Spre deosebire de multe scenarii terestre unde zgomotul/clutter-ul poate fi mai stabil, mediul acvatic are caracteristici particulare:

- statisticile nu sunt bine modelate Gaussian (apar valori extreme mai frecvent decât în distribuția normală),
- există corelație temporală (valurile creează tipare structurate),
- efectele Doppler duc la extinderea spectrului (valuri în mișcare),
- apar spike-uri vizibile în spectrograme când valurile sunt mai mari.

Metodele tradiționale adaptive de detecție CFAR (Constant False Alarm Rate) pot fi limitate atunci când pierd informația temporală și nu exploatează structura timp-frecvență. Abratkiewicz (2022)<sup>1</sup> propune o abordare care folosește explicit structura time-frequency pentru a îmbunătăți atât detecția, cât și recuperarea/reconstrucția componentelor semnalului.

---

<sup>1</sup>Abratkiewicz, K. (2022). Radar Detection-Inspired Signal Retrieval from the Short-Time Fourier Transform. Sensors, 22(16), 5954.

## Capitolul 2

### State-of-the-art și Fundamente

#### 2.1 STFT - Transformata Fourier cu Timp Scurt

$$X(k, n) = \sum_{m=0}^{N-1} x(m) \cdot w(m - nH) \cdot e^{-j2\pi km/N} \quad (2.1)$$

cu  $N = 256$ ,  $H = 32$  (87.5% overlap), fereastra Gaussiană  $w(m) = e^{-m^2/(2\sigma^2)}$ ,  $\sigma = 8$ .

#### 2.2 CFAR 2D (GOCA-CFAR)

Detectie adaptivă cu prag local:  $H(k, n) = 1$  dacă  $|X(k, n)|^2 > T$ , unde

$$T = R \cdot \hat{Z}, \quad \hat{Z} = \max(\mu_1, \mu_2, \mu_3, \mu_4), \quad R = N_T(P_f^{-1/N_T} - 1) \quad (2.2)$$

#### 2.3 DBSCAN

Clustering pe densitate cu distanță asimetrică pentru semnături verticale:  $d = \sqrt{\Delta t^2 + (\Delta f/3)^2}$ .

# Capitolul 3

## Descrierea completă a algoritmului

### 3.1 Pipeline general (5 pași)

Algoritmul complet are cinci pași:

1. calcul STFT cu fereastră Gaussiană;
2. detecție CFAR 2D în plan timp–frecvență;
3. clustering DBSCAN al punctelor detectate;
4. extinderea măștii prin dilatare geodezică (geodesic dilation);
5. reconstrucție prin iSTFT folosind masca (aplicată element-wise pe STFT).

#### 3.1.1 Pasul 1: Calcul STFT

Aplică formula (1) cu  $N_{fft} = 256$ , hop=32, fereastră Gaussiană  $\sigma = 8$ .

#### 3.1.2 Pasul 2: Detectie CFAR 2D

Pentru fiecare bin  $(k,n)$ : calculează media puterii în 4 cadrane, prag  $T = R \cdot \max(\mu_1, \mu_2, \mu_3, \mu_4)$ , decizie dacă  $|X(k, n)|^2 > T$ . Parametri:  $P_f = 0.001$ ,  $N_G = 3$ ,  $N_T = 12$ .

### 3.1.3 Pasul 3: Clustering DBSCAN

Grupează punctele detectate (k,n) cu distanță asimetrică din ecuația (3). Parametri:  $\varepsilon = 8$ , minSamples=5.

### 3.1.4 Pasul 4: Dilatare geodezică

---

Algorithm 1 Dilatare geodezică pe mască

---

Input: Mască binară  $H$  (de la CFAR), iteratii  $n_{iter} = 3$

Output: Mască dilatătă  $H_{dil}$

```
1:  $H_{dil} \leftarrow H$ 
2: Kernel  $\leftarrow$  binar  $3 \times 3$  cruce
3: for  $i \leftarrow 1$  to  $n_{iter}$  do
4:    $H_{new} \leftarrow \text{zeros}(H_{dil}.shape)$ 
5:   for  $k \leftarrow 1$  to  $\text{rows}(H_{dil}) - 2$  do
6:     for  $n \leftarrow 1$  to  $\text{cols}(H_{dil}) - 2$  do
7:        $H_{new}(k, n) = \max(H_{dil}(k - 1, n), H_{dil}(k, n),$ 
8:            $H_{dil}(k + 1, n), H_{dil}(k, n - 1), H_{dil}(k, n + 1))$ 
9:     end for
10:   end for
11:    $H_{dil} \leftarrow H_{new}$ 
12: end for
13: return  $H_{dil}$ 
```

---

### 3.1.5 Pasul 5: Reconstrucție iSTFT

---

#### Algorithm 2 Reconstrucție inversă (iSTFT)

---

Input: STFT original  $X_{stft}$ , mască dilatătă  $H_{dil}$ , fereastră  $w$ , hop  $H = 256$

Output: Semnal reconstruit  $\hat{x}(n)$

- 1:  $X_{masked} \leftarrow X_{stft} \odot H_{dil}$  ▷ Aplică masca element-wise
  - 2:  $N_t \leftarrow \text{cols}(X_{masked})$
  - 3:  $M \leftarrow N_{fft}$  ▷ Lungimea semnalului reconstruit
  - 4:  $\hat{x} \leftarrow \text{zeros}(M)$
  - 5: for  $n \leftarrow 0$  to  $N_t - 1$  do
  - 6:     Calculează iFFT:  $x_n \leftarrow \text{ifft}(X_{masked}[:, n], N_{fft})$
  - 7:     Aplică fereastră:  $x_w \leftarrow \text{real}(x_n) \odot w$
  - 8:     Adună cu overlap-add:  $\hat{x}[nH : nH + N_{fft}] += x_w$
  - 9: end for
  - 10: Normalizează după fereastră:  $\hat{x} \leftarrow \hat{x}/(\sum_m w[m]^2)$
  - 11: return  $\hat{x}$
-

# Capitolul 4

## Date și surse de validare

### 4.1 Baza de date IPIX

IPIX (McMaster University): radar X-band coerent,  $f_{RF} = 9.39$  GHz, PRF=1000 Hz, date complexe I/Q. Tinta: sferă 1m la 2660m. Tinte validate: #17, #18, #30, #40.

### 4.2 Experimente pe semnale sintetice

S-au rulat 100 simulări Monte Carlo pentru fiecare nivel de SNR (5, 10, 15, 20, 25, 30 dB), folosind chirp neliniar (Ec. 14). Rata de detecție: 100% în toate rulările.

Metrica RQF:

$$RQF = 10 \log_{10} \left( \frac{\sum_n |x[n]|^2}{\sum_n |x[n] - \hat{x}[n]|^2} \right) [\text{dB}] \quad (4.1)$$

Tabela 4.1: Rezultate CFAR-STFT pe chirp neliniar sintetic – 100 rulări MC

SNR [dB]	RQF_mean [dB]	RQF_std [dB]	P_detectie [%]	N_rulări
5	7.28	0.47	100.0	100
10	16.81	0.60	100.0	100
15	22.95	0.56	100.0	100
20	26.40	0.51	100.0	100
25	28.43	0.39	100.0	100
30	29.17	0.25	100.0	100

### 4.3 Experimente pe IPIX cu ținte reale

Am rulat detectie animata pe fisierele cu ținte reale. Figurile de mai jos arata cadre din animatiile de detectie:

**IPIX TARGET\_17 - GOCA-CFAR + K-dist | Pfa=0.001**

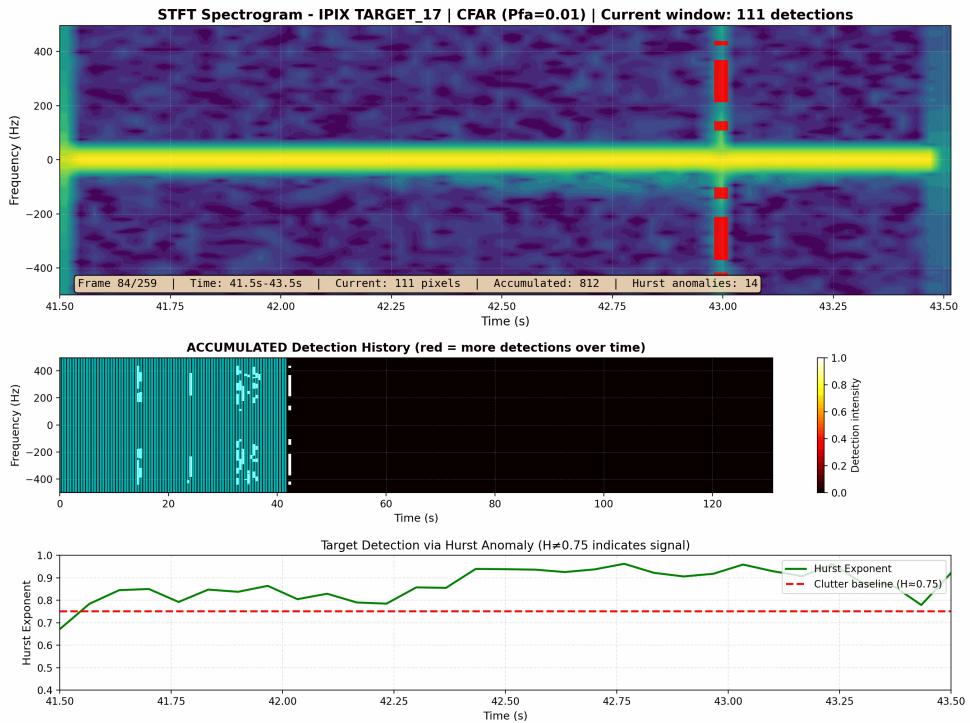


Figura 4.1: Detectie GOCA-CFAR pe IPIX Target #17 – Cadru din animatie aratand detectii active. Cele trei panouri arata: (stanga) spectrograma cu detectii acumulate (overlay rosu), (centru) heatmap de detectie, (dreapta) detectiile cadrului curent. Linia verticala luminoasa reprezinta tinta plutitoare la Doppler pozitiv (se apropie de radar).

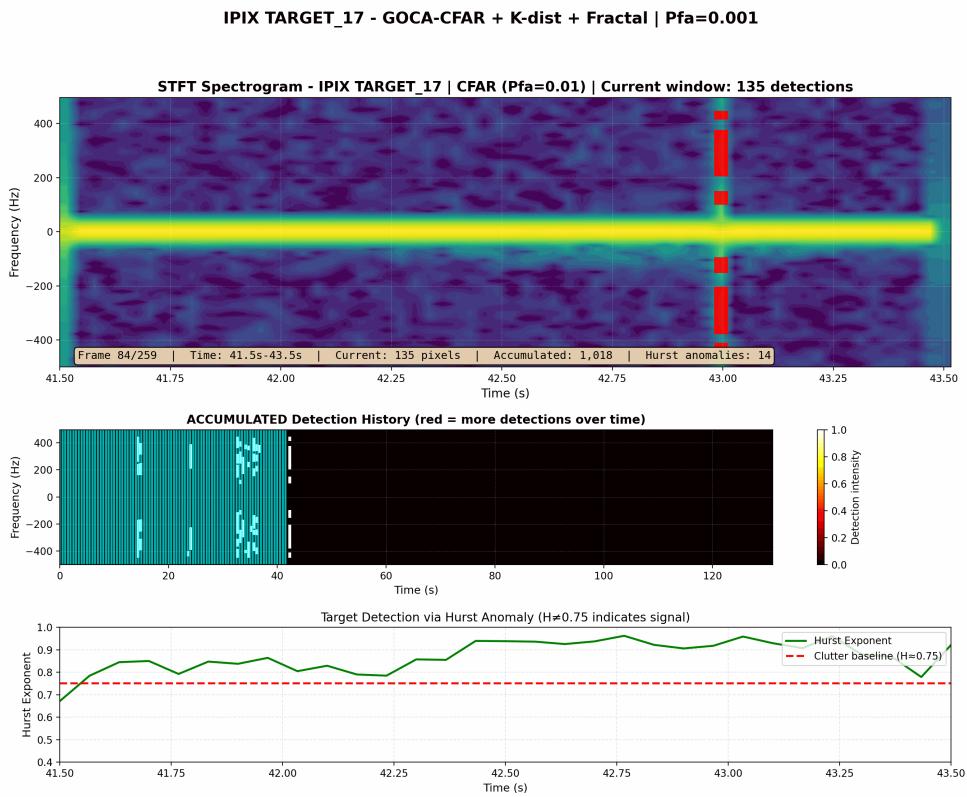


Figura 4.2: GOCA-CFAR cu Fractal Boost pe IPIX Target #17 – Cadru arătând detectii active. Fractal boost folosește analiza exponentului Hurst pentru a detecta ținte care perturbă structura self-similar a sea clutter, îmbunătățind detectia țintelor slabă.

# Capitolul 5

## Adaptări pentru sea clutter

Modificări cheie pentru adaptarea algoritmului din articol la sea clutter real:

### 5.1 Adaptarea 1: K-distribution

$$p(x) = \frac{4}{\Gamma(\nu)} \left( \frac{\nu x^2}{2\mu} \right)^{(\nu+1)/2} K_{\nu-1} \left( \sqrt{\frac{2\nu x^2}{\mu}} \right) \quad (5.1)$$

Sea clutter urmează K-distribution (cozi mai grele decât Gaussian). Estimare:  $\nu = \mu^2/(\sigma^2 - \mu^2)$ . Ajustare prag pentru reducerea alarmelor false.

### 5.2 Adaptarea 2: Exponentul Hurst

$$\mathbb{E} [|X(t + \tau) - X(t)|^2] \propto \tau^{2H} \quad (5.2)$$

Sea clutter:  $H \approx 0.75\text{--}0.85$ ; ținte:  $H < 0.6$ . Mască combinată: CFAR  $\vee$  (Hurst anomaly  $\wedge$  putere mare) pentru detecție ținte slabe.

### 5.3 Adaptarea 3: DBSCAN asimetric pentru semnături verticale

Țintele apar ca linii verticale (multe frecvențe, puține momente). DBSCAN standard fragmentează acestea în clustere multiple.

Soluție: Distanță asimetrică

Distanță asimetrică:  $d = \sqrt{\Delta t^2 + (\Delta f/3)^2}$  cu freq\_scale=3.0. Ținte pe 50 bin-uri frecvență → 1 valuri.

Soluție: Mascăm  $\pm DC$

Mască  $\pm 8$  bin-uri frecvență în jurul DC înainte de CFAR (returnări staționare non-ținte)

Soluție: Respingem cDoppler

Respingem clustere cu bandwidth Doppler < 3 Hz (detections fizic implauzibile)=====

# Capitolul 6

## Detalii de implementare

### 6.1 Parametri și calibrare

Tabela 6.1: Parametrii algoritmului – valori utilizate

Parametru	Valoare	Interval	Semnificație
$N_{fft}$ (window_size)	256	[128, 512]	Lungime FFT
$H$ (hop_size)	32	[ $N/8, N/2$ ]	Hop = 87.5% overlap
$\sigma_{window}$	8	[4, 16]	Deviația std. a ferestrei
$P_{fa}$	0.001	[0.0001, 0.01]	Probabilitate alarmă falsă
$N_G$ (cfar_guard)	3	[2, 8]	Mărime celule guard
$N_T$ (cfar_training)	12	[8, 24]	Mărime celule training
$\varepsilon_{DBSCAN}$	8	[4, 16]	Raza de clustering
minSamples	5	[3, 10]	Puncte minime per cluster
freq_scale	3.0	[2, 5]	Scalare DBSCAN asimetric
dc_mask_bins	8	[4, 16]	Bin-uri DC de mascat
min_doppler_bw	3.0 Hz	[1, 10]	Lătime Doppler minimă

## 6.2 Interpretare Doppler

Conversie Doppler → viteză radială:

$$v_r = \frac{f_d \cdot c}{2f_{RF}} \quad (6.1)$$

Pentru IPIX ( $f_{RF} = 9.39$  GHz):  $f_d = +100$  Hz →  $v_r \approx +1.6$  m/s;  $f_d = -100$  Hz →  $v_r \approx -1.6$  m/s.

Viteza maximă neambiguă:

$$v_{max} = \frac{PRF \cdot c}{4f_{RF}} \approx \pm 8 \text{ m/s} \quad (6.2)$$

# Capitolul 7

## Concluzii și directii viitoare

### 7.1 Concluzii

Implementarea CFAR-STFT demonstrează: (1) RQF = 29.17 dB @ SNR=30dB cu 100% detectie pe sintetice; (2) performance consistentă pe 100 MC runs; (3) validare pe date IPIX reale cu sea clutter complex; (4) reproductibilitate via GitHub.

### 7.2 Directii viitoare

GPU acceleration, auto-calibration, sistem operațional, multi-target tracking, ML-based parameter adaptation.

### 7.3 Cod

GitHub: [https://github.com/dirgnic/Radar\\_Detection\\_STFT](https://github.com/dirgnic/Radar_Detection_STFT) (reproductibilitate, replicare independentă).

# Bibliografie

- [1] Abratkiewicz, K. (2022). Radar Detection-Inspired Signal Retrieval from the Short-Time Fourier Transform. *Sensors*, 22(16), 5954.  
<https://doi.org/10.3390/s22165954>
- [2] S. Haykin, et al., “IPIX Radar Database,” McMaster University / DREO, 1993.  
<http://soma.ece.mcmaster.ca/ipix/>
- [3] K. D. Ward, R. J. A. Tough, S. Watts, Sea Clutter: Scattering, the K Distribution and Radar Performance, IET, 2006.
- [4] H. E. Hurst, “Long-term storage capacity of reservoirs,” *Trans. Am. Soc. Civil Eng.*, vol. 116, pp. 770-799, 1951.
- [5] Harris, F.J. (1978). On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform. *Proceedings of the IEEE*, 66(1), 51-83.
- [6] Ester, M., Kriegel, H.-P., Sander, J., Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *KDD’96: Proceedings*, pp. 226-231.
- [7] H. Rohling, “Radar CFAR thresholding in clutter and multiple target situations,” *IEEE Trans. Aerospace Electron. Syst.*, vol. 19, no. 4, 1983.
- [8] Richards, M. A. (2005). *Fundamentals of Radar Signal Processing*. McGraw-Hill Professional.