

# Analiza semnalelor radar în prezența ecourilor marine

Abordare CFAR-STFT cu adaptări pentru sea clutter real

Ingrid Corobana    Teodora Nae

Universitatea din București  
Facultatea de Matematică și Informatică  
Coordonator: Prof. Dr. Cristian Rusu

Procesarea Semnalelor – 2026



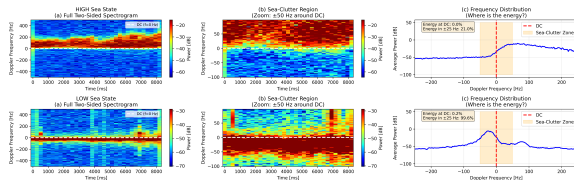
# Problema și Soluția CFAR-STFT

## Sea Clutter:

- Ecouri radar de la suprafața mării
- Non-Gaussian (spike-uri frecvente)
- Ascunde ținte mici

## Pipeline CFAR-STFT:

- 1 STFT → spectrogramă
- 2 CFAR 2D → detecție adaptivă
- 3 DBSCAN → clustering
- 4 Dilatare + iSTFT → reconstrucție

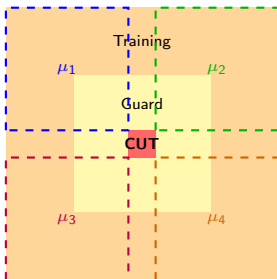


**SEA-CLUTTER EXPLANATION:**

- Roșu pe mijloc DC (Full Hz) = ariea slabă de la suprafața mării
- Galben/Verde în jurul DC = clutter-ul maritim în regiune (valuri, spumă)
- Albăstru lateral = spațiu termic (uniform, rădă)
- "Dotted" = energia este concentrată în anumite zone timp-frecvență (ridges)
- Ținte reale ar apărea ca benzi clare departe de DC ( $\pm 100$ - $400$  Hz)



# GOCA-CFAR 2D – Detecție Adaptivă



**GOCA = Greatest-Of Cell Averaging**

$$\hat{Z} = \max(\mu_1, \mu_2, \mu_3, \mu_4)$$

$$T = R \cdot \hat{Z}$$

$$R = N_T(P_f^{-1/N_T} - 1)$$

**Decizie:**

$$|X(k, n)|^2 \geq T \Rightarrow \text{Detectat}$$

## Parametri IPIX

$N_G = 3$ ,  $N_T = 12$ ,  $P_f = 0.001$  (mult mai mic decât paper!)



# Adaptări pentru Sea Clutter Real

## 1. K-Distribution:

- Sea clutter  $\neq$  Gaussian
- Pondere mai mare în cozi
- Ajustare prag CFAR

## 2. Hurst Boost:

- Clutter:  $H \approx 0.75\text{--}0.85$
- Țintă perturbă:  $H < 0.6$
- Detectează ținte slabe

## 3. DBSCAN Asimetric:

- Ținte = linii verticale în TF
- Scalare frecvență:  $s_f = 3.0$
- Un cluster per țintă

## 4. Mascare DC:

- $\pm 8$  binuri în jurul 0 Hz
- Elimină returnări staționare



# Rezultate pe Date Sintetice

SNR [dB]	RQF [dB]	Detecție
5	7.28	100%
10	16.81	100%
20	26.40	100%
30	<b>29.17</b>	100%

100 rulări Monte Carlo per SNR

**RQF** = Reconstruction Quality Factor

$$\text{RQF} = 10 \log_{10} \frac{\|x\|^2}{\|x - \hat{x}\|^2}$$

- ✓ Detecție 100%
- ✓ RQF crește cu SNR



# Detectie pe Date IPIX Reale

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

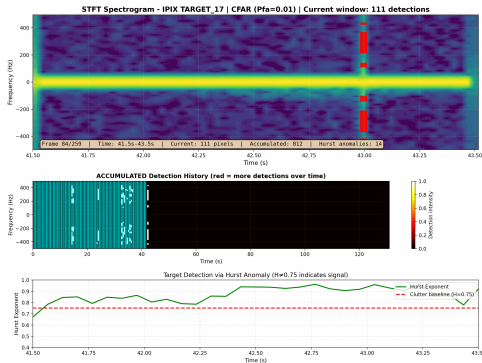


Figura: GOCA-CFAR pe Target #17

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

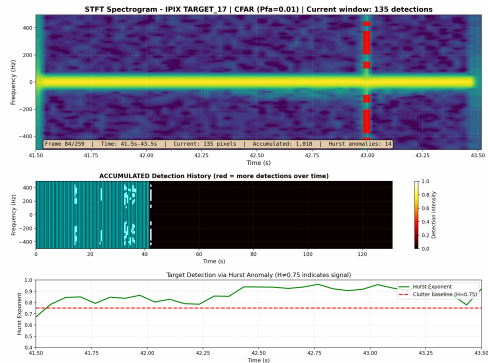


Figura: Cu Fractal Boost



# Comparație Metode CFAR

Metodă	Robustețe clutter	Ținte slabe	Ideal pentru
CA-CFAR	Scăzută	Ridicată	Zgomot omogen
OS-CFAR	Foarte bună	Medie	Multi-target
SOCA-CFAR	Bună	Medie	Margini clutter
GOCA-CFAR	Foarte bună	Bună	Sea clutter

**Observație experimentală:** CA-CFAR eșuează pe IPIX – detectează clutter, nu ținte!  
GOCA-CFAR + DBSCAN = performanța optimă pentru sea clutter neomogen



# Rezultate Comparative pe IPIX

Metodă	Sea State	Componente	Viteză [m/s]
CA-CFAR + HDBSCAN	HIGH	$1.0 \pm 0.0$	-0.05
CA-CFAR + HDBSCAN	LOW	$1.0 \pm 0.0$	-0.01
Triangulare Delaunay	HIGH	4.2	+1.30
Triangulare Delaunay	LOW	13.6	-0.30

## Probleme identificate:

- CA-CFAR: variabilitate zero  $\Rightarrow$  grupează tot clutter-ul într-un cluster
- Triangulare: fragmentare excesivă, inconsistentă între scenarii



- ❶ **Implementare completă** CFAR-STFT în Python
- ❷ **Validare sintetică**: RQF = 29.17 dB @ SNR=30dB, detecție 100%
- ❸ **Validare IPIX**: Detecții consistente pe date reale
- ❹ **Adaptări**: K-distribution, Hurst boost, DBSCAN asimetric
- ❺ **Comparație**: GOCA-CFAR superior CA-CFAR și triangulării

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

Mulțumim! Întrebări?