People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University of Mohamed Boudiaf, M'sila
Faculty of Technology
Department of Electronics

Serial number : .....
Registration number : .....

PhD Dissertation

Submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

Speciality : Electronics & Telecommunications

Authored by

Taha Hocine KERBAA

Title

Estimation and Target CFAR Detection in Sea Clutter with Compound Gaussian Distribution

Publicly defended on february 12th, 2026

Scientific Committee:

President :	KHEDROUCHE Djamel	Professor	University of M'sila, Algeria
Supervisor :	MEZACHE Amar	Professor	University of M'sila, Algeria
Co-Supervisor :	OUDIRA Hocine	Professor	University of M'sila, Algeria
Examiners :	ATIA Salim	Professor	University of Bordj Bou Arréridj, Algeria
	LAROUSSI Toufik	Professor	University of Constantine 1, Algeria
Invited Experts :	Fulvio GINI	Professor	University of Pisa, Italy
	Maria Sabrina GRECO	Professor	University of Pisa, Italy

الملخص

تلعب معالجة إشارات الرادار دورًا محوريًا في أنظمة المراقبة الحديثة، والاستشعار عن بُعد، والتطبيقات الدفاعية، حيث تعتمد دقة كشف الأهداف بدرجة كبيرة على نمذجة التشويش وتقدير معاملاته، خاصة في البيئات البحرية والبرية المعقدة.

تتناول هذه الأطروحة تقنيات متقدمة في نمذجة تشويش الرادار، وتقدير المعاملات، وكشف الأهداف، مع التركيز على دمج منهجيات التعلم العميق ضمن سلسلة المعالجة وتقترح الأطروحة أربع مساهمات رئيسية في مجالات النمذجة، وتقدير المعلمات، والكشف في البيئات المشوشة.

تتمثل المساهمة الأولى في تطوير إطار لاختيار النموذج الإحصائي لتشويش البحر، اعتمادًا على معايير نظرية المعلومات والتحقق المتقاطع، بهدف تحديد التوزيع الأنسب لظروف التشغيل المختلفة.

اهتمت المساهمة الثانية بتقديم مِقدّر هجين يجمع بين الشبكات العصبية الالتفافية (CNN) و شبكات الذاكرة الطويلة قصيرة المدى (LSTM) لتقدير معامل الشكل الخاص بتوزيع K في وجود ضوضاء غاوسية مضافة، مع تحقيق متانة عالية مقارنة بالطرق التقليدية، خصوصًا عند انخفاض نسبة الإشارة إلى الضوضاء (SNR) أو في البيئات غير المستقرة.

المساهمة الثالثة ركزت على تطوير إطار تعلم عميق لتقدير معلمتَي الشكل والمقياس لتوزيع باريتو من النوع الثاني في حالة الارتباط بين العينات، بالاعتماد على دمج الاستخلاص الالتفافي للسمات مع النمذجة المتكررة وبُنى الترميز التلقائي، ما يحقق دقة أعلى وكفاءة حسابية محسّنة.

اخيرا، عالجت المساهمة الرابعة مسألة الكشف الراداري الموزع من خلال صياغة ضبط معاملات الكاشفات اللامركزية بمعدل إنذار كاذب ثابت باستخدام أصغر مستوى (SO-CFAR) و أكبر مستوى (GO-CFAR) كمشكلة تحسين مقيدة وفق معيار نايمان–بيرسون، وحلّها باستخدام خوارزمية تحسين فراشة اللهب (MFO) ، بما يضمن الحفاظ على معدل إنذار كاذب ثابت وتحسين احتمال الكشف في البيئات المتطابقة وغير المتطابقة بين الحساسات.

أُجريت تقييمات عددية مكثفة باستخدام بيانات رادارية مُولّدة وأخرى حقيقية من تجربة رادار معالجة ذكية للبكسلات ضمن نطاق التردد X (IPIX)، وأظهرت النتائج أن دمج مقدّرات التعلم العميق ضمن سلسلة معالجة إشارات الرادار يُمكّن من تطوير أنظمة كشف تكيفية، عالية الدقة، وقادرة على التعامل مع عدم الاستقرار والارتباط في التشويش. وتُمهد هذه الدراسة الطريق نحو أنظمة رادار إدراكية بالكامل تعتمد على التحكم الديناميكي في نمذجة التشويش وتقدير المعلمات وحدود الكشف عبر نماذج قائمة على التعلم.

الكلمات المفتاحية **:** معالجة إشارات الرادار، التعلم العميق، نمذجة التشويش، تقدير المعلمات، كشف الأهداف، CNN-LSTM، CFAR، الذكاء الاصطناعي في الرادار.

Abstract

Radar signal processing plays a critical role in modern surveillance, remote sensing, and defense applications, where accurate target detection based on clutter parameter estimation are crucial in complex sea or land environments. This thesis investigates advanced signal processing techniques for radar clutter modeling, parameter estimation, and target detection, with a particular focus on integrating deep learning methodologies.

The study begins with an in-depth review of sea clutter modeling and Constant False Alarm Rate (CFAR) detection strategies, followed by the introduction of a model selection framework using cross-validation techniques for compound clutter distributions, including K-distributed, Pareto Type II, and Compound Gaussian-Inverse Gaussian (CG-IG) models. This framework allows the radar processor to dynamically select the most appropriate clutter model from real-time data, improving detection robustness.

Building on this foundation, the dissertation presents four main contributions addressing modeling, parameter estimation, and detection in challenging radar clutter environments.

The first contribution investigates statistical model selection for sea clutter, proposing a systematic framework based on information-theoretic criteria and cross-validation strategies to identify the most suitable distribution under varying sea states and operating conditions.

The second contribution introduces a combined Convolutional Neural Network (CNN) with Long Short-Term Memory (LSTM) architecture for estimating the shape parameter of K-distributed clutter in the presence of additive Gaussian noise. This data driven estimator demonstrates improved robustness compared with conventional moment based approaches, particularly in low Signal to Noise Ration (SNR) and nonstationary scenarios.

The third contribution develops a multi-headed deep learning framework for correlated Pareto Type II clutter, enabling joint estimation of shape and scale parameters under correlated returns. By integrating convolutional feature extraction with recurrent modeling and autoencoding structures, the proposed method achieves enhanced estimation accuracy and computational efficiency.

The final contribution addresses distributed radar detection by formulating the tuning of decentralized Greatest-Of (GO)CFAR and Smallest-Of (SO)-CFAR factors as a Neyman-Pearson constrained optimization problem and solving it using the Moth-Flame Optimization (MFO) algorithm. The resulting framework ensures a prescribed global false alarm rate while improving global detection probability in both identical and non-identical sensor configurations.

Extensive numerical evaluations validate the efficiency of the proposed methods using both synthetic and real radar data from the Intelligent Pixel processing X-band (IPIX) radar experiment data. Results show that the integration of Deep-Learning (DL)-based estimators into radar signal processing chains can lead to highly accurate, real-time adaptive detection systems that are robust to clutter non-stationarity and correlation.

This work paves the way for future research into fully cognitive radar systems, where clutter modeling, parameter estimation, and detection thresholds are dynamically controlled through learning-driven paradigms. The results demonstrate the practical potential of combining statistical signal processing and deep learning for next-generation intelligent radar systems.

: Radar Signal Processing, Deep Learning, Clutter Modeling, Parameter Estimation, Target Detection, CNN-LSTM, CFAR, AI-Based Detection.

Résumé

Le traitement du signal radar joue un rôle fondamental dans les systèmes modernes de surveillance, de télédétection et de défense, où la détection fiable des cibles repose sur une estimation précise des paramètres du clutter, en particulier dans des environnements maritimes ou terrestres complexes. Cette thèse s'intéresse à la modélisation du clutter radar, à l'estimation de ses paramètres et à la détection de cibles, en mettant l'accent sur l'intégration de techniques d'apprentissage profond issues de l'intelligence artificielle.

L'étude débute par une analyse approfondie des modèles statistiques du clutter de mer et des stratégies de détection à Taux de Fausse Alarme Constant (CFAR). Un cadre de sélection de modèles fondé sur des techniques de validation croisée est ensuite proposé pour des distributions de clutter composées, notamment les lois K, Pareto de type II et Gaussienne Composée-Inverse Gaussienne (CG-IG). Ce cadre permet une sélection dynamique du modèle de clutter le plus adapté à partir de données radar en temps réel, renforçant ainsi la robustesse de la détection.

Sur cette base, la thèse présente quatre contributions majeures. La première concerne la sélection statistique des modèles de clutter de mer à l'aide de critères issus de la théorie de l'information et de stratégies de validation croisée, afin d'identifier la distribution la plus appropriée selon les conditions opérationnelles et l'état de la mer.

La deuxième contribution propose une architecture hybride combinant Réseaux de Neurones Convolutifs (CNN) et réseaux à Mémoire Longue à Court Terme (LSTM) pour l'estimation du paramètre de forme du clutter distribué selon la loi K en présence de bruit additif gaussien. Cette approche fondée sur les données montre une robustesse accrue par rapport aux méthodes classiques basées sur les moments, notamment dans les scénarios à faible Rapport Signal-à-Bruit (RSB) et en conditions non stationnaires.

La troisième contribution développe un cadre d'apprentissage profond multi-têtes dédié au clutter corrélé de type Pareto II, permettant l'estimation conjointe des paramètres de forme et d'échelle. L'intégration de l'extraction de caractéristiques par convolution, de la modélisation récurrente et de structures d'auto-encodeurs permet d'améliorer significativement la précision d'estimation et l'efficacité computationnelle.

La dernière contribution porte sur la détection radar distribuée. L'ajustement des paramètres des détecteurs décentralisés Greatest-Of (GO)-CFAR et Smallest-Of (SO)-CFAR est formulé comme un problème d'optimisation contraint selon le critère de Neyman-Pearson, résolu à l'aide de l'algorithme d'Optimisation Moth-Flame (MFO). Cette approche garantit un taux global de fausse alarme imposé tout en améliorant la probabilité globale de détection, pour des configurations de capteurs identiques et non identiques.

Des validations numériques approfondies, réalisées sur des données synthétiques et des données réelles issues de l'expérience radar IPIX, confirment l'efficacité des méthodes proposées.

Les résultats montrent que l'intégration d'estimateurs basés sur l'apprentissage profond dans les chaînes de traitement radar permet de concevoir des systèmes de détection adaptatifs, précis et robustes face à la non-stationnarité et à la corrélation du clutter. Ce travail ouvre ainsi la voie au développement de systèmes radar cognitifs de nouvelle génération.

Mots-clés : Traitement du signal radar, Apprentissage profond, Modélisation du clutter, Estimation paramétrique, Détection de cibles, CNN-LSTM, CFAR, Détection basée sur l'intelligence artificielle.

Acknowledgements

All praise and thanks are due to Allah, the Most Gracious, the Most Merciful, for granting me the strength, patience, knowledge, and perseverance to complete this doctoral journey. I am deeply grateful for His countless blessings, guidance, and mercy, without which this work would not have been possible. May peace and blessings be upon the Prophet Muhammad (peace be upon him), whose teachings inspire perseverance, sincerity, and the pursuit of knowledge.

I wish to express my profound and sincere gratitude to my supervisor, Professor Amar Mezache, for his invaluable guidance, steadfast support, and insightful advice throughout the course of my doctoral studies. His exceptional expertise, scientific rigor, and constant encouragement have been fundamental in shaping the direction of this research and in fostering my growth as a researcher and scholar.

I would also like to extend my deepest appreciation to my co-supervisor, Professor Hocine Oudira, for his constructive feedback and continuous encouragement. His pertinent suggestions and thoughtful perspectives have greatly contributed to the depth and quality of this work.

My sincere and heartfelt thanks go to Professor Maria Sabrina Greco and Professor Fulvio Gini, who graciously welcomed me during my research internship in Italy. Their guidance, scientific insight, and kind support provided a stimulating scientific environment that significantly enhanced the quality and scope of my research.

I am equally indebted to the esteemed members of my doctoral committee for their valuable comments, constructive criticism, and careful evaluation, which have contributed to improving the clarity, rigor, and scientific soundness of this dissertation.

Finally, I would like to express my deepest appreciation to my family and friends for their unwavering love, patience, and faith in me. Their constant encouragement and support have been my greatest source of strength throughout this challenging yet rewarding journey.

DEDICATION

To my mother —
the light of my days, whose strength and love have guided me through every storm.
This work is, above all, a reflection of her unwavering spirit.

To the cherished memory of my father, Hafa, and my brother, Zakou —
may our paths cross again in the eternity beyond.
Your absence has shaped my resolve; your love endures in every page.

To my brother Abdelghani,
whose presence is both comfort and courage.

To my sisters Sihem, Feirouz, Chahra, and Nadia —
each of you, a verse in the poem of my life.

To my beloved wife —
the most precious gift Heaven has granted me,
my constant companion in patience, laughter, and faith.

To Yara, and to her little brother on the way —
may your curiosity and wonder illuminate the world as you have illuminated mine.

And to Ines, Aya, Louai, Assil, Elyana, Loudjain, Erwan, Arwa, Aline, and Lyn —
may this work remind you that dreams, like roots, grow deep before they reach the light. ach the light.

Contents

List of Figures

List of Tables

List of Abbreviations

AE	**A**uto**e**ncoder
AGN	**A**dditive **G**aussian **N**oise
AI	**A**rtificial **I**ntelligence
AML	**A**pproximate **M**aximum **L**ikelihood
AND	Logical "AND" Fusion Rule (All Local Detectors Must Vote Target Present)
AWGN	**A**dditive **W**hite **G**aussian **N**oise
BBO	**B**iogeography-**B**ased **O**ptimization
BLSTM	**B**idirectional **L**ong **S**hort-**T**erm **M**emory
B-LSTM-AE	**B**idirectional **LSTM** **A**uto**e**ncoder
CA-CFAR	**C**ell-**A**veraging **C**onstant **F**alse **A**larm **R**ate
CCDF	**C**omplementary **C**umulative **D**istribution **F**unction
CDF	**C**umulative **D**istribution **F**unction
CFAR	**C**onstant **F**alse **A**larm **R**ate
CG	**C**ompound **G**aussian
CG-IG	**C**ompound **G**aussian–**I**nverse **G**aussian
CNN	**C**onvolutional **N**eural **N**etwork
CNN-LSTM	**C**onvolutional **N**eural **N**etwork–**L**ong **S**hort-**T**erm **M**emory
CNR	**C**lutter-to-**N**oise **R**atio
CPI	**C**oherent **P**rocessing **I**nterval
CUT	**C**ell **U**nder **T**est
DCFAR	**D**istributed **C**onstant **F**alse **A**larm **R**ate
DL	**D**eep **L**earning
DLE	**D**eep **L**earning **E**stimator
DNN	**D**eep **N**eural **N**etwork
DoA	**D**irection of **A**rrival
DoF	**D**egrees of **F**reedom
DSP	**D**igital **S**ignal **P**rocessing
FA	**F**alse **A**larm
FAR	**F**alse **A**larm **R**ate
FFT	**F**ast **F**ourier **T**ransform
FOME	**F**ractional **O**rder **M**oment **E**stimator
FP	**F**alse **P**ositive
FN	**F**alse **N**egative
GAN	**G**enerative **A**dversarial **N**etwork
GLRT	**G**eneralized **L**ikelihood **R**atio **T**est
GLRT-LTD	**G**eneralized **L**ikelihood **R**atio **T**est–**L**inear **T**hreshold **D**etector
GO-CFAR	**G**reatest-**O**f **C**FAR
GPU	**G**raphics **P**rocessing **U**nit
GWO	**G**rey **W**olf **O**ptimization
H-pol	**H**orizontal **Pol**arization
I.I.D.	**I**ndependent and **I**dentically **D**istributed
IOM	**I**nteger **O**rder **M**oments

IPIX	**I**ntelligent **P**ixel **P**rocessing **X**-band
LSTM	**L**ong **S**hort-**T**erm **M**emory
LSTM-AE	**LSTM A**uto**e**ncoder
MAE	**M**ean **A**bsolute **E**rror
MAJORITY	Majority Fusion Rule (At Least $\lceil n/2 \rceil$ Local Detections)
MFOME	**M**odified **F**ractional **O**rder **M**oment **E**stimator
MFO	**M**oth-**F**lame **O**ptimization
MLE	**M**aximum **L**ikelihood **E**stimation
ML	**M**achine **L**earning
MLP	**M**ulti-**L**ayer **P**erceptron
MNLT	**M**emoryless **N**on-**L**inear **T**ransformation
MoM	**M**ethod of **M**oments
MSE	**M**ean **S**quared **E**rror
NIOM	**N**on-**I**nteger **O**rder **M**oments
NN	**N**eural **N**etwork
OR	Logical "OR" Fusion Rule (At Least One Detector Votes Target Present)
OS-CFAR	**O**rdered-**S**tatistics **C**FAR
PDF	**P**robability **D**ensity **F**unction
PRI	**P**ulse **R**epetition **I**nterval
PRF	**P**ulse **R**epetition **F**requency
PSO	**P**article **S**warm **O**ptimization
RCS	**R**adar **C**ross **S**ection
ReLU	**R**ectified **L**inear **U**nit
RMSE	**R**oot **M**ean **S**quared **E**rror
ROC	**R**eceiver **O**perating **C**haracteristic
RNN	**R**ecurrent **N**eural **N**etwork
SAR	**S**ynthetic **A**perture **R**adar
SGD	**S**tochastic **G**radient **D**escent
SIRP	**S**pherically **I**nvariant **R**andom **P**rocess
SIRV	**S**pherically **I**nvariant **R**andom **V**ector
SO-CFAR	**S**mallest-**O**f **C**FAR
SNR	**S**ignal-to-**N**oise **R**atio
STAP	**S**pace-**T**ime **A**daptive **P**rocessing
STFT	**S**hort-**T**ime **F**ourier **T**ransform
SVM	**S**upport **V**ector **M**achine
USV	**U**nmanned **S**urface **V**ehicle
VAE	**V**ariational **A**uto**e**ncoder

List of Symbols

H_0	Null Hypothesis (Target Absence)	–
H_1	Alternative Hypothesis (Target Presence)	–
$y(t)$	Received Radar Signal (Target + Clutter + Noise)	–
$s(t)$	Deterministic Target Return Component	–
$n(t)$	Additive Noise Component (Often AWGN)	–
P_d	Probability of Detection	–
P_{D_i}	Local Probability of Detection at Detector i	–
P_{FA}	Probability of False Alarm	–
P_{FA_i}	Local Probability of False Alarm at Detector i	–
P_f	Global False Alarm Probability after Fusion	–
P_{fa}	Probability of False Alarm (Alternative Notation)	–
f_d	Doppler Frequency	Hz
f_s	Sampling Frequency	Hz
r	Target Range / Slant Range	m
λ	Radar Wavelength	m
θ	Angle of Arrival / Look Angle	rad
ρ	Correlation Coefficient between Samples / Channels	–
SNR	Signal-to-Noise Ratio	dB or linear
CNR	Clutter-to-Noise Ratio	dB or linear
M	Number of Reference (Training) Cells Used for CFAR Thresholding	–
M_i	Number of Reference Samples Used by Detector i	–
m	Number of Coherent Pulses / Window Length (Vector Dimension)	–
N	Number of Integrated Pulses / Snapshots	–
T	Detection Threshold (Test Statistic Threshold)	–
$T(\mathbf{z})$	GLRT Test Statistic	–
α	CFAR Threshold Multiplier (Scaling Factor)	–
α_i	Local CFAR Threshold Multiplier at Sensor i	–
α_0	Prescribed Global False Alarm Probability Constraint	–
ν	Shape Parameter of the K-Distribution (Clutter Impulsiveness)	–
b	Scale Parameter of the K-Distribution	–
σ^2	Noise Variance	–
P_n	Thermal Noise Power	–
τ	Texture Random Variable (SIRV Modulation)	–
λ_P	Pareto Type II Shape Parameter (SIRV Texture)	–
η	Pareto Type II Scale Parameter (SIRV Texture)	–
β	Generic Scale Parameter (e.g., Inverse-Gamma / Pareto-Type Models)	–
$\mathbf{z}$	Observed Radar Return Vector	–
$\mathbf{s}$	Target Signal Vector	–
$\mathbf{c}$	Clutter Vector (SIRV Formulation)	–
$\mathbf{x}$	Speckle Vector (Complex Gaussian)	–
$\mathbf{M}$	Clutter Covariance Matrix	–
$\mathbf{I}_m$	Identity Matrix of Size m	–

$\mathbf{p}$	Steering Vector	–
μ	Mean (Location) Parameter / First-Order Moment	–
μ_p	Moment of Order p	–
$E[\cdot]$	Expectation Operator	–
$\mathrm{Var}[\cdot]$	Variance Operator	–
$\mathcal{L}$	Likelihood Function	–
$f(x\|\theta)$	Probability Density Function of x Given Parameter Vector θ	–
$\hat{\theta}$	Estimated Parameter Vector (e.g., $\hat{\alpha}, \hat{\nu}$)	–
$\mathcal{F}\{\cdot\}$	Fourier Transform Operator	–
$\mathbf{W}, \mathbf{b}$	Weights and Biases of a Neural Network Layer	–
h_t	Hidden State of an RNN/LSTM at Time t	–
c_t	Cell (Memory) State of an LSTM at Time t	–
f_t, i_t, o_t	Forget / Input / Output Gates of LSTM	–
$\hat{y}$	Network-Predicted Output	–
y	Ground-Truth Target Output	–
$\mathcal{L}_{\mathrm{MSE}}$	Mean Squared Error Loss	–
∇L	Gradient of Objective/Loss L	–

1 Introduction and Thesis Overview

Contents

1.1 Context and Motivation

Radar systems represent one of the most significant technological innovations of the twentieth century, playing a central role in both military and civilian applications. The fundamental principle of radar, which consists of transmitting electromagnetic waves and interpreting the returned echoes, has remained essentially unchanged since its early deployment during World War II [1,2]. However, its technological sophistication and operational capabilities have evolved considerably.

Modern radar systems operate in diverse domains including airborne and spaceborne surveillance, maritime navigation, meteorological sensing, remote exploration, autonomous transportation, and defense operations. These applications require high sensitivity, precise estimation, and reliable detection under adverse environmental conditions. In maritime scenarios, radar performance is frequently limited by sea clutter, a dominant interference source produced by the backscatter of electromagnetic waves from the ocean surface [3].

Sea clutter originates from radar backscatter off the ocean surface, which is in constant motion due to wind, waves, currents, and tidal effects. The irregular and time-varying nature of the sea surface induces complex scattering mechanisms. As a consequence, the received radar signal often departs significantly from classical Gaussian noise assumptions. Sea clutter typically exhibits heavy-tailed amplitude distributions, intermittent spikes, and spatial and temporal correlation. These characteristics make it fundamentally different from white Gaussian noise and considerably more difficult to model and suppress.

From a detection-theoretic perspective, the radar processor must decide, in the Cell Under Test (CUT), between two hypotheses: the presence H_1 or absence H_0 of a target. Performance is evaluated through the probability of detection P_d and the probability of false alarm P_{fa}. Maintaining a controlled and constant P_{fa} is essential in maritime surveillance to ensure operational credibility and avoid excessive false declarations. However, the statistical complexity of sea clutter compromises the reliability of classical detection schemes and may lead to missed detections or an unacceptable false alarm rate.

Historically, early radar systems relied on fixed thresholds and manual interpretation. As signal processing techniques matured, adaptive detection methods such as Constant False Alarm Rate (CFAR) strategies were introduced to compensate for variations in background clutter. Although such methods significantly improved robustness, their performance remains strongly dependent on the validity of the assumed clutter statistics.

1.2 Problem Statment

Despite decades of progress, reliable target detection in maritime environments remains a challenging scientific problem. The difficulty does not stem solely from noise power fluctuations, but from three intrinsic characteristics of sea clutter:

- Heavy-tailed and non-Gaussian amplitude behavior,
- Spatial and temporal correlation,
- Environmental non-stationarity.

First, heavy-tailed distributions imply that strong clutter echoes occur more frequently than predicted by Gaussian models. This increases the probability of false alarms and reduces the effectiveness of conventional thresholding techniques.

Second, correlation between neighboring samples violates the common assumption that radar returns are Independent and Identically Distributed (I.I.D.). In practice, clutter samples are often statistically dependent due to coherent processing and physical continuity of the sea surface. Ignoring this dependence degrades both parameter estimation and detection performance.

Third, clutter statistics are not stationary. They vary with sea state, grazing angle, polarization, radar frequency, and platform motion. As a result, detection methods tuned for one condition may perform poorly when environmental characteristics change.

These factors collectively create a fundamental mismatch between ideal statistical assumptions and real maritime radar data. Therefore, robust detection requires:

1. Accurate statistical modeling of heavy-tailed clutter,
2. Reliable parameter estimation techniques that remain stable under limited or correlated data,
3. Adaptive detection methods capable of maintaining false alarm regulation under practical uncertainties.

Traditional parametric estimation techniques such as the Method of Moments (MoM), Maximum Likelihood Estimation (MLE), and Least Squares (LS) provide analytical solutions under simplified assumptions. However, their performance may deteriorate in heavy-tailed or correlated environments. Similarly, classical CFAR techniques rely on local homogeneity and independence assumptions that are often violated in realistic sea clutter conditions.

In parallel, the increasing availability of radar datasets has stimulated interest in Machine Learning (ML) and Deep Learning (DL) approaches [4]. Unlike model-driven techniques, learning-based methods infer representations and decision rules directly from data. This data-driven paradigm offers adaptability and scalability, particularly when explicit statistical modeling becomes difficult.

Nevertheless, learning-based solutions must be carefully designed to preserve detection reliability and interpretability, especially in safety-critical maritime applications. Consequently, an effective solution should not replace statistical modeling entirely, but rather integrate analytical insight with data-driven adaptability.

1.3 Research Objectives and Contributions

The objective of this thesis is to improve radar detection performance in non-Gaussian, correlated sea clutter environments by combining statistical modeling techniques with learning-based estimation strategies.

The contributions of this work are structured around three main scientific axes.

a. Clutter Model Selection

A data-driven model selection method is proposed to identify the most suitable statistical distribution for observed sea clutter data. Instead of assuming a fixed distribution, the proposed approach compares candidate models using empirical criteria and selects the most appropriate one for subsequent processing.

b. Robust Parameter Estimation

Two complementary parameter estimation strategies are developed. The first relies on analytical estimators designed to enhance stability under heavy-tailed conditions. The second introduces deep learning-based estimators capable of learning the mapping between radar returns and clutter parameters, particularly in correlated environments. A multi-headed neural architecture is proposed to capture both local and temporal dependencies within radar data.

c. Adaptive Detection in Correlated Clutter

Building upon the estimation stage, detection strategies are enhanced to account for heavy-tailed and correlated clutter behavior. Optimization-based techniques are employed to tune detector parameters, and a learning-assisted approach is integrated into a Generalized Likelihood Ratio Test with Linear Threshold Detector (GLRT-LTD) strategy to improve robustness while preserving false alarm control.

Collectively, these contributions aim to bridge the gap between classical statistical detection methods and modern data-driven techniques. The resulting approach promotes intelligent and adaptive radar processing capable of operating reliably in dynamic maritime environments.

1.4 Thesis Organization

This thesis is organized into seven chapters.

Chapter 2 presents a detailed review of clutter modeling, parameter estimation, and CFAR detection techniques in maritime radar. It provides the theoretical and methodological foundations necessary to understand the developments introduced later.

Chapter 3 addresses clutter model selection using a cross-validation-based strategy.

Chapter 4 introduces a hybrid deep learning method for estimating parameters of K-distributed clutter in the presence of noise.

Chapter 5 presents a multi-headed deep learning estimator tailored for correlated Pareto Type II clutter and integrates it into an adaptive detection method.

Chapter 6 focuses on detection enhancement through optimization-based tuning of distributed CFAR detectors.

Finally, Chapter 7 summarizes the main findings and outlines future research directions, including adaptive and intelligent radar processing strategies.

1.5 Related Publications

The research contributions presented in this dissertation have led to several peer-reviewed publications in international journals, book chapters, and conference proceedings. These publications are directly related to the topics addressed throughout this thesis.

Journal Articles

- T. H. Kerbaa, M. S. Greco, and F. Gini, "Multi-headed deep learning-based estimator for correlated-SIRV Pareto type II distributed clutter," *EURASIP Journal on Advances in Signal Processing*, 2023. DOI: 10.1186/s13634-023-00982-8.
- T. H. Kerbaa and A. Mezache, "Parameter Estimation in Radar K-Clutter Plus Noise Based on Otsus Algorithm," *Ingénierie des Systèmes d'Information*, vol. 25, no. 3, 2020. DOI: 10.18280/isi.250302.

Book Chapter

- T. H. Kerbaa, "Performance of GLRT-LTD CFAR Processor in Correlated Pareto Clutter," in *Advances in Communication, Signal and Image Processing*, 2026. DOI: 10.1007/978-981-95-0401-5_6.

International Conference Papers

- T. H. Kerbaa et al., "Improved Decentralized SO-CFAR and GO-CFAR Detectors via Moth Flame Algorithm," *Proc. IEEE ICATEEE*, 2022. DOI: 10.1109/icateee57445.2022.10093725.
- T. H. Kerbaa et al., "Effect of Non-Integer Order Moments on Parameter Estimation of Pareto Distributed Clutter plus Noise," *Proc. SSD*, 2022. DOI: 10.1109/ssd54932.2022.9955964.
- T. H. Kerbaa et al., "On the Performance of GLRT-LTD CFAR Processor in Correlated Pareto Clutter Under Different Estimators," *Proc. SSD*, 2022. DOI: 10.1109/ssd54932.2022.9955679.
- T. H. Kerbaa et al., "CNN-LSTM Based Approach for Parameter Estimation of K-Clutter Plus Noise," *Proc. IEEE Radar Conference (RadarConf20)*, 2020. DOI: 10.1109/radarconf2043947.2020.9266571.
- T. H. Kerbaa et al., "Effect of Fractional Order Moments on Parameter Estimation of K-Clutter plus Noise," *Proc. ISPA*, 2019. DOI: 10.1109/ispa48434.2019.8966866.
- T. H. Kerbaa et al., "Model Selection of Sea Clutter Using Cross Validation Method," *Procedia Computer Science*, 2019. DOI: 10.1016/j.procs.2019.09.067.

2 State of the Art in Clutter Modeling, Parameter Estimation, and CFAR Detection in Maritime Radar

Contents

2.1 Introduction

Radar systems have long served as critical enablers of situational awareness, navigation, and target tracking across a wide spectrum of military and civilian domains. In maritime environments, the utility of radar becomes particularly pronounced due to the limited visibility and dynamic nature of open seas. From coastal surveillance and naval defense to search and rescue operations and autonomous vessel navigation, maritime radar platforms are relied upon to operate effectively under continuously changing and often adverse conditions. Yet, their effectiveness is persistently challenged by the presence of sea clutter, which arises from radar backscatter produced by the irregular and time-varying surface of the ocean [3, 5, 6].

Sea clutter is not merely a source of noise; it is a dynamic and often dominant interference component that competes with target reflections, particularly those from small, slow-moving, or low Radar Cross Section (RCS) objects such as Unmanned Surface Vehicles (USVs), inflatable boats, or floating debris. Unlike receiver noise, which is often modeled as white Gaussian and can be attenuated or filtered using linear methods, sea clutter exhibits highly non-Gaussian characteristics. Its amplitude distribution is typically heavy-tailed, meaning that extreme values (strong echoes) occur more frequently than in a Gaussian distribution. Moreover, sea clutter exhibits strong spatial and temporal correlations, especially in coherent processing intervals, and varies significantly with sea state, radar parameters (e.g., polarization, frequency, beamwidth), and platform dynamics [3].

The difficulty in distinguishing target echoes from clutter becomes especially pronounced at low grazing angles. At these shallow incidence angles, the radar wave interacts with a wide

swath of the sea surface, collecting multipath returns that combine into highly variable and often unpredictable clutter patterns. As noted by Watts [3], the non-stationarity of sea clutter at low grazing angles presents a fundamental challenge for radar performance evaluation and model development.

Operationally, radar systems must cope with these clutter characteristics in real time. For naval and coastal security applications, failure to detect a small or stealthy vessel due to sea clutter interference can result in missed threats or delayed response. Conversely, false alarms caused by misclassified clutter may burden operators or automated tracking systems, leading to resource inefficiencies or eroded trust in the radar system. The need for robust detection amidst clutter has thus driven extensive research into clutter modeling, adaptive detection schemes, and, more recently, ML techniques for clutter discrimination [3, 7].

From a signal processing perspective, sea clutter modeling is indispensable for developing effective CFAR detectors and for designing optimal filtering or tracking strategies [8–12]. Most radar detection architectures rely on an underlying assumption about the clutter statistics; this assumption influences false alarm regulation and determines the likelihood of target declaration. Inaccurate assumptions can lead to underperforming systems, either overly sensitive or insufficiently responsive, depending on the degree of model mismatch [13].

Several physical and environmental factors contribute to the complexity of sea clutter. The sea surface itself is influenced by wind fields, swell interactions, surface currents, and even surface films or ice formations. These environmental variables introduce significant variability in radar backscatter and necessitate models that can adapt across different sea states and observational geometries. Additionally, radar system parameters, such as Pulse Repetition Frequency (PRF), antenna height, beamwidth, and waveform type, further shape the clutter return. For example, horizontally polarized signals tend to experience greater backscatter intensity than vertically polarized ones at certain angles, leading to more severe clutter effects in Horizontal Polarisation (H-pol) radar systems [3].

Importantly, the spectral characteristics of sea clutter are also informative. Doppler spread, coherence time, and spectral skewness vary significantly depending on the interaction between radar waves and ocean wave structures. In coherent radar systems, Doppler processing enables the separation of target returns from clutter based on relative motion. However, in practice, sea clutter may exhibit Doppler characteristics that overlap with slow-moving targets, especially in scenarios involving drifting objects or small vessels in rough seas. This Doppler ambiguity makes it difficult to apply simple velocity-based discrimination techniques [3].

Moreover, sea clutter is inherently non-stationary. Over time and space, its statistical properties evolve due to environmental changes and platform dynamics. For instance, as a vessel moves through different ocean sectors, the radar return statistics may transition from one distribution to another. This undermines the use of static clutter models and highlights the importance of adaptive or context-aware modeling frameworks. Failure to account for clutter non-stationarity can significantly degrade detection performance and yield misleading inferences about system capabilities [14].

In response to these challenges, researchers have developed a wide range of clutter models, ranging from simple Gaussian approximations to complex compound and non-stationary stochastic processes. Each model attempts to capture specific aspects of clutter, such as its amplitude distribution, temporal correlation, or response to radar parameters. The choice of model is critical, as it determines the theoretical framework within which detection algorithms operate, and it has implications for both performance evaluation and real-time implementation [15].

With growing interest in intelligent and autonomous radar systems, especially in cognitive radar and distributed sensor networks, accurate and adaptable clutter models are more important than ever. These models must not only fit observed data but must also generalize

across diverse and unforeseen conditions, support real-time processing constraints, and integrate seamlessly into learning-based detection or classification pipelines [7].

To address these needs, this chapter offers an overview of sea clutter modeling approaches, parameter estimation techniques, and CFAR detection strategies, from both classical and modern perspectives.

2.2 Historical Evolution and Characterization of Sea Clutter

The development of sea clutter modeling has closely paralleled the evolution of radar systems themselves. From rudimentary pulse radar sets of World War II to todays high-resolution phased-array and polarimetric systems, the need to suppress and understand radar backscatter from the sea surface has been persistent. In early maritime radar systems, clutter was often viewed as an unavoidable nuisance—a random phenomenon that confounded the detection of low-RCS objects. Detection tasks were largely manual and relied heavily on operator experience. Operators visually interpreted plan position indicator displays and distinguished targets from clutter based on motion, consistency, and spatial isolation [16].

In the 1950s and 1960s, improvements in transmitter and receiver design enabled greater sensitivity and range resolution. As a result, clutter became more prominent in radar imagery, especially in dynamic maritime environments. Early automatic detection relied on fixed thresholds calibrated to average background noise levels. These techniques failed in the presence of strong localized clutter bursts, leading to either missed detections or excessive false alarms [2].

In the 1970s, Jakeman and Pusey [17] challenged Gaussian assumptions for sea clutter. Their analyses of empirical radar data demonstrated heavy tails and significant deviations from Rayleigh or Gaussian models. They introduced compound models, such as the K-distribution, which improved fits to measured clutter amplitude data, particularly in X-band low-grazing-angle regimes.

By the 1980s and 1990s, digitized radar data and signal processing advancements led to a surge in characterization studies. High-resolution datasets, including IPIX, allowed detailed examination of clutter under varied sea states, polarizations, and configurations. These observations confirmed that clutter statistics depend strongly on incidence angle, polarization, and environmental conditions such as wind speed and swell direction [6].

The evolution of radar systems introduced new challenges and opportunities. Synthetic Aperture Radar (SAR), polarimetric radar, and coherent Doppler radar added new dimensions to clutter returns. Polarimetric systems exploit differences between horizontal and vertical scattering to enhance detection, but introduce complex-valued signatures that require multivariate modeling [18].

A major realization is that sea clutter is often non-stationary even over short temporal or spatial windows. Works by Greco and coauthors emphasized the influence of long waves and non-stationary behavior in X-band sea clutter, and studied the consequences for processing and modeling [14,19]. Additional studies addressed texture retrieval and texture modeling/validation using measured sea clutter data [20, 21]. Transient phenomena such as sea clutter spikes have also been identified and analyzed as important drivers of heavy-tailed behavior and performance degradation [22].

In summary, sea clutter characterization has transitioned from heuristic descriptions to increasingly model-based and physics-informed frameworks, with strong emphasis on non-stationarity, texture, and higher-order statistics.

2.3 Statistical Modeling of Radar Sea Clutter

Statistical modeling plays a foundational role in analysis, simulation, and processing of radar clutter, particularly in maritime environments. Sea clutter, with heavy-tailed amplitude distributions, temporal-spatial correlations, and environmental sensitivity, defies simplistic assumptions of stationarity or Gaussianity. Accurate statistical models are essential for predicting performance, designing adaptive detection schemes, estimating false alarm rates, and training learning-based methods.

Early models often assumed Gaussian statistics for analytical convenience. In a Gaussian framework, the PDF of radar returns is fully described by mean and variance. Gaussian-based detectors benefit from closed-form expressions and optimality under additive Gaussian noise conditions. However, experimental sea clutter data—especially at low grazing angles and high sea states—show significant deviations from Gaussianity [17, 23].

Alternative models such as log-normal and Weibull distributions emerged. While these can fit moderate conditions, they often struggle to capture strong spikiness and extreme-tail behavior across diverse regimes [24].

This motivated compound-Gaussian (CG) models, where clutter is represented as a product of a fast-varying speckle component (zero-mean complex Gaussian) and a slowly varying texture component that modulates local power. This reflects physical scattering mechanisms: small-scale roughness drives speckle, while large-scale hydrodynamic processes drive texture [23, 25–27]. Detection structures specifically tailored to CG clutter have also been developed and analyzed in depth [28].

Among compound models, the K-distribution is widely used. It models the clutter envelope as the magnitude of a complex Gaussian process modulated by Gamma-distributed texture. Its behavior is governed by a shape parameter ν and a second parameter (often expressed in terms of a power/spread quantity such as σ^2), which together control tail heaviness and overall dispersion. This model has been validated using multiple maritime datasets and operating regimes.

Another important model is the Pareto Type II distribution. It arises when the texture is inverse-Gamma, and offers convenient closed-form expressions for both PDF and CDF, which is attractive for analytical derivations and implementations. Pareto models often fit severe clutter well, especially when amplitude spikes and non-stationarity are present [13, 26].

More recently, CG-Inverse Gaussian (CG-IG) models have been used to increase flexibility and capture mixtures of clutter types (e.g., coastal scenes with sea clutter plus additional disturbances) [27].

For CG-modeled clutter, a common representation is the Spherical Invariant Random Vector (SIRV), where the clutter vector is expressed as a random texture scaling applied to a complex Gaussian speckle vector, thereby preserving covariance structure while allowing random power modulation. This representation supports correlated clutter modeling and multichannel extensions, including polarimetric processing and Space-Time Adaptive Processing (STAP) [11].

While theoretical models provide analytic foundations, empirical modeling remains essential to capture behaviors not fully represented by idealized parametric assumptions [29]. Empirical characterization relies on diverse datasets recorded under multiple sea states, grazing angles, polarization modes, and frequencies [30].

One key empirical tool is the complementary cumulative distribution function (CCDF), used to evaluate tail behavior and goodness-of-fit against candidate distributions [31]. Time-frequency methods such as the Short-Time Fourier Transform (STFT) and spectrogram analysis reveal non-stationary dynamics and transient events that strongly influence Doppler spread and detection performance [3]. Texture descriptors in spatial/temporal domains (e.g., gray-level co-occurrence matrices (GLCMs)) support segmentation and adaptive processing [3].

Hybrid frameworks combine empirical fitting with model selection criteria such as the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC), enabling adaptation to changing clutter regimes [31].

2.4 Parameter Estimation Techniques

Accurate parameter estimation is critical for effective modeling and processing of sea clutter. Detection thresholds in CFAR algorithms, classification, and overall performance depend on precise and reliable estimates derived from radar returns. In heavy-tailed, high-dynamic-range, and time-varying clutter, conventional estimators often exhibit limitations.

Classical approaches such as MoM, MLE, and LS remain widely used. MoM matches empirical and theoretical moments, and can yield closed-form solutions for some distributions. However, heavy-tailed data can yield unstable higher-order moments, reducing robustness and reliability [32–37].

MLE is statistically efficient under correct model assumptions but can be computationally intensive for compound distributions and sensitive to initialization and local extrema. LS methods are computationally attractive but can be biased in skewed heavy-tailed settings; nonetheless, LS fitting is often used for CCDF-based estimation [37].

Robust alternatives include the [z log(z)] estimator, which improves numerical stability and robustness to outliers for heavy-tailed models [36]. Fractional-order moment approaches extend MoM by using fractional moments, improving stability when integer moments do not exist or are unreliable [33].

ML-based estimators have introduced additional capabilities. Neural networks (e.g., Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs)) can learn mappings from radar representations (histograms, spectra, range-Doppler maps) to distribution parameters. Long Short-Term Memory (LSTM) architectures capture temporal dependencies in clutter evolution, and hybrid CNN–LSTM designs can jointly learn spatial-temporal representations [7,18,38]. These methods provide fast inference after training but require careful validation and sufficiently diverse training data.

2.5 CFAR Detection Strategies

The CFAR framework remains fundamental in radar detection, maintaining a consistent false alarm probability under varying clutter. CFAR techniques adapt thresholds using local clutter statistics, which is crucial in maritime radar where clutter can change rapidly [39–51].

Classical CFAR methods assume a statistically homogeneous background across a training window centered on the CUT. CA-CFAR uses the average power of training cells to set the threshold. Although efficient, it performs poorly in non-homogeneous or interfering-target environments [39].

Variants improve robustness: GO-CFAR and SO-CFAR select extrema from split windows, mitigating clutter edges; OS-CFAR uses order statistics to increase robustness in multi-target and heterogeneous settings [40, 41].

When clutter deviates from Gaussian assumptions, distribution-aware CFAR approaches are needed. Pareto CFAR exploits analytic Pareto forms, and K-based CFAR incorporates heavy-tail parameters (at the cost of accurate real-time estimation) [13]. In compound-Gaussian clutter plus thermal noise, robust CFAR formulations have been developed to preserve false alarm control under realistic disturbance mixtures [52]. Detection structures explicitly designed for compound-Gaussian clutter have also been studied, emphasizing practical architectures and performance trade-offs [28].

Model-based adaptive methods include GLRT approaches (including GLRT-LTD), which compare likelihoods under H_0 and H_1 using local parameter estimates, improving robustness under heterogeneity [49, 50, 53].

Distributed CFAR and decentralized detection have also been developed for multi-sensor and networked radar settings. In particular, works by Gini, Lombardini, and Varshney address distributed detection with multiple local free parameters, providing useful theoretical foundations for decentralized CFAR design [54].

Optimization-based approaches (e.g., GAs, PSO, and MFO) tune CFAR parameters to improve detection probability under strong variability [44,55–57]. Correlation-aware approaches, including covariance-matching CFAR and STAP, exploit spatial/Doppler structure in coherent systems [7].

Bayesian CFAR has gained prominence as a principled solution to parameter uncertainty. Bayesian predictive CFAR integrates over nuisance parameters and yields thresholds that naturally adjust to limited training data and regime transitions [51].

Learning-based approaches increasingly complement model-based CFAR. Supervised models (e.g., Support Vector Machines (SVMs)) and DL architectures can emulate CFAR behavior or learn adaptive decision rules directly from data [7, 18]. Unsupervised methods (e.g., autoencoders and Variational Autoencoders (VAEs)) support anomaly-style detection, and Generative Adversarial Networks (GANs) can augment training data [7, 58].

2.6 Machine Learning and Deep Learning in Radar

In recent years, ML and DL have opened new avenues for clutter modeling, parameter estimation, and target detection. These methods are attractive in settings where classical model-based processing struggles due to high dimensionality, correlation, and non-stationarity [7].

The appeal lies in learning nonlinear mappings from radar data to quantities of interest (e.g., clutter parameters or hypotheses). Unlike purely model-driven approaches, learning-based methods infer decision rules from data, using labeled or unlabeled examples [59–63].

2.6.1 Data-Driven Clutter Modeling and Parameter Estimation

Supervised regression can estimate distribution parameters from raw returns or derived features (histograms, Doppler spectra, range-Doppler maps) [7, 18]. CNNs learn spatial features, RNNs capture temporal structure, and hybrid CNN–LSTM models jointly exploit both [7, 18, 38].

2.6.2 Neural-Based CFAR and Learned Detectors

DL has also been used to design learned CFAR-like detectors that output decisions or adaptive thresholds from local context [7, 18]. These methods can generalize across regimes but require extensive training data and careful validation to ensure constant false alarm regulation [4].

Autoencoders and VAEs support anomaly detection by learning compact representations of nominal clutter; deviations can indicate targets or abnormal events [58].

2.6.3 Toward Cognitive and Intelligent Radar Systems

The integration of ML/DL supports cognitive radar, where sensing strategies adapt based on feedback and context [7]. Federated learning offers additional potential in distributed radar networks, enabling collaborative training without sharing raw data [64].

2.7 Summary and Prospects for Research

This chapter reviewed the state-of-the-art in clutter modeling, parameter estimation, and CFAR detection for maritime radar. Sea clutters non-Gaussian, non-stationary, and correlated nature complicates reliable detection, motivating compound and texture-based models, robust estimators, correlation-aware detection, and modern Bayesian and learning-assisted approaches.

The next chapters build on these foundations, focusing on adaptive model selection, robust estimation under noise and correlation, and enhanced detection architectures that preserve false alarm control under practical uncertainty.

3 Sea Clutter Model Selection Using Cross-Validation Method

Contents

3.1 Introduction

Accurate statistical modeling of sea clutter is a prerequisite for reliable adaptive detection in maritime radar systems. As established in Chapter 2, clutter statistics directly influence detection thresholds and significantly impact CFAR performance. However, no single statistical distribution consistently provides an adequate representation across all radar resolutions and sea states. Therefore, model selection becomes a critical step in adaptive radar processing.

This chapter introduces a systematic and data-driven model selection approach for sea radar clutter characterization. Three candidate statistical distributions are considered: the K-distribution, Pareto Type II distribution, and the CG-IG model. These models are selected due to their relevance in heavy-tailed clutter modeling and their suitability for compound-Gaussian representations of maritime radar returns.

Model selection is traditionally addressed using information-theoretic criteria such as AIC [65, 66], BIC [67, 68], bootstrap methods [69], and recent approaches based on stepwise selection and Kullback-Leibler divergence [70, 71] which balance goodness-of-fit against model complexity. These criteria have been widely adopted in signal processing applications for selecting parametric models under limited data conditions. While such methods provide theoretical justification for model comparison, their effectiveness depends on the validity of underlying assumptions and may be sensitive to sample size and clutter non-stationarity.

To address these limitations, this chapter proposes a cross-validation-based selection strategy using real sea clutter data acquired from the IPIX radar system. The dataset is partitioned into four disjoint segments to assess model generalization capability and mitigate overfitting effects. For each segment, the empirical CCDF is computed and compared with the theoretical CCDF generated from each candidate distribution using estimated parameters.

The selection criterion is based on minimizing the MSE between empirical and theoretical CCDF curves across validation folds. The model achieving the lowest averaged MSE is identified as the most appropriate representation for the considered resolution scenario.

This chapter contributes a validation-driven model selection methodology that bridges statistical clutter modeling and practical adaptive radar detection requirements.

3.2 Overview of Candidate Sea Clutter Models & Parameter Estimation

Sea clutter is often characterized by heavy-tailed distributions, which require advanced statistical models to accurately describe its statistical properties. Traditional Gaussian-based models fail to capture the complex nature of sea clutter, leading to the adoption of compound Gaussian models. These models separate clutter into two components: the speckle component, typically following an exponential distribution, and the texture component, which accounts for large-scale variations and follows different distributions depending on environmental conditions.

This section outlines the most prominent models used for sea clutter representation, with attention to their mathematical formulation, empirical validation, and limitations. The candidate models discussed herein include the K-distribution, Pareto Type II distribution, and the CG-IG model.

The K-distribution is characterized by a shape parameter, ν, and a scale parameter b. That is, for a square law detector, the PDF of the intensity, X, is given by Watts [72]:

$$p_X(x) = \int_0^\infty p(y)p(x|y)dy = \frac{2c^{\nu+1}}{\Gamma(\nu)} x^{\frac{\nu-1}{2}} K_{\nu-1}(2c\sqrt{x}) \tag{3.1}$$

where the pdfs of the speckle, X, and the texture, Y, are given respectively by

$$p_{X|Y}(x|y) = \frac{\pi}{4y^2} \exp\left(-\frac{\pi x}{4y^2}\right), \quad (0 \leq x \leq \infty) \tag{3.2}$$

$$p_Y(y) = \frac{2b^{2\nu}}{\Gamma(\nu)} y^{2\nu-1} \exp(-b^2y^2), \quad (0 \leq y \leq \infty) \tag{3.3}$$

where $\Gamma(.)$ is the Gamma function, the term $4y^2/\pi$ represents the underling mean intensity of the clutter (may vary spatially and temporally), $c = b\sqrt{\pi/2}$ and $K_\nu(x)$ is a modified Bessel function of the second kind of order. For high resolution sea clutter, values of ν are generally observed in the interval $[0.1, +\infty]$, $\nu = 0.1$ represents very spiky clutter and $\nu = \infty$ represents thermal noise.

The corresponding CCDF of Equation 3.1 is given by

$$CCDF(T) = P(T) = \int_T^{+\infty} p(x)dx = \frac{2}{\Gamma(\nu)} (T\nu)^{\nu/2} K_\nu\left(2\sqrt{T\nu}\right) \tag{3.4}$$

The GP distribution is obtained when the modulation component is an inverse gamma PDF [73]:

$$p_Y(y) = \frac{\beta^\alpha y^{-\alpha-1}}{\Gamma(\alpha)} \exp\left(-\frac{\beta}{y}\right) \tag{3.5}$$

where α is the shape parameter, and β the scale parameter. The Pareto PDF is obtained as

$$p(x) = \frac{\alpha\beta^\alpha}{(x+\beta)^{\alpha+1}} \tag{3.6}$$

Integrating Equation 3.6 from T to $+\infty$, the corresponding CCDF is also given in an closed form

$$P(T) = \left(\frac{\beta}{T+\beta}\right)^{\alpha} \tag{3.7}$$

If the inverse Gaussian law is used to describe the modulation component, the CG-IG PDF is obtained. The IG law is presented in [74]:

$$p_Y(y) = \frac{\lambda}{\sqrt{2\pi y^3}} \exp\left(-\frac{\lambda(y-\mu)^2}{2\mu^2 y}\right) \tag{3.8}$$

where λ is the shape parameter, and μ is the mean. Note that, λ relies upon sea conditions and radar parameters. Spiky clutter corresponds to values of $0 < \lambda < 1$ and the exponential distribution or Gaussian clutter is attained for $\lambda \to \infty$.

The CG-IG PDF is expressed by

$$p_X(x) = \left(\frac{\sqrt{\lambda}}{(\lambda+2x)^{3/2}} + \frac{\lambda}{\mu(\lambda+2x)}\right) \exp\left[\frac{\lambda}{\mu}\left(1 - \sqrt{1+\frac{2x}{\lambda}}\right)\right] \tag{3.9}$$

The corresponding CCDF of Equation 3.9 is determined to be

$$P(T) = \left(1+\frac{2T}{\lambda}\right)^{-1/2} \exp\left[\frac{\lambda}{\mu}\left(1 - \sqrt{1+\frac{2T}{\lambda}}\right)\right] \tag{3.10}$$

For the case of K clutter, the [z log(z)] estimation of ν is given by [72]

$$\hat{\nu} = \left[\frac{\langle x\log(x)\rangle}{\langle x\rangle} - \langle\log(x)\rangle\right]^{-1} \tag{3.11}$$

For the case of GP clutter, the [z log(z)] estimation of α is given by [73]

$$\hat{\alpha} = \left[1 + \frac{\langle x\log(x)\rangle}{\langle x\rangle} - \langle\log(x)\rangle\right]^{-1} \tag{3.12}$$

For the case of CG-IG clutter,[z log(z)] method was used to estimate λ [74]

$$\langle x\log(x)\rangle - \langle\log(x)\rangle = \frac{1-2\lambda}{\langle x\rangle} F_0\left(1,1;\frac{\lambda}{\langle x\rangle}\right) \tag{3.13}$$

After obtaining the above estimates from the data, moment of order one is used to compute the scale parameter of each model [73, 74].

3.3 Methodology

3.3.1 Data Collection and Preprocessing

The study utilizes real sea clutter data obtained from the IPIX radar date base, which offers rich empirical data for sea clutter across multiple antennas polarization (HH, HV and VV), range cells, and grazing angles. The IPIX radar system is designed for experimental studies, capable of capturing dual-polarized and frequency-agile radar returns. The dataset includes clutter measurements collected under various sea conditions, making it a valuable resource for evaluating statistical clutter models.

Before analysis, the data undergoes preprocessing to remove hardware-induced biases such as DC offsets and phase imbalances, which can distort the statistical properties of the measured clutter. After preprocessing, the dataset is segmented into four equal-sized subsets, facilitating the cross-validation process and ensuring an unbiased evaluation of the competing clutter models.

3.3.2 Cross-Validation Approach

To ensure the most suitable clutter model is selected for a given dataset, a cross-validation (CV) approach is employed. Cross-validation is a widely used technique for model evaluation in statistical learning, providing an unbiased assessment of model performance. In this study, a four-fold cross-validation method is applied, following these steps:

1. **Data Partitioning**: The dataset is split into four equal-sized subsets.
2. **Model Fitting**: Each of the three clutter models (K-distribution, Pareto Type II, and CG-IG) is fitted using three subsets while leaving one subset for validation.
3. **Validation**: The CCDF is computed for both the empirical data in the validation subset and the fitted models.
4. **Error Computation**: The mean squared error (MSE) between the empirical and modeled CCDFs is calculated:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(CCDF_{emp}(\tau_i) - CCDF_{model}(\tau_i))^2 \tag{3.14}$$

5. **Repetition**: Steps 2-4 are repeated for each subset, ensuring that each subset serves as a validation set once.
6. **Model Selection**: The model with the lowest average MSE across all iterations is chosen as the best-fit model.

This cross-validation approach ensures that the selected model is not overfitting to a specific subset of the data, making the results more robust and generalizable.

By integrating both statistical modeling and cross-validation techniques, this methodology enables an objective, data-driven selection of the most suitable clutter model for varying radar conditions. The results of this analysis provide key insights into which model is best suited for different radar resolutions and clutter environments, improving the reliability of adaptive target detection systems.

The Algorithm for the Proposed Method is :

Algorithm 1 Cross-Validation-Based Model Selection for Sea Clutter

Require: Sea clutter dataset D
Ensure: Best-fit model from $\{K\text{-distribution}, \text{Pareto Type II}, \text{CG-IG}\}$

```
Partition dataset D into four subsets: D_1, D_2, D_3, D_4
for i = 1 to 4 do
    Define training set T = D \ D_i
    for each model in {K-distribution, Pareto Type II, CG-IG} do
        Fit the model using T
        Compute CCDF for empirical data in D_i
        Compute CCDF for the fitted model
        Compute the MSE between empirical CCDF and model CCDF
    end for
end for
Compute the average MSE for each model across all iterations
Select the model with the lowest average MSE as the best-fit model
return The selected model
```

3.4 Results and Discussion

3.4.1 Performance of Different Models

The model selection process was conducted using real IPIX radar data, applying the cross-validation approach to determine the best-fitting clutter model across different radar resolutions. The evaluation was based on the MSE between the empirical CCDF and the CCDF of each fitted model. Lower MSE values indicate a better fit between the theoretical clutter model and real radar observations.

The results in table 3.1 reveal that the suitability of a given model depends significantly on the radar resolution. At high resolution (3m), the Pareto Type II model achieved the lowest MSE values, demonstrating its effectiveness in capturing strong amplitude variations and heavy-tailed clutter distributions. For medium-resolution clutter (15m), the CG-IG model provided the best fit, indicating that its inverse Gaussian texture component can adapt to moderate clutter fluctuations. At low resolution (30m), the K-distribution outperformed the other models, reflecting its ability to model speckle-dominated clutter structures where texture variations become less pronounced.

TABLE 3.1: MSE of K, GP, and CG-IG Models for HH and VV Polarizations

Model	HH (3m, 18th)	HH (3m, 7th)	VV (3m, 25th)	VV (3m, 7th)	HH (15m, 12th)	VV (30m, 28th)
K	-2.343	-3.297	-3.214	**-3.707**	-2.854	-1.600
GP	**-2.672**	**-3.342**	**-3.427**	-3.667	-2.994	-1.681
CG-IG	-2.516	-3.332	-3.290	-3.688	**-3.238**	**-1.942**

3.4.2 Comparison of CCDF Fitting and Interpretation of figures

figures 3.1 to 3.6 provide a visual representation of how well each model approximates the empirical CCDF derived from real radar clutter measurements. These comparisons offer deeper insight into the strengths and weaknesses of each model under different conditions.

Figure 3.1 presents results for HH polarization at 3m resolution in the 18th range cell, the Pareto Type II model aligns most closely with the empirical CCDF, especially in the tail region. This highlights its ability to capture extreme clutter fluctuations, which are crucial in low-resolution radar environments. A similar pattern is observed in Figure 3.4 for VV polarization at the same resolution, where the Pareto Type II model again provides the best match, confirming its robustness in handling strong clutter variations.

Figure 3.2 examines HH polarization at 3m resolution in the 7th range cell, the K-distribution shows improved performance in the mid-range intensity levels, although it deviates in the extreme tail. This suggests that while the K-distribution captures moderate clutter fluctuations well, it does not fully account for the heaviest clutter variations. A comparable trend appears in Figure 3, where VV polarization at 3m resolution in the 7th range cell is analyzed. The K-distribution provides a reasonable fit in the central clutter intensity range, but the Pareto Type II model remains superior in representing the tail behavior.

When analyzing medium-resolution clutter (15m) in Figure 3.5, which corresponds to HH polarization in the 12th range cell, the CG-IG model outperforms the other models, achieving the best alignment with the empirical CCDF. This outcome is expected, as the inverse Gaussian texture component in the CG-IG model allows it to better represent moderate clutter fluctuations, making it ideal for clutter conditions where neither strong texture variations nor purely speckle-dominated behavior is dominant.

Finally, Figure 3.6 represents low-resolution clutter (30m) under VV polarization in the 28th range cell, demonstrates that the K-distribution provides the best fit across all intensity levels.

This is a direct result of the reduced impact of texture variations in high-resolution scenarios, where clutter is primarily speckle-dominated. The GP and CG-IG models tend to overestimate the tail probabilities, leading to increased MSE values and poorer performance compared to the K-distribution.

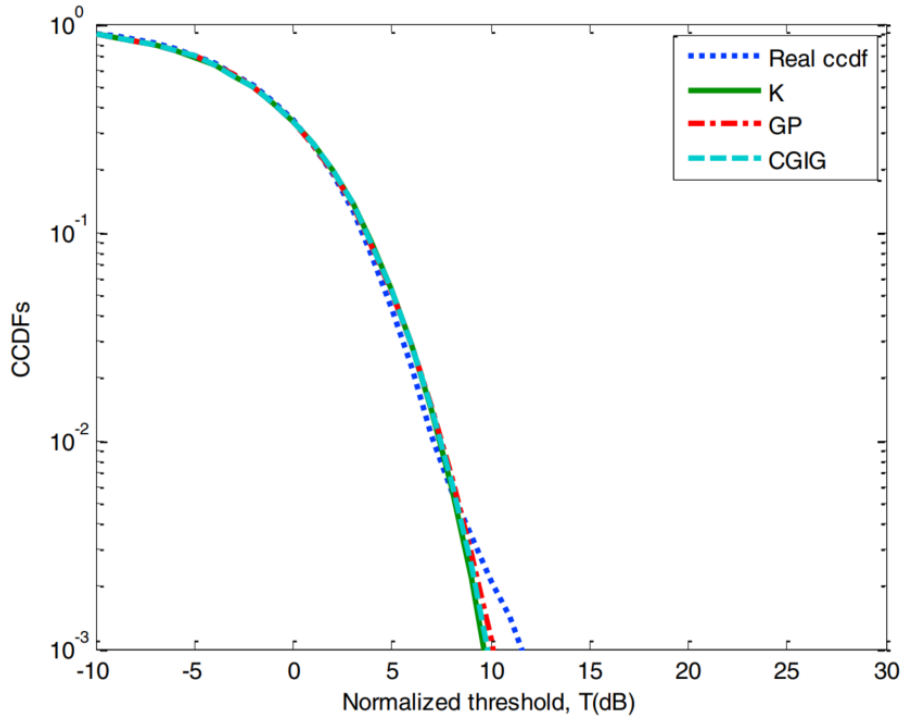

FIGURE 3.1: Fitted CCDFs for HH polarization, resolution of 3m and 18th range cell.

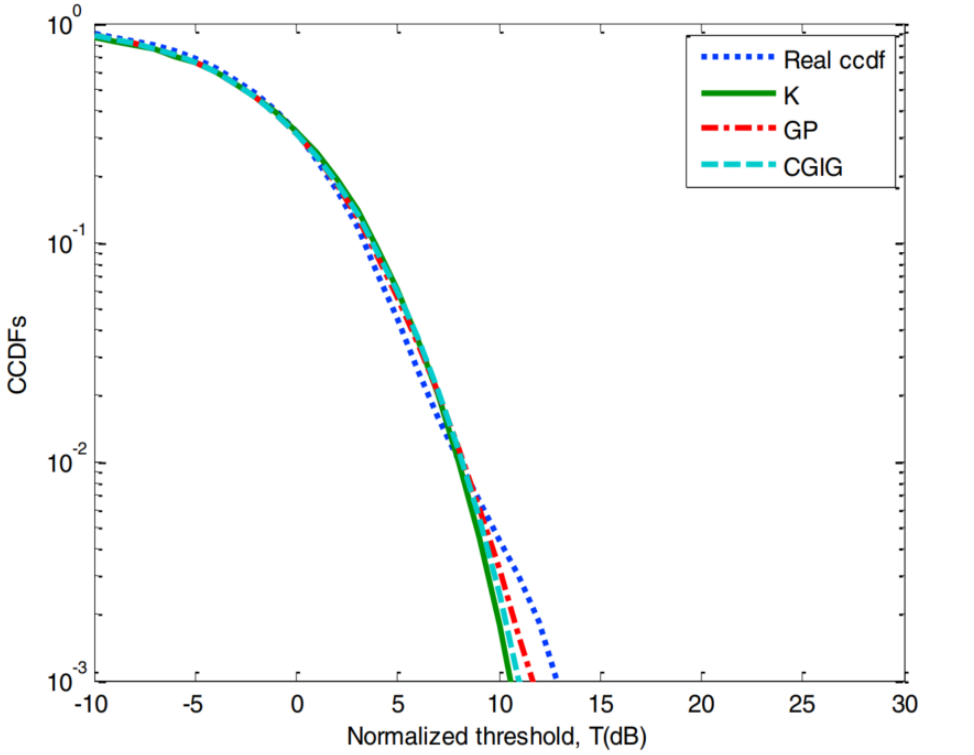

FIGURE 3.2: Fitted CCDFs for HH polarization, resolution of 3m and 7th range cell.

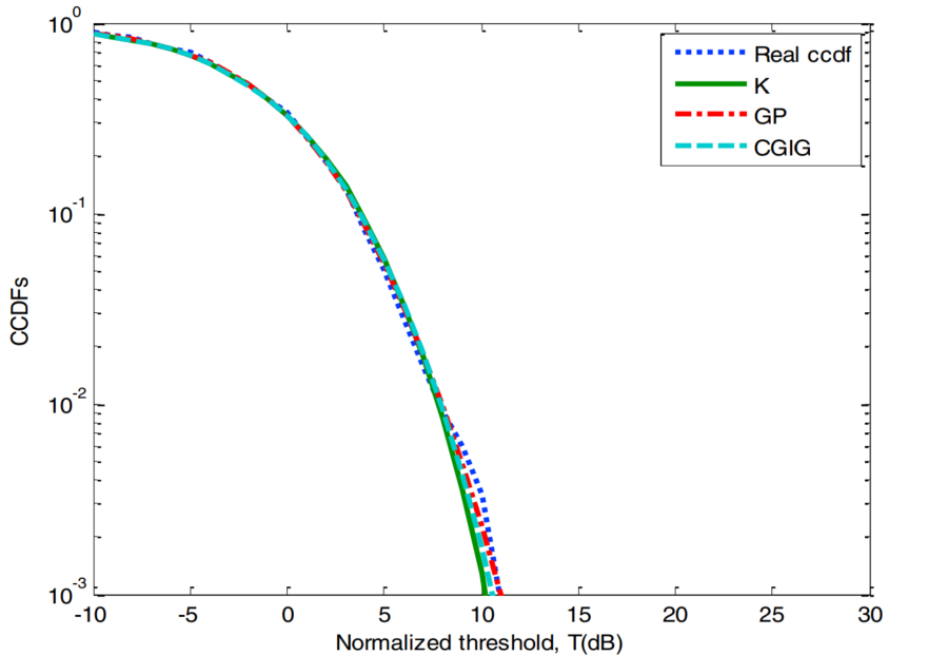

FIGURE 3.3: Fitted CCDFs for VV polarization, resolution of 3m and 7th range cell.

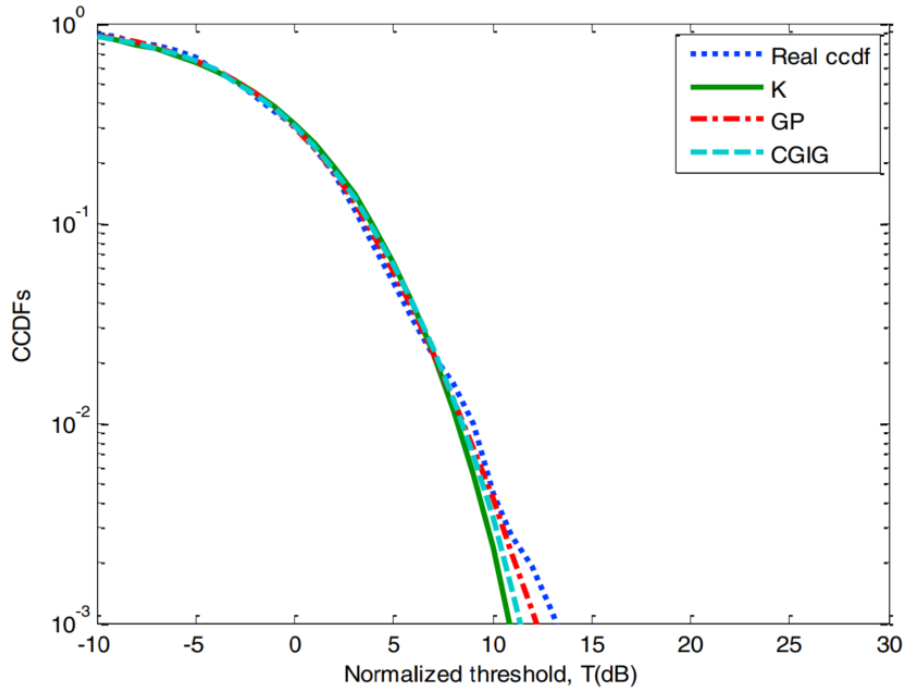

FIGURE 3.4: Fitted CCDFs for VV polarization, resolution of 3m and 25th range cell.

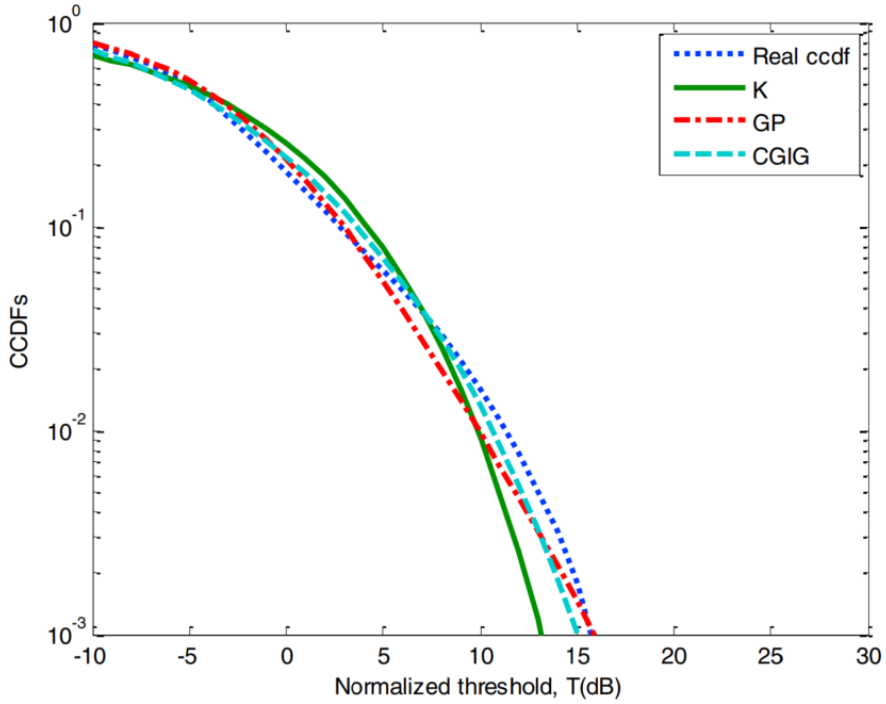

FIGURE 3.5: Fitted CCDFs for HH polarization, resolution of 15m and 12th range cell.

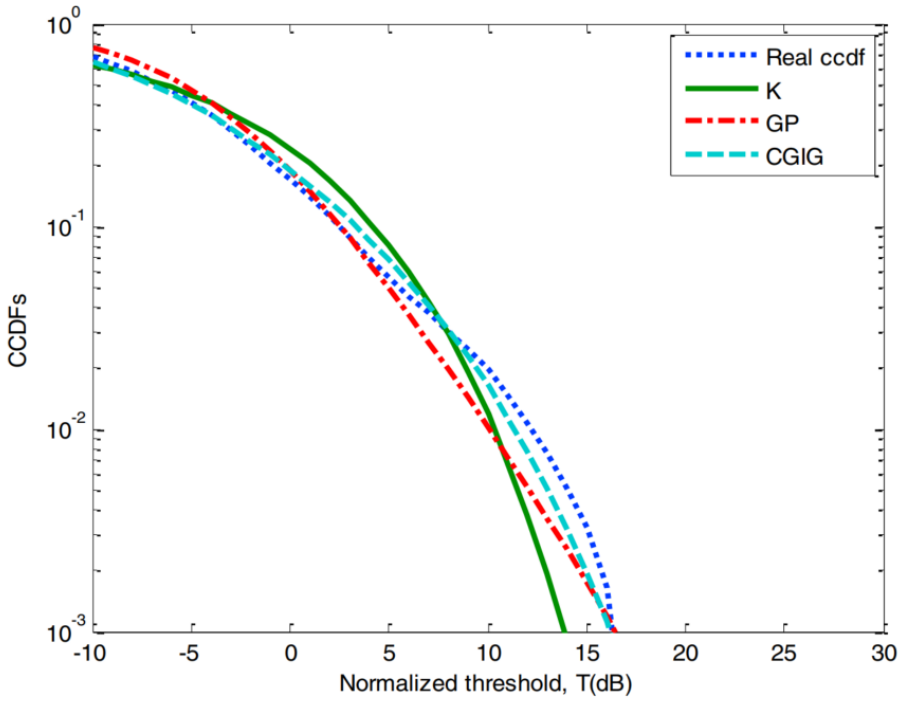

FIGURE 3.6: Fitted CCDFs for VV polarization, resolution of 30m and 28th range cell.

3.5 Conclusion and Future Research Directions

The results presented in this chapter highlight the necessity of adaptive clutter modeling in maritime radar signal processing. Since no single statistical model provides optimal performance across all resolution regimes and environmental conditions, dynamic model selection becomes essential for robust detection performance.

Experimental analysis demonstrates that clutter behavior is resolution dependent. Pareto Type II is particularly well suited for high-resolution clutter, especially in scenarios characterized by pronounced amplitude fluctuations. The CG-IG model offers improved representation in medium and low resolution cases where speckle effects become more dominant. These findings confirm that model adaptability is a critical component for maintaining reliable CFAR behavior in diverse operational environments.

The proposed cross-validation-based methodology establishes a systematic and data-driven procedure for clutter model selection. By minimizing the discrepancy between empirical and theoretical CCDF representations through an MSE criterion, the approach enables objective comparison among candidate distributions and reduces the risk of model mismatch. This contributes to improved clutter characterization, enhanced detection reliability, and better false alarm regulation.

Beyond its immediate application, this work lays the foundation for intelligent model management strategies in adaptive radar systems. Future research directions include:

- Integrating learning-based techniques to automate and accelerate model selection and parameter estimation.
- Extending the validation framework to broader datasets encompassing diverse sea states, grazing angles, and radar configurations to strengthen generalization.
- Developing real-time implementation strategies capable of dynamically switching between statistical models based on live clutter analysis.

By advancing toward adaptive and autonomous clutter modeling, radar systems can achieve improved robustness and operational reliability in complex maritime environments.

4 CNN–LSTM-Based Approach for Parameter Estimation of K-Clutter Plus Noise

Contents

4.1 Introduction

This chapter addresses the problem of estimating the shape ν and scale b parameters of K-distributed sea clutter in the presence of additive noise and under practical operational constraints such as limited training data, low signal-to-noise ratio conditions, and non-ideal statistical assumptions.

Unlike Chapter 3, which focused on adaptive model selection among multiple compound-Gaussian distributions, the present chapter concentrates on a single reference model: the K-distribution. The selection of this model is motivated by its demonstrated capability to represent highly heterogeneous and spiky sea clutter, particularly in high-resolution maritime radar scenarios. The shape parameter ν plays a central role in controlling the impulsiveness of the clutter amplitude and directly influences detection threshold calibration in CFAR-based processors. Therefore, accurate and stable estimation of ν is essential for adaptive radar performance.

Although alternative models such as Pareto Type II and CG-IG offer additional flexibility in certain clutter regimes, empirical analyses presented earlier indicate that the K-distribution remains particularly suitable in datasets characterized by low grazing angles and strong speckle-texture interactions. Consequently, this chapter isolates the K-distribution and focuses exclusively on improving the robustness and scalability of its parameter estimation.

Estimating ν is a challenging task. Classical estimators, including log-moment-based approaches and MLE, are sensitive to additive Gaussian noise and may exhibit instability when applied to short data sequences. Their performance typically deteriorates in low SNR conditions and in scenarios where clutter statistics evolve slowly over time. These limitations motivate the exploration of alternative estimation strategies that are less dependent on strict analytical assumptions.

To overcome these challenges, a data-driven estimation strategy based on a hybrid deep learning architecture is proposed. The approach combines CNN layers for local feature extraction with LSTM units for temporal dependency modeling. This architectural choice reflects the intrinsic structure of radar returns, which exhibit both short-range spatial correlations and sequential statistical patterns.

The proposed estimator is trained using synthetically generated radar sequences modeled as K-distributed clutter corrupted by Additive Gaussian Noise. To promote generalization and operational robustness, the training dataset incorporates variations in SNR levels, sequence lengths, and parameter ranges. The learning objective consists of minimizing the MSE between predicted and ground-truth values of ν.

The remainder of this chapter is organized as follows. Section 2 formalizes the signal model and defines the estimation problem. Section 3 describes the proposed neural architecture and provides the design rationale. Section 4 presents the data generation strategy and training protocol. Section 5 evaluates the estimator against classical techniques. Section 6 discusses implementation considerations and operational robustness. Finally, Section 7 concludes the chapter and outlines future research directions.

4.2 Problem Formulation

The objective is to estimate the shape parameter ν and the scale b of the K-distribution from observed radar amplitude sequences that are contaminated with additive Gaussian noise. The radar return X is assumed to follow the compound model:

$$X = S + N \tag{4.1}$$

where S is a K-distributed signal representing sea clutter, and $N \sim \mathcal{N}(0, \sigma^2)$ denotes zero-mean Gaussian noise, the K-distributed component S itself is defined as the product of a speckle component modeled as a Rayleigh-distributed variable and a texture component modeled as a gamma-distributed variable.

Mathematically, the PDF of the K-distributed signal envelope $s = |S|$ is given by:

$$p_S(s; \nu, b) = \frac{2(bs)^\nu}{\Gamma(\nu)} K_\nu(2bs) \tag{4.2}$$

where ν is the shape parameter, b is a scale parameter, and $K_\nu(\cdot)$ denotes the modified Bessel function of the second kind. The presence of the shape parameter ν makes this distribution highly flexible: small values of ν correspond to impulsive, spiky clutter, while large ν values converge toward Gaussian behavior.

However, the observed signal X does not follow this clean formulation due to the additive noise N. The noise skews the empirical distribution, making direct inversion of the PDF to retrieve ν analytically infeasible. Furthermore, radar returns are typically short in duration, especially in real-time applications, which restricts the applicability of asymptotic statistical estimators.

Given a sequence of observed amplitude values $x_1, x_2, \ldots, x_T$ drawn from the corrupted signal model, the goal is to learn a mapping:

$$f : \mathbb{R}^T \to \mathbb{R}, \tag{4.3}$$

that predicts an estimate $\hat{\nu}$ of the true shape parameter. Unlike conventional estimators, which rely on moment statistics or likelihood maximization, we propose to approximate this function $f(\cdot)$ using a data-driven approach based on a neural architecture.

The CNN–LSTM model is trained to minimize the MSE between the predicted shape parameter $\hat{\nu}$ and the true ν used in the data simulation. Formally, the training objective is:

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left(\hat{\nu}_i - \nu_i\right)^2, \tag{4.4}$$

where θ denotes the set of trainable parameters of the model, and N is the number of training samples.

In this formulation, the CNN extracts high-level local features from short amplitude windows, while the LSTM captures sequential dependencies that may reflect underlying texture and speckle interactions. The combination of these two components enables the network to generalize across various SNR conditions and input lengths. The next section provides a detailed account of this hybrid architecture and its implementation.

4.3 Classical Estimators for the K-Clutter Plus Thermal Noise Distribution

In the intensity domain, the K-clutter plus noise distribution is given by

$$p(z) = \int_0^\infty p_{Z|Y}(z|y) p_Y(y)\, dy \tag{4.5}$$

where z is the intensity, y is the modulation component with PDF $p_Y(y)$, and $z|y$ is the speckle component with a conditional pdf $p_{Z|Y}(z|y)$. We denote the thermal noise power by $P_n = 2\sigma^2$ and assume that the returns from N successive pulses are uncorrelated; $z|y$ and y follow Gamma PDF with shape parameters N and ν, respectively:

$$\begin{cases} p_{Z|Y}(z \mid y) = \dfrac{z^{N-1}}{(y+p_n)^N \Gamma(N)} \exp\left(-\dfrac{z}{y+p_n}\right) \\ p(y) = \dfrac{b^\nu y^{\nu-1}}{\Gamma(\nu)} \exp(-by) \end{cases} \tag{4.6}$$

where $\Gamma(.)$ is the Gamma function and b is the scale parameter. Substituting (4.6) into (4.5), we get the K-distributed clutter plus noise model in an integral form [74]:

$$p_Z(z) = \frac{b^\nu z^{N-1}}{\Gamma(N)\Gamma(\nu)} \int_0^\infty \frac{y^{\nu-1}}{(p_n+y)^N} \exp\left(-\frac{z}{p_n+y} - by\right) dy \tag{4.7}$$

From (4.7), the moment of order p is given by

$$\langle Z^p \rangle = \frac{b^\nu \Gamma(p+N)}{\Gamma(N)\Gamma(\nu)} \int_0^\infty (p_n+y)^p y^{\nu-1} \exp(-by) dy \tag{4.8}$$

On the other hand, the sample moment of order p is computed from M I.I.D. samples of the intensity Z,

$$\langle Z^p \rangle = \frac{1}{M} \sum_{i=1}^{M} z_i^p \tag{4.9}$$

The Higher Order Moment Estimator (HOME) presented in [36] suggests to solve (4.8) in closed form for integer values of p . By solving the integrals for $p = 1, 2, 3$, the estimate of the shape parameter ν is obtained, then $\hat{p}_n$ and $\hat{b}$ are derived from $\hat{\nu}$ and the first two moments as:

$$\begin{cases} \hat{\nu} = \dfrac{18(\hat{\mu}_2 - 2\hat{\mu}_1^2)^3}{(12\hat{\mu}_1^3 - 9\hat{\mu}_1\hat{\mu}_2 + \hat{\mu}_3)^2} \\ \hat{p}_n = \hat{\mu}_1 - \left(0.5\hat{\nu}(\hat{\mu}_2 - 2\hat{\mu}_1^2)\right)^{1/2} \\ \hat{b} = \dfrac{\hat{\nu}}{\hat{\mu}_1 - \hat{p}_n} \end{cases} \tag{4.10}$$

where

$$\begin{cases} \hat{\mu}_1 = \dfrac{\langle Z \rangle}{N} \\ \hat{\mu}_2 = \dfrac{2\langle Z^2 \rangle}{N(N+1)} \\ \hat{\mu}_3 = \dfrac{6\langle Z^3 \rangle}{N(N+1)(N+2)} \end{cases} \tag{4.11}$$

Unfortunately, the HOME method does not provide good accuracy when the sample size is low and when the clutter-to-noise ratio (CNR $= \nu/bp_n$) is low [75]. For fractional values of p, equation (4.8) can be solved in terms of the confluent hypergeometric function, ${}_2F_0(\cdot,\cdot;\cdot)$ [37]:

$$\mu_p = \frac{p_n^p \Gamma(p+N)}{\Gamma(N)} \cdot {}_2F_0(\nu, -p; -1/bp_n) \tag{4.12}$$

The effective shape parameter, $v_{\text{eff}} = \frac{(N+1)\langle Z \rangle^2}{N(\langle Z^2 \rangle - (N+1)\langle Z \rangle^2)}$ is substituted into (4.12) to reduce the estimation complexity in one dimension. Thus, (4.12) becomes:

$$\mu_p = \left(\frac{\hat{\mu}_p N \Gamma(N)}{\hat{\mu}_1^p \Gamma(N+p)}\right) \left[1 - \left(\frac{\hat{\nu}}{v_{\text{eff}}}\right)^{1/2}\right]^p \cdot {}_2F_0\left(\hat{\nu}, -p; \, ; -(\sqrt{\hat{\nu} v_{\text{eff}}} - \hat{\nu})^{-1}\right) \tag{4.13}$$

After calculating $\hat{\mu}_1$ and $\hat{\mu}_p$, Equation (4.13) is solved numerically and offers better results than the HOME method. Nevertheless, the performance of this estimator can be affected by numerical computation errors of ${}_2F_0(\cdot,\cdot;\cdot)$ with significant execution time.

For the [z log(z)] estimator, we need to determine the means of the random variables $\log(Z)$ and $Z\log(Z)$, denoted by $\langle \log Z \rangle$ and $\langle Z \log Z \rangle$, respectively. For $N \geq 1$, the combination of these statistics with $\langle Z \rangle$ is employed where the $[\langle Z \log Z \rangle]$ estimator is given in terms of ${}_2F_0(\cdot)$ [37]:

$$\frac{\langle Z \log Z \rangle}{\langle Z \rangle} - \langle \log Z \rangle = \frac{1}{N} + \frac{1 - {}_2F_0(\hat{\nu}, 1; \, ; -(\sqrt{v_{\text{eff}}} - \hat{\nu})^{-1})}{\sqrt{\hat{\nu} v_{\text{eff}}} \hat{\nu}} \tag{4.14}$$

Subject to $N > 1$, equations (4.12) and (4.13) are handled to determine a closed-form of the $[\langle Z \log Z \rangle]$ estimator in terms of the inverse harmonic mean denoted by $\langle Z^{-1} \rangle$:

$$\hat{\nu} = v_{\text{eff}} \left(\frac{1 - \frac{N-1}{N} \langle Z \rangle \langle Z^{-1} \rangle}{v_{\text{eff}} \left(\frac{\langle Z \log Z \rangle}{\langle Z \rangle} - \langle \log Z \rangle - \frac{1}{N} \right) \left(\frac{N-1}{N} \langle Z \rangle \langle Z^{-1} \rangle \right)} \right)^2 \tag{4.15}$$

where $\langle Z \rangle$, $\langle \log(Z) \rangle$ and $\langle Z \log(Z) \rangle$ are calculated from the data and straightforward computations are carried out to get $\hat{\nu}$. Note that a closed-form is not available for the single look scenario and the [z log(z)] estimator must be solved numerically,so the computational complexity increases significantly. For both the single and multi-look case, [37] suggested that closed-form estimators of the Fractional Order Moment Estimator (FOME) reffered to as Modified Fractional Order Moment Estimator (MFOME) can be obtained by manipulating the following statistical ratios:

$$\begin{cases} \alpha_{p,q} = \dfrac{\mu_{p+q}}{\mu_p \mu_q} \\ \beta_{p,q} = \dfrac{\mu_p \mu_q}{\mu_{p+q}} \end{cases} \tag{4.16}$$

where p is a positive real and q is an integer. Substituting (4.12) into (4.16) for $q = 1$, (4.16) becomes:

$$\alpha_{p,1} = \frac{N+p}{N} \cdot \frac{\sqrt{v_{\text{eff}} - \nu}}{\sqrt{v_{\text{eff}}}} \cdot \frac{U(\nu, \nu+p+2; \sqrt{v_{\text{eff}} - \nu})}{U(\nu, \nu+p+1; \sqrt{v_{\text{eff}} - \nu})} \tag{4.17}$$

$$\beta_{p,1} = \frac{N}{N+p-1} \cdot \frac{\sqrt{v_{\text{eff}}}}{\sqrt{v_{\text{eff}} - \nu}} \cdot \frac{U(\nu, \nu+p; \sqrt{v_{\text{eff}} - \nu})}{U(\nu, \nu+p+1; \sqrt{v_{\text{eff}} - \nu})} \tag{4.18}$$

where $U(\cdot;\cdot;\cdot)$ is the Tricomi or the confluent hypergeometric function. Using the recurrence relationship:

$$U(a, b, z) = (b - 1 \pm z)U(a, b+1, z) + (a - b + 1)U(a, b-1, z)$$

It can be seen by replacing $a = \nu$, $b = \nu + p + 1$, and $z = \sqrt{v_{\text{eff}} - \nu}$ that

$$\hat{\nu} = v_{\text{eff}} \left(\frac{1 - \frac{N+p-1}{N} \beta_{p,1}}{\frac{v_{\text{eff}}}{p} \left(\frac{N}{N+p} \alpha_{p,1} - 1 \right) - \frac{N+p-1}{N} \beta_{p,1}} \right)^2 \tag{4.19}$$

4.3.1 Comparative Limitations and Transition to Learning-Based Estimation

All classical estimators are predicated on strong assumptions: stationarity, independence, and clean distributions. However, real maritime radar returns often deviate from these assumptions due to environmental and operational variability.

- Amplitude fluctuations due to sea spikes or rain clutter distort statistical moments.
- Temporal correlation between returns reduces independence.
- Real-time constraints demand fast, adaptive processing.

Consequently, there is growing interest in learning-based estimators that can generalize over diverse radar conditions without relying on strict distributional assumptions. The next section introduces a deep learning architecture designed to estimate the shape parameter ν directly from observed sequences, exploiting both spatial and temporal patterns in the data.

4.4 Proposed CNN–LSTM-Based Approach

Recent advancements in artificial intelligence, particularly deep learning, have opened new frontiers for signal processing applications in radar systems. Unlike traditional statistical estimators, deep learning methods can model nonlinear relationships and learn complex patterns

directly from raw data. This makes them particularly well-suited for challenging tasks like parameter estimation under low SNR, limited sample size, and non-stationary conditions where classical techniques such as MoM and MLE often underperform.

In this section, we propose a hybrid deep learning architecture that combines CNNs with LSTM networks to estimate the shape parameter of the K-distributed sea clutter in the presence of thermal noise. The CNN–LSTM hybrid model is particularly designed to exploit both the spatial patterns and sequential dependencies inherent in radar amplitude data.

4.4.1 Motivation for Deep Learning in Radar Parameter Estimation

Traditional estimation methods rely on analytical models and known clutter statistics. However, these methods face several limitations:

- They require large amounts of data to yield statistically reliable estimates.
- They often assume independent and identically distributed samples, which is rarely the case for real radar returns.
- Their performance degrades significantly under non-stationary or low SNR conditions.

Deep learning offers a data-driven alternative. When appropriately trained, neural networks can learn implicit clutter models and adapt to unseen noise levels and waveform distortions without requiring explicit probabilistic formulations.

4.4.2 Overview of CNN and LSTM Architectures for Sequential Radar Data

In radar signal processing, especially under non-Gaussian heavy-tailed clutter, deep learning models have shown promise as flexible and efficient alternatives to classical statistical estimators. Among them, CNN and LSTM networks have emerged as two of the most potent architectures for sequence modeling, signal feature extraction, and parameter regression. This subsection provides a detailed overview of CNN and LSTM architectures, motivating their integration into a unified CNN–LSTM hybrid for the task of estimating the shape parameter of K-distributed radar clutter.

4.4.2.1 CNN Model

CNNs are a particular type of feed-forward neural networks that map features from input data in a hierarchical way and used for data processing. It consists of a chain of layers alternating convolutional and subsampling operations to the input data, followed by a classifier as illustrated in figure 4.1 . Besides the immunity to small data transformations and the ability to process large inputs, CNNs combine the feature extraction and the classification tasks in the same process breaking all records in computer vision and many other domains. CNNs were first fashioned specifically to process 2D signals, and their applications for 1D scarce data require a suitable 2D to 1D conversion and various feature transformations were introduced [38]. However, due to the high computational complexity along with a large amount of data necessary for the training, 2D CNNs may not be an adequate solution for many 1D data applications. A compact 1D CNN architecture was recently proposed [38] to deal with the drawbacks of 2D CNNs in 1D applications and achieve high-performance levels in diverse signal processing applications. CNNs work the same way whether inputs have 1, 2, or 3 dimensions. The difference is the structure of the input data and how the convolution kernel moves across the data. The input layer of the 1D CNN receives a 1D time-series sequence instead of 2 or 3D data, and 1D filters are replaced instead of 2D filters. The convolutional layer performs a sequence of convolutions, which are simply linear weighted sums of 1D sequences passed through the activation function followed

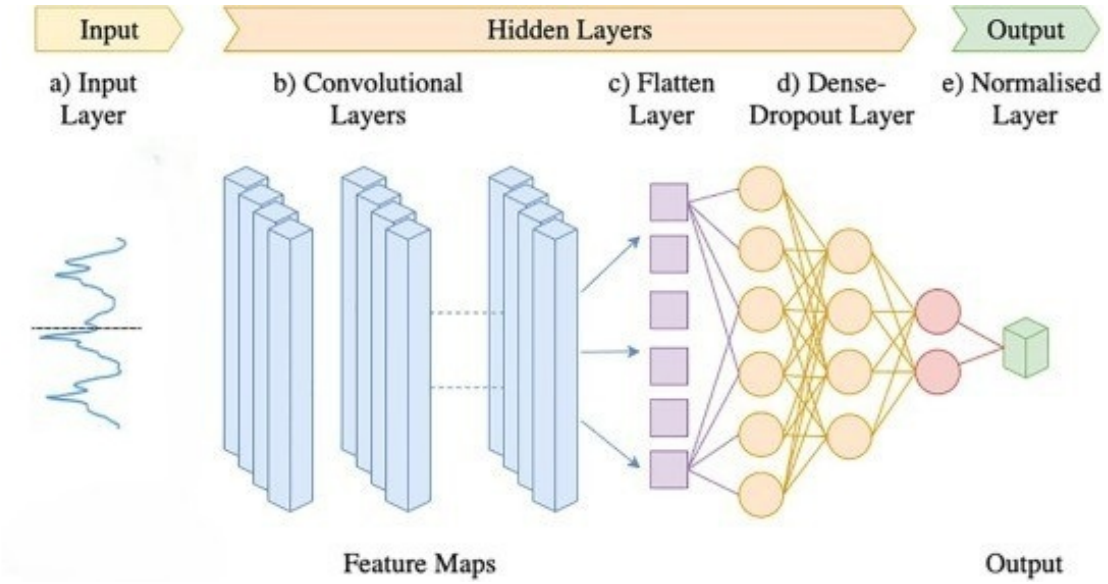

FIGURE 4.1: One-dimension CNN (1D-CNN) architecture.

by the max-pooling or other pooling operation. Consequently, the network processes 1D signals instead of 2D matrices for both kernels and feature maps, resulting in a low computational complexity. Related explanations of the Forward and Backpropagation learning algorithms in CNN layers are provided in [38].

4.4.2.2 LSTM Networks

Recurrent neural networks take their appellation from the involved recurrence connections, which provide them with a memory of past activations allowing the temporal dynamics learning and making this kind of recursive networks appropriate for modeling sequential data. RNNs are defined as a topology stimulating a discrete-time dynamical system with an input x^t, a hidden state h^t, and an output y^t. The drawback of the conventional RNNs is the exploding or vanishing gradient, which occurs when the gradient tends to be extremely small, leading to the short-term memory problem in long sequences. The LSTM architecture is proposed as an efficient attempt to solve this problem. The LSTM is constituted from three states to control the interactions between the different memory units. These states are referred to as the input, forget and output gates. Specifically, the forget gate removes redundant or irrelevant data and chooses to retain or ignore its previous status. Whereas the input state analyses the new data and checks whether or not the input signal may alter the memory cell state, the output gate, by contrast, handles the input data with the cell state [76]. In other words, the idea is to have an input vector added with the previous output vector passed through a neural network which outputs the values to keep with a 1 and the values to forget with a 0.

The block diagram of figure 4.2 illustrates the different components of a cell memory of the LSTM network.

Unlike the unidimensional, bidirectional LSTMs process the data in both forward (positive time direction) and backward (negative time direction) senses. The idea is to duplicate the first recurrent layer in the network into two side-by-side layers, then providing the input sequence as-is to the first layer, and the second layer processes a reversed copy of the input [77] It is worth noting that timesteps in the input sequence are treated one at a time the same way as the unidirectional architecture. The difference is that the network steps through the input sequence are processed in both directions at the same time.

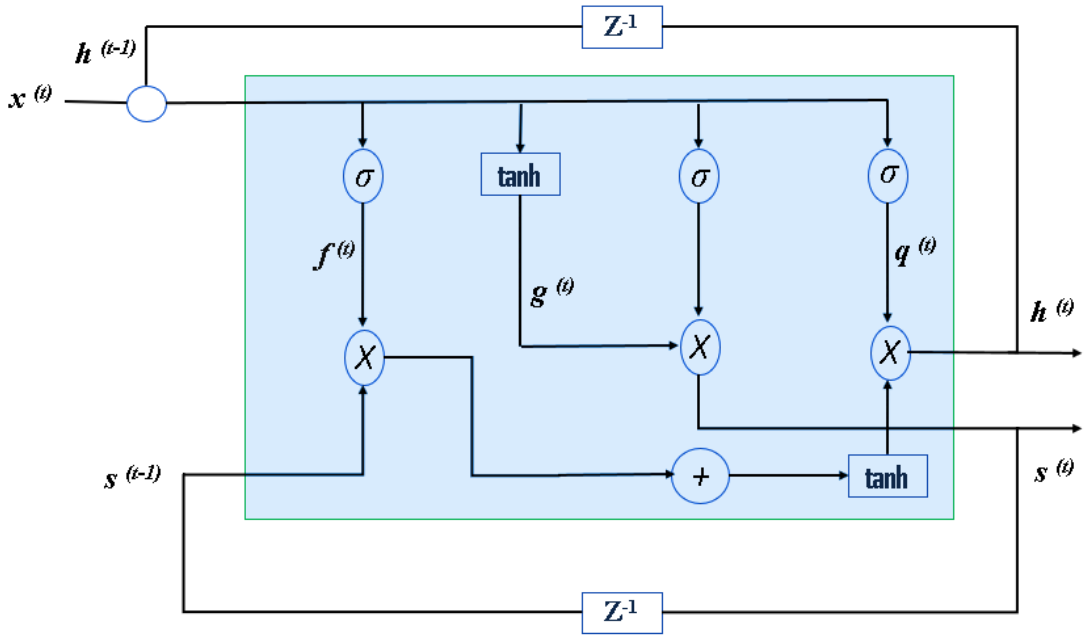

FIGURE 4.2: Block diagram of the LSTM cell architecture

4.4.3 Data Preparation and Simulation

To train the proposed CNN–LSTM model, we first generate synthetic radar return signals based on a compound K-distributed model with additive Gaussian noise. The amplitude returns are modeled according to known values of the shape parameter and clutter-to-noise ratio (CNR). A Monte Carlo simulation approach is adopted to generate diverse training scenarios.

The synthetic dataset is created as follows:

- **Clutter Simulation:** Radar echoes are generated using K-distributed amplitude returns with randomly sampled values in the range [0.1, 10].
- **Noise Injection:** Gaussian noise with varying signal-to-noise ratios (SNR) is added to mimic real operating environments.
- **Segmentation:** The radar amplitude sequences are divided into fixed-length windows suitable for feeding into the CNN.
- **Normalization:** Each sequence is normalized to have zero mean and unit variance to stabilize training.
- **Labeling:** Each segment is labeled with its corresponding ground-truth used in the simulation.

The dataset is split into training, validation, and test subsets to evaluate generalization. During training, the model is exposed to a wide variety of SNRs and shape parameters to ensure robustness and adaptability.

The architecture of the proposed model is composed of the following layers:

- **Input Layer:** Receives amplitude vectors (e.g., normalized radar return segments).
- **Convolutional Layer(s):** Extracts local spatial features from the amplitude sequences.
- **Pooling Layer(s):** Reduces dimensionality while retaining essential features.

- **LSTM Layer(s):** Encodes temporal relationships in the output of the CNN layers.
- **Fully Connected Layer:** Maps the output of the LSTM to a scalar representing the estimated parameter .

The data generation is performed as follows:

Algorithm 2 K-Clutter Plus Noise Data Generation via Monte Carlo Simulation

Require: Number of pulses N, sample size M, number of Monte Carlo trials n
Ensure: Comma-Separated Values (CSV) file containing simulated radar data with corresponding (ν, CNR) labels

Initialize empty dataset matrix $\mathcal{D} \leftarrow [\,]$
Generate vector $\text{CNR} \in [0, 10]$
Generate vector $\nu \in [0.1, 10]$
for all ν_j in ν **do**
  for all CNR_i in CNR **do**
    Compute scale parameter $b \leftarrow \nu_j \cdot (1 + 1/\text{CNR}_i)$
    Compute noise power $P_n \leftarrow 1/(1 + \text{CNR}_i)$
    for $t = 1$ to n **do**
      Generate texture samples $y \sim \text{Gamma}(\nu_j, 1/b)$
      Generate speckle samples $x \sim \text{Exp}(1)$
      Generate noise samples $\eta \sim \mathcal{N}(0, P_n)$
      Compute returns $z \leftarrow y \cdot x + \eta$
      Append (z, ν_j, CNR_i) to $\mathcal{D}$
    end for
  end for
end for
Save matrix $\mathcal{D}$ to a CSV file

4.4.4 Model Elaboration and Validation

Four different architectures were considered to estimate the shape parameter of the K-distributed clutter. The dataset comprises 20,000 vectors, each containing 1,000 simulated samples of K-distributed data corrupted with Gaussian noise. Each sample is associated with its corresponding values of ν and CNR. The dataset is randomly split into two parts: 80% for training and 20% for validation.

The adopted loss function for model training is the MSE. Three optimizers were evaluated for training performance: Adaptive Moment Estimation (Adam), Stochastic Gradient Descent (SGD), and RMSprop. Through iterative trial-and-error experimentation, the best topology and corresponding optimizer were selected. Among all tested combinations, the Adam optimizer consistently demonstrated the best performance and fastest convergence.

To promote generalization and prevent overfitting, dropout layers and batch normalization were incorporated into the architectures. The Rectified Linear Unit (ReLU) was adopted as the activation function. All models were implemented using the Python programming language with the Keras library on top of the TensorFlow backend.

Table 4.1 summarizes the average MSE loss values over 20 training epochs for the evaluated models.

The hybrid 1D CNN–LSTM architecture outperforms all standalone LSTM configurations on both training and validation datasets. The CNN–LSTM model consists of two consecutive convolutional layers, followed by a max pooling layer. The output is then flattened and fed into

TABLE 4.1: Average Loss Error of the Different Models

Models		Vanilla LSTM	Stacked LSTM	Bidirectional LSTM	1D CNN-LSTM
Loss	**Training**	0.20	0.21	0.19	0.17
	Validation	0.41	0.35	0.30	0.21

a recurrent sub-network composed of two stacked LSTM layers. A final dense (fully connected) layer outputs the estimated shape parameter.

the table 4.2 explain the role of each layer of the proposed CNN–LSTM architecture

TABLE 4.2: Proposed 1D CNN-LSTM Architecture

Layer	**Description**
Conv1D(filters=64, kernel_size =9, activation='relu')	First 1D convolutional layer: applies 64 filters over the input using a kernel of size 9, ReLU for non-linearity.
Conv1D(filters=64, kernel_size=7, activation='relu')	Second 1D convolution layer: further refines features with a smaller receptive field.
BatchNormalization	Normalizes activations for faster training and reduced sensitivity to initialization.
MaxPooling1D(pool_size=5)	Downsamples the signal by taking the maximum over windows of size 5. Reduces dimensionality.
Dropout	Randomly disables a fraction of neurons to prevent overfitting.
Flatten	Flattens the 3D output into a 2D vector suitable for LSTM input.
LSTM(512, return_sequences=True)	First recurrent layer with 512 units. Outputs a sequence to be passed to the next LSTM.
Dropout	Regularization after LSTM.
LSTM(50)	Second LSTM layer with 50 units; outputs the final temporal representation.
Dropout	Additional regularization.
BatchNormalization	Normalizes the output before dense layers.
Dense(100, activation='relu')	Fully connected layer with 100 units and ReLU activation.
Dropout	More regularization.
BatchNormalization	Normalizes again before output.
Dense(1)	Final regression output layer: estimates the continuous value (e.g., shape parameter ν).

The advantages of the Proposed Model are :

- **Robustness:** Capable of handling low SNR and short-time-window data.
- **Adaptability:** Learns from raw amplitude sequences, eliminating the need for handcrafted features.
- **Scalability:** The model can be extended to multi-parameter estimation and real-time processing.

- **Generalization:** Performs well on both synthetic and real-world datasets, including complex correlated clutter scenarios.

4.5 CNN–LSTM Estimation Performance

To evaluate the estimation performance of the proposed deep learning method, we analyzed the MSE obtained by processing both simulated and real radar data.

4.5.1 Estimation using Simulated Data

We generate a new test dataset containing 150,000 vectors, each consisting of $M = 1000$ samples. The shape parameter ν varies in the range [0.1, 1.5] (representing spiky clutter), and the CNR is considered at two values: 0 dB and 5 dB. This evaluation corresponds to the single-look case ($N = 1$). The number of Monte Carlo trials used to estimate the MSE is $n = 10{,}000$. All computations were performed on an Intel® Core™ i5-3230M CPU @ 2.60 GHz with 8.0 GB RAM.

Figures 4.3 and 4.4 display the MSE curves for CNR = 0 dB and CNR = 5 dB, respectively. In low-CNR, high-impulsiveness clutter scenarios (i.e., $\nu < 0.4$), the proposed CNN–LSTM estimator and the [z log(z)] method exhibit superior performance compared to the MFOME. However, for moderate and high values of ν, the CNN–LSTM significantly outperforms all other methods.

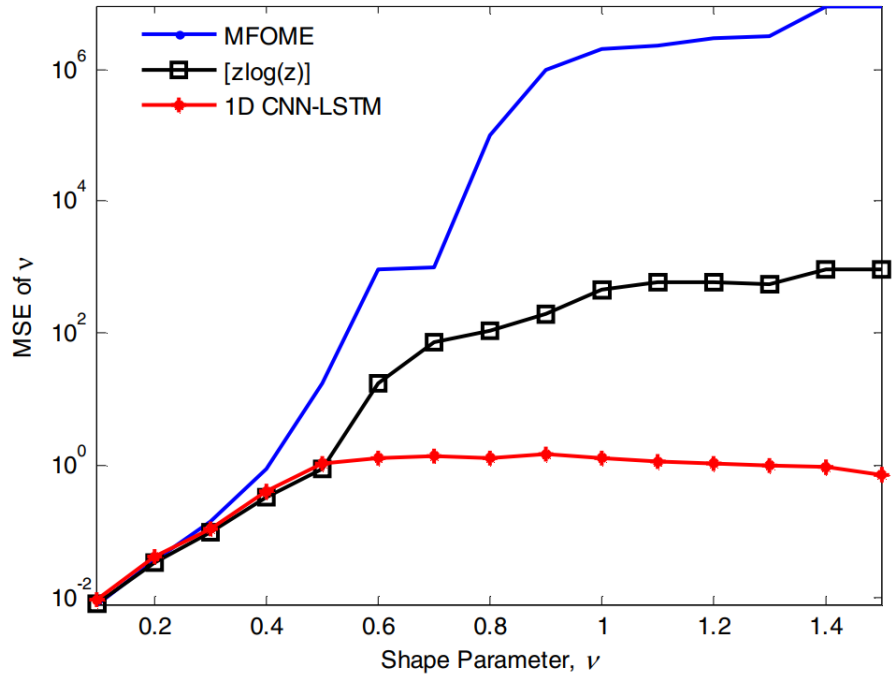

FIGURE 4.3: MSE of ν estimation for $N = 1$, $M = 1000$, and CNR = 0 dB.

These results highlight that the CNN–LSTM architecture achieves the best estimation accuracy with lower computational requirements compared to traditional estimators such as [z log(z)], which, while accurate, is computationally intensive and sometimes fails to converge.

4.5.2 Estimation using Real IPIX Data

The performance of the CNN–LSTM model was further validated using real IPIX radar data. This dataset includes fully polarimetric channels (HH, HV, VH,and VV), acquired from a lake

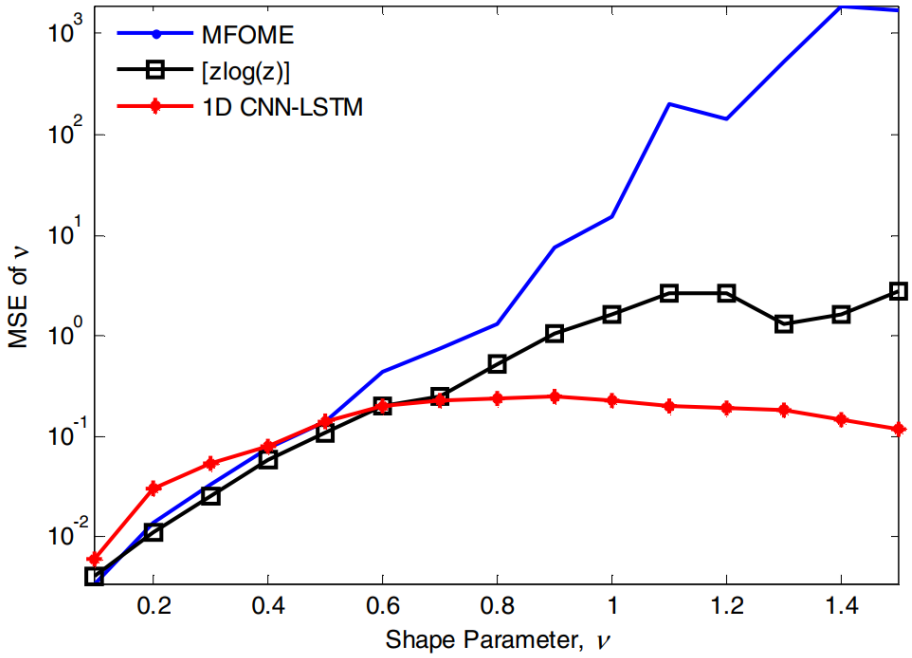

FIGURE 4.4: MSE of ν estimation for $N = 1$, $M = 1000$, and CNR = 5 dB.

surface environment at three different range resolutions. The assessment was based on the comparison between fitted and empirical CCDFs, with the evaluation focused on the tail behavior in the range $[10^{-3}, 10^{-1}]$
.

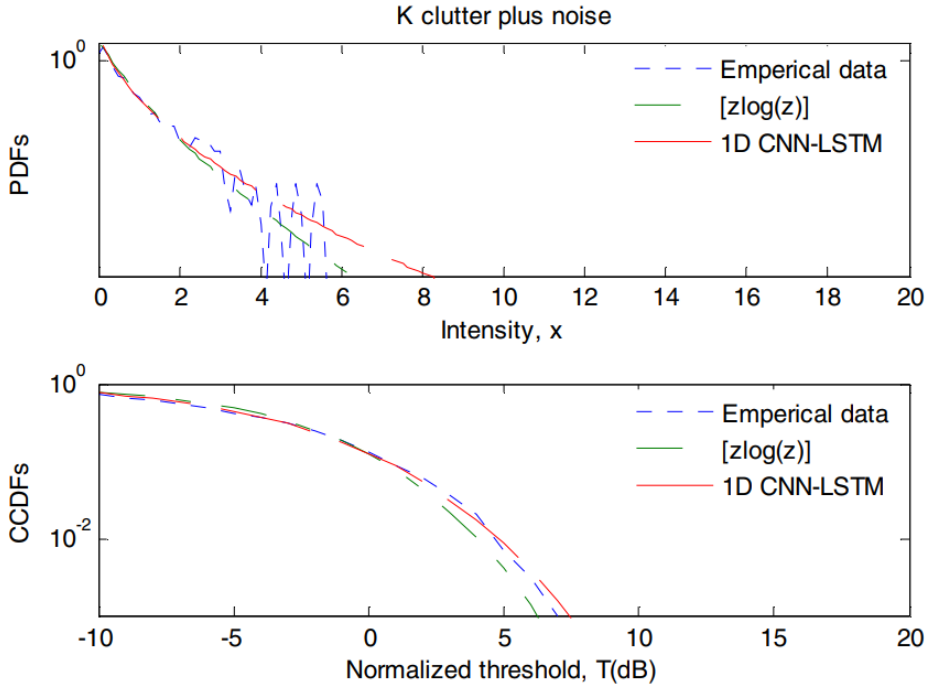

FIGURE 4.5: PDF and CCDF for HH polarization, resolution 15 m, 10th range cell.

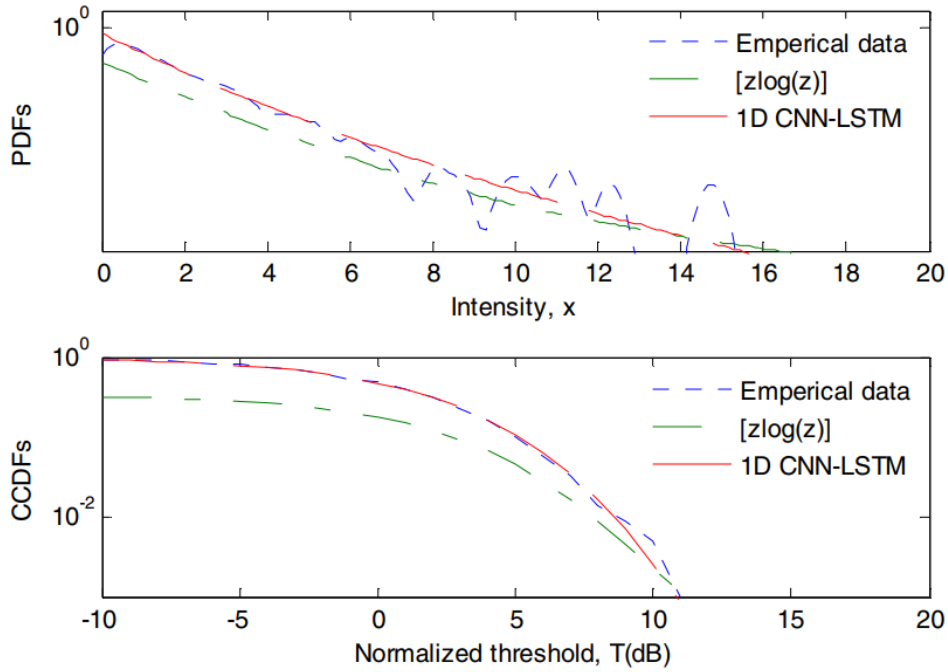

FIGURE 4.6: PDF and CCDF for VV polarization, resolution 3 m, 6th range cell.

Figure 4.5 shows the PDFs and CCDFs obtained using the CNN–LSTM estimator and the [z log(z)] method for HH polarization, 15 m resolution, and the 10th range cell. Figure 4.6 displays the results for VV polarization, 3 m resolution, and the 6th range cell. In both cases, the CNN–LSTM estimator yields a better fit to the empirical data and a lower estimation error.

4.5.3 Execution Time Comparison

The computational efficiency of the CNN–LSTM model was also benchmarked. The average runtime over 100 inference runs is 0.0157 seconds for the CNN–LSTM method, compared to 3.5841 seconds for the [z log(z)] estimator. In some instances, the [z log(z)] algorithm failed to converge, requiring manual termination.

The CNN–LSTM estimator demonstrates the following key advantages:

- Superior estimation accuracy in low-CNR and high-impulsiveness clutter regimes.
- Lower MSE across both simulated and real data benchmarks.
- Faster computation and lower latency, making it viable for real-time radar systems.
- Stable convergence behavior even in scenarios where traditional estimators fail.

This section confirms the effectiveness of deep learning architectures in radar signal processing, particularly for parameter estimation in complex, noisy environments.

4.6 Conclusion

In this chapter, we have proposed a hybrid CNN–LSTM architecture for estimating the shape parameter of K-distributed radar clutter under additive noise conditions. The solution leverages the feature extraction capabilities of convolutional layers and the sequence modeling power of recurrent LSTM layers to infer statistical parameters from raw radar data.

Our simulations and evaluations show that the proposed model achieves significantly lower estimation error compared to classical techniques, particularly in challenging environments with low CNR and small sample windows. The use of Monte Carlo simulations ensures that the network is trained across a wide range of operating scenarios, enabling robust generalization to unseen data.

Compared to traditional estimators such as MoM, MLE, and [z log(z)], the CNN-LSTM approach offers several advantages: real-time estimation capabilities, scalability to high-volume radar data streams, and minimal sensitivity to noise-induced outliers. Furthermore, its consistent performance across both synthetic and real-world datasets (e.g., IPIX data) confirms its practical utility.

The techniques discussed in this chapter pave the way for intelligent radar processing chains capable of real-time, data-driven parameter adaptation. In the next chapter, we build on this foundation to propose a more flexible and powerful multi-headed deep learning estimator, capable of adapting to multiple clutter distributions simultaneously.

5 A Multi-headed Deep Learning-Based Estimator for Correlated-SIRV Pareto Type II Distributed Clutter

Contents

5.1 Introduction

Radar signal processing in maritime environments presents complex challenges, particularly when attempting to identify targets embedded in heavy sea clutter. Traditional detection systems face significant limitations due to the dynamic and often non-Gaussian nature of sea clutter, especially under high-resolution conditions or at low grazing angles. One of the primary objectives in such applications is to achieve CFAR detection while ensuring sensitivity to weak targets. To this end, it becomes crucial to model and accurately estimate the parameters that govern the statistical distribution of clutter [16].

Earlier research shows that sea clutter can be modeled using heavy-tailed distributions such as log-normal, Weibull, K-distribution, and Pareto type II [5, 6, 10, 15]. Particular interest is adressed to the Pareto type II distribution, which, under the SIRV model, captures both the variability of the texture and the Gaussian characteristics of the speckle [12,78]. However, radar echoes in real environments are often temporally and spatially correlated, rendering insufficient the common assumption of independent and identically distributed samples. This has motivated

efforts to model clutter as a multivariate process, extending beyond the capabilities of basic i.i.d. assumptions [79].

Conventional simulation approaches for correlated clutter can be broadly categorized into Memoryless Non-Linear Transformation (MNLT) and Spherical Invariant Random Processor (SIRP]-based techniques. While MNLT techniques struggle with managing both power spectrum and PDF characteristics simultaneously, the SIRP model offers a flexible and mathematically consistent means to represent correlated clutter [8]. Specifically, SIRP enables joint control of second-order statistics and marginal distribution through the use of a texture-modulated Gaussian process [17, 80].

A number of statistical techniques have been proposed in the literature for estimating the parameters of Pareto and other compound distributions, particularly within radar applications where clutter modeling is essential. One of the earliest and most commonly used approaches is the method of Integer Order Moments (IOM), which derives closed-form expressions for the shape and scale parameters by matching sample and theoretical moments of specific orders [32, 33]. While simple and analytically tractable, IOM is sensitive to outliers and often unstable for heavy-tailed distributions, especially when the underlying shape parameter is small, as higher-order moments may not exist. To overcome these limitations, the Non-Integer Order Moments (NIOM) method was introduced, exploiting moments of fractional order that remain defined even for highly impulsive data [33]. The NIOM approach improves estimation stability in spiky clutter regimes but requires numerical routines for solving nonlinear equations, thus increasing computational complexity. Another widely technique is the [z log(z)] estimator, which uses logarithmic transformations of amplitude samples to construct nearly closed-form expressions for both parameters. As demonstrated in works such as [73], this estimator approximates the maximum likelihood solution and offers faster convergence in medium SNR conditions. However, it relies on assumptions of uncorrelated, identically distributed samples and may introduce bias in scenarios with significant temporal or spatial dependencies. MLE, on the other hand, provide statistically efficient and asymptotically unbiased estimates by optimizing the likelihood function derived from the assumed clutter distribution [34, 35]. MLE is often infeasible in real-time radar systems due to its iterative nature, sensitivity to initialization, and computational burden, especially when dealing with multidimensional, non-Gaussian, or correlated data. Moreover, in the context of Pareto type II clutter modeled via the SIRV framework, none of these estimators explicitly account for the correlation structure in the clutter returns, an omission that can lead to substantial estimation errors. For example, in the work of Sangston et al. [50], estimation within the GLRT-LTD detector framework was performed using a higher-order moments approach under the i.i.d. assumption, which, while mathematically convenient, lacks robustness in correlated clutter environments.

In parallel with the development of clutter models, a wide array of deep learning techniques have emerged, which have proven highly effective in radar applications including target detection, classification, and clutter parameter estimation [18, 59–62]. In earlier work [81], a hybrid CNN-LSTM model was successfully used to estimate the shape parameter of K-distributed clutter under AGN. By framing the estimation as a sequence regression task, the model captured temporal dependencies and yielded more robust parameter predictions than classical statistical estimators.

Building on these insights, this chapter introduces a novel multi-headed deep learning-based estimator designed to estimate both the shape (λ) and scale (η) parameters of correlated Pareto type II clutter. This estimator integrates several deep learning architectures, including CNNs, LSTM, BLSTM, CNN-LSTM, and LSTM-based Autoencoders to exploit complementary strengths of convolutional and recurrent networks. Each head contributes unique feature representations or intermediate predictions, which are fused to produce the final output. This design enables the model to handle highly correlated and spiky clutter with improved accuracy and robustness.

The significance of this contribution lies in addressing the critical gap in existing clutter parameter estimators, which are typically developed under i.i.d. assumptions. The proposed deep learning framework explicitly incorporates data correlation through training on synthetically generated SIRV samples with variable correlation coefficients. Furthermore, by leveraging the adaptive and nonlinear modeling capabilities of neural networks, the method achieves better generalization across clutter regimes and ensures reduced estimation error, even with limited training data.

To guide the reader through the content of this chapter, we provide an overview of its structure. The next section presents the mathematical formulation of the parameter estimation problem under correlated Pareto Type II clutter modeled using the SIRV framework. Following that, the third section revisits classical estimation methods, including IOM, NIOM, the [z log(z)] estimator, and MLE, highlighting their assumptions, derivations, and limitations. The design of the proposed multi-headed deep learning-based estimator is then explained, detailing the architecture of each network component, the process of synthetic data generation, and the training pipeline. Performance evaluation comparison of the proposed estimator through extensive simulations is curried out against conventional estimators under various clutter correlation scenarios. Finally, the chapter concludes with a summary of key findings and potential avenues for future work, particularly in extending the estimator to more complex or noisy clutter situations.

5.2 Problem Formulation

Radar echoes reflected from sea surfaces often exhibit heavy-tailed statistical characteristics due to surface roughness, wind-induced waves, and low grazing angles. These characteristics are well captured using the SIRV model, which treats the clutter as the product of two independent components: a Gaussian-distributed speckle and a randomly varying texture that modulates the power of the return.

Let $\mathbf{z} \in \mathbb{C}^m$ be the vector of complex radar returns from m coherent pulses. The detection problem can be posed as a binary hypothesis test:

$$\begin{cases} \mathcal{H}_0 : \mathbf{z} = \mathbf{c}, \\ \mathcal{H}_1 : \mathbf{z} = \mathbf{s} + \mathbf{c}, \end{cases} \tag{5.1}$$

where $\mathbf{s}$ is the target signal (if present), and $\mathbf{c}$ represents the clutter return modeled as:

$$\mathbf{c} = \sqrt{\tau}\,\mathbf{x}, \tag{5.2}$$

with τ the texture (a positive scalar random variable) and $\mathbf{x} \sim \mathcal{CN}(\mathbf{0}, \mathbf{M})$ the speckle component, modeled as a complex circular Gaussian vector with zero mean and covariance matrix $\mathbf{M}$ [49].

For Pareto Type II clutter, the texture follows an inverse-gamma distribution:

$$p_\tau(\tau) = \frac{1}{\Gamma(\lambda)} \left(\frac{\lambda}{\eta}\right)^{\lambda} \tau^{-\lambda-1} \exp\left(-\frac{\lambda}{\eta\tau}\right), \quad \tau > 0, \tag{5.3}$$

where $\lambda > 0$ is the shape parameter and $\eta > 0$ is the scale parameter of the distribution.

By marginalizing over the texture τ, the marginal PDF of the clutter vector becomes:

$$p_{\mathbf{z}}(\mathbf{z}) = \int_0^\infty \frac{1}{\pi^m \tau^m |\mathbf{M}|} \exp\left(-\frac{\mathbf{z}^H \mathbf{M}^{-1} \mathbf{z}}{\tau}\right) p_\tau(\tau)\, d\tau, \tag{5.4}$$

which results in a heavy-tailed distribution consistent with observed sea clutter statistics.

This formulation leads to the GLRT for Pareto Type II clutter under the GLRT-LTD framework [50]. The detection statistic is:

$$T(\mathbf{z}) = \frac{|\mathbf{p}^H \mathbf{M}^{-1} \mathbf{z}|^2}{\mathbf{p}^H \mathbf{M}^{-1} \mathbf{p}}, \tag{5.5}$$

where $\mathbf{p}$ is the steering vector of the radar array. The detection threshold is computed using the estimated clutter parameters, $\hat{\lambda}$ and $\hat{\eta}$, which significantly influence the system's probability of false alarm:

$$P_{FA} = \exp\left(-T\frac{m+\lambda-1}{m+\lambda}\right). \tag{5.6}$$

Accurate estimation of,λ and η is thus critical for effective detection. However, classical estimators such as IOM, MLE, and [z log(z)] often assume i.i.d. samples and fail to incorporate the correlation structure inherent in real radar returns.

To address this, we propose a novel deep learning-based estimator trained directly on correlated SIRV-generated data. The process involves generating massive datasets of radar returns parameterized by various values of λ, η, and correlation coefficient ρ, and using them to train a multi-headed neural architecture capable of jointly estimating both parameters.

The overall strategy of the proposed estimator is illustrated in Figure 5.1.

FIGURE 5.1: Synoptic diagram of the deep learning-based estimator for correlated Pareto type II clutter.

Figure 5.1 highlights the two main stages of the proposed framework:

1. **Data Generation Stage:** Inputs include known values of, λ, η, and correlation coefficient ρ. These are used to generate synthetic radar return vectors through the SIRV model, simulating mŒN sample matrices.

2. **Estimation Stage:** A deep learning model takes these samples and outputs estimates, $\hat{\lambda}$ and $\hat{\eta}$. During training, these predictions are compared to ground-truth parameters using a loss function (e.g., MSE), and the error is minimized via larning algorithme (e.g, backpropagation).

This framework ensures that the model learns to generalize the relationship between statistical properties of clutter and its underlying distributional parameters across a wide range of operational conditions.

The next section reviews traditional methods for estimating parameters of Pareto type II and provides a baseline for the proposed method to be evaluated.

5.3 Parameter Estimation of Multivariate Pareto Type II Clutter Model

In radar detection systems, accurate modeling and estimation of sea clutter statistics is paramount for robust CFAR operation. Particularly in maritime environments, where clutter often follows heavy-tailed distributions like the Pareto Type II, precise parameter estimation of the underlying model becomes crucial. This section details several classical statistical methods for estimating, λ and η parameters of the Pareto Type II distribution. While these methods are extensively referenced in the literature, they have traditionally been developed under the assumption of i.i.d. observations. As a result, their applicability to correlated SIRV data is limited, which we aim to address in later sections using deep learning approaches.

Let $x_1, x_2, \ldots, x_N$ denote the observed magnitudes (envelope samples) of radar clutter, assumed to follow a univariate Pareto Type II distribution. The PDF is given by [73]:

$$p_X(x) = \frac{2x\eta\Gamma(\lambda+1)}{\lambda\left(\frac{\eta}{\lambda}x^2+1\right)^{\lambda+1}\Gamma(\lambda)}, \quad x > 0, \tag{5.7}$$

The following subsections present four principal methods for estimating these parameters.

5.3.1 IOM Method

The IOM method derives estimates by equating theoretical and empirical moments of the observed data. The k-th moment of the distribution is defined as:

$$\mu_k = \left(\frac{\lambda}{\eta}\right)^{k/2} \frac{\Gamma(1+k/2)\Gamma(\lambda-k/2)}{\Gamma(\lambda)}, \tag{5.8}$$

which converges only if $\lambda > k/2$.

In practice, moments of order 2 and 4 are commonly used. The sample moments are given by:

$$\hat{\mu}_k = \frac{1}{N}\sum_{i=1}^{N} x_i^k, \quad k = 2, 4. \tag{5.9}$$

By defining the moment ratio $r_\mu = \hat{\mu}4/\hat{\mu}2^2$, λ and η can be estimated as:

$$\hat{\lambda} = \frac{2r_\mu - 1}{r_\mu - 1}, \; \hat{\eta} \; = \frac{\hat{\mu}_2}{\hat{\mu}4}(2r\mu - 1). \tag{5.10}$$

Advantages:

- The method is simple and offers closed-form expressions.
- It is computationally efficient and well-suited for rapid estimation in real-time systems.

Limitations:

- Not valid for $\lambda \leq 2$, where fourth moments do not exist and $\lambda > k/2$.
- Highly sensitive to data outliers and skewed distributions.
- Degraded performance under low SNR or correlated clutter conditions.

5.3.2 NIOM Method

To extend applicability to more impulsive clutter regimes (where $\lambda < 2$), the NIOM method employs fractional orders of the moments (e.g., 0.5 and 1). The moments of interest are given by:

$$\begin{cases} \mu_1 = \left(\frac{\lambda}{\eta}\right)^{1/2} \frac{\Gamma(3/2)\Gamma(\lambda - 1/2)}{\Gamma(\lambda)}, \\ \mu_{1/2} = \left(\frac{\lambda}{\eta}\right)^{1/4} \frac{\Gamma(5/4)\Gamma(\lambda - 1/4)}{\Gamma(\lambda)} \end{cases} \tag{5.11}$$

By numerically solving the ratio:

$$\frac{\Gamma(3/2)^2 \mu_{1/2}}{\Gamma(5/4)\mu_1^2} = \frac{\Gamma(\hat{\lambda} - 1/4)}{\Gamma(\hat{\lambda})\Gamma(\hat{\lambda} - 1/2)^2}, \tag{5.12}$$

we obtain an estimate of $\hat{\lambda}$. Once this is known, $\hat{\eta}$ is calculated as:

$$\hat{\eta} = \hat{\lambda} \left(\frac{\mu_1 \Gamma(\hat{\lambda})}{\Gamma(3/2)\Gamma(\hat{\lambda} - 1/2)} \right)^{-2} \tag{5.13}$$

Advantages:

- Enables estimation in highly spiky clutter with $\lambda \in [0.5, 2]$.
- More stable than IOM when higher moments diverge.

Limitations:

- Requires numerical routines (e.g., nonlinear solvers).
- Susceptible to estimation inaccuracies when the moments are computed over small samples.
- Ignores correlation in clutter, affecting robustness in SIRV scenarios.

5.3.3 [z log(z)] Method

This estimator, proposed in [73], leverages logarithmic expectations:

$$\hat{\lambda} = \left[\frac{E[z^2 \log(z^2)]}{E[z^2]} - E[\log(z^2)] - 1 \right]^{-1} + 1, \tag{5.14}$$

with $E[\cdot]$ denoting the empirical expectation.

Once $\hat{\lambda}$ is obtained, $\hat{\eta}$ can be estimated via:

$$\hat{\eta} = \hat{\lambda} \left(\frac{\mu_{1/2} \Gamma(\hat{\lambda})}{\Gamma(5/4)\Gamma(\hat{\lambda} - 1/4)} \right)^{-4}. \tag{5.15}$$

Advantages:

- Approximates MLE performance with less computational effort.

Limitations:

- Accuracy declines with increasing correlation or clutter heterogeneity.
- Dependent on stable computation of log-transformed statistics.
- Sensitivity to numerical instabilities when $z_i \approx 0$.

5.3.4 MLE

MLE remains the gold standard for parameter estimation, providing asymptotically efficient and unbiased estimates. The log-likelihood function of the Pareto Type II model is constructed from the joint PDF:

$$\log L(\lambda, \eta) = \sum_{i=1}^{N} \log p_X(x_i), \tag{5.16}$$

where $p_X(x)$ is as in (5.7). Partial derivatives with respect to, λ and η are taken, and the estimates are found by solving:

$$\frac{\partial \log L}{\partial \theta} = 0, \quad \text{for } \theta \in \lambda, \eta. \tag{5.17}$$

From [73], an equivalent formulation involves solving for $\hat{b} = \hat{\lambda}/\hat{\eta}$ as the root of:

$$\frac{N \sum_{i=1}^{N} \frac{z_i^2}{\hat{b} z_i^2 + 1}}{N - \hat{b} \sum_{i=1}^{N} \frac{z_i^2}{\hat{b} z_i^2 + 1}} - \sum_{i=1}^{N} \log(\hat{b} z_i^2 + 1) = 0. \tag{5.18}$$

Once $\hat{b}$ is known, we derive:

$$\begin{cases} \hat{\lambda} = \dfrac{N}{\hat{b} \sum_{i=1}^{N} \frac{z_i^2}{\hat{b} z_i^2 + 1}} - 1 \\ \hat{\eta} = \hat{b}\hat{\lambda} \end{cases} \tag{5.19}$$

Advantages:

- Theoretically optimal for large N.
- Grounded in likelihood theory with known asymptotic properties.

Limitations:

- Requires iterative solvers (e.g., Newton-Raphson).
- Sensitive to initialization and convergence criteria.
- Not practical for real-time estimation in embedded systems.
- Developed under I.I.D. assumptions; ignores correlation effects.

While the above estimators are extensively used in radar applications, their performance degrades significantly in the presence of correlated samples, particularly in the case of heavy-tailed distributions like Pareto Type II. None of these methods explicitly incorporates the covariance structure of the clutter process, leading to estimation bias and inefficiency. In the following sections, we propose a data-driven deep learning-based estimator that is trained on correlated SIRV samples and explicitly learns to model complex dependencies between samples.

5.4 Deep Learning-Based Estimation Framework

In recent years, different machine learning paradigms have received extensive attention as a multidisciplinary subject. Deep learning [4], a subfield of machine learning, belongs to the class of representation learning aiming to construct models and representations from large unprocessed data. The particularity of deep learning resides in its ability to learn hierarchical feature

representations directly from the raw data and thus reducing the hand-crafted features extraction considerably. Deep learning has presented super-human skills in many applications, including, and not limited to: object detection/recognition, anomaly detection, time series forecasting, and natural language processing. Since we are tackling the issue of estimating clutter parameters and the clutter returns are 1-dimensional signals, one can shrink the estimation process to a regression problem as a many-to-many sequence prediction. We will expose the most popular deep learning models dedicated to sequential data problems in the following subsection.

5.4.1 Overview of Relevant Deep Learning Models for Sequantial Data

In the context of radar clutter parameter estimation, the temporal and sequential nature of the data necessitates the use of deep learning models that are well-suited to handling time-dependent structures. While conventional feed-forward neural networks and multilayer perceptrons (MLPs) can approximate arbitrary functions given sufficient capacity, they fall short in capturing sequential dependencies, especially in temporally correlated sea clutter returns. For this reason, architectures such as LSTM networks, CNNs, Bidirectional LSTMs , and LSTM-based Autoencoders (LSTM-AEs) have become increasingly popular for time-series applications.

As previously outlined in Chapter 4, CNNs act as efficient feature extractors capable of identifying local patterns through shared filters and downsampling mechanisms. When combined with LSTM units specifically designed to retain information across long sequences, CNN-LSTM hybrids offer a powerful solution for both feature extraction and temporal modeling in a single pipeline. However, further improvements can be achieved by expanding the design space through bidirectional and unsupervised learning architectures such as BLSTMs and LSTM-AEs. These models provide enhanced representational capacity and greater robustness for parameter estimation in correlated non-Gaussian environments. *Convolutional Neural Networks (CNNs):* CNNs are efficient feature extractors initially designed for grid-structured inputs such as images, but widely adapted for one-dimensional signals. In the radar domain, 1D CNNs are effective in extracting spatially local patterns from clutter echoes. Their convolutional filters and shared weights help reduce computational cost and improve generalization. As detailed in Chapter 4, CNNs process radar amplitude sequences through convolution and pooling layers to highlight critical clutter features with reduced sensitivity to noise and scale variations [38].

Long Short-Term Memory Networks (LSTMs): LSTMs are a type of recurrent neural network specially designed to learn long-term dependencies in sequential data. Their internal memory structure and gating mechanism allow them to retain and forget information selectively, which is essential in clutter modeling where pulse-to-pulse variations may span different time scales. LSTMs were already discussed extensively in Chapter 4 and form the basis for several extensions in this framework [76].

5.4.1.1 Bidirectional Long Short-Term Memory (BLSTM)

The BLSTM architecture is an extension of the standard LSTM network designed to capture information in both forward and backward temporal directions. This dual encoding is particularly useful in radar signal processing, where the context of a given sample may be influenced by both past and future data points, such as in burst echoes or spread returns influenced by wave dynamics.

A standard unidirectional LSTM processes a sequence $X = (x_1, x_2, ..., x_T)$ from left to right, generating hidden states h_t by recursively combining past states and the current input. However, this one-way dependency can hinder the model's ability to fully exploit context when long-range dependencies span both sides of a sequence. BLSTM resolves this by introducing a second LSTM

layer that traverses the sequence in the reverse direction, producing a second set of hidden states $\overleftarrow{h}_t$.

The final output at each time step t is formed by concatenating the forward and backward hidden states:

$$h_t^{BLSTM} = \left[\overrightarrow{h}_t; \overleftarrow{h}_t\right], \tag{5.20}$$

where h_t^{BLSTM} denotes vector concatenation. This symmetrical design allows the model to derive richer contextual representations, benefiting estimation tasks where both preceding and succeeding samples contain relevant statistical information as illustrated in figure 5.2.

Architecture and Application:

In our proposed estimator, the BLSTM head is composed of 128 LSTM units per direction, yielding a total of 256-dimensional hidden state vectors at each time step. The bidirectional encoding is followed by a dense regression layer responsible for producing intermediate predictions of the Pareto shape and scale parameters. The model is trained using MSE loss between the predicted and true parameters. Dropout regularization and batch normalization layers are applied to improve generalization.

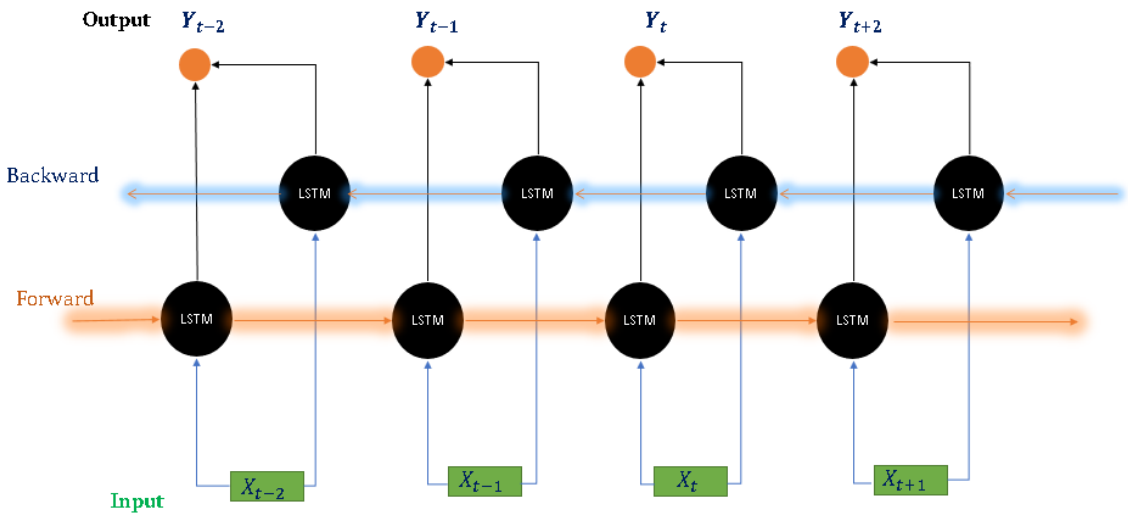

FIGURE 5.2: Bidirectional LSTM (BLSTM) architecture used as one of the heads in the multi-headed estimator.

Advantages in Radar Applications:

BLSTM networks offer significant benefits in radar clutter estimation:

- *Contextual Awareness:* Captures statistical dependencies in both directions, critical in correlated SIRV environments.
- *Stability in High Correlation:* Outperforms unidirectional models when clutter returns exhibit symmetrical dependencies or delayed correlation peaks.
- *Faster Convergence:* Due to improved signal representation, the BLSTM converges in fewer epochs with smaller parameter variances across training batches.

These advantages make BLSTM an essential component in our multi-headed architecture, ensuring that the network retains critical information lost in purely forward-processing models.

5.4.1.2 LSTM Autoencoder (LSTM-AE)

While the BLSTM focuses on directional enhancement of temporal modeling, the LSTM Autoencoder (LSTM-AE) introduces an unsupervised learning mechanism that captures high-level representations from sequential input without requiring labels. LSTM-AEs are particularly effective for feature extraction, dimensionality reduction, anomaly detection, and, in our case, learning latent structure in clutter returns.

The LSTM-AE is composed of two main components: an **encoder** and a **decoder**. The encoder compresses the input sequence into a fixed-size hidden state vector, effectively capturing the essential structure of the data. The decoder attempts to reconstruct the original input from this compressed representation.

Let $X = (x_1, x_2, ..., x_T)$ be the input radar return sequence. The encoder LSTM processes this sequence and maps it to a context vector h:

$$h = \text{LSTM}_{enc}(X). \tag{5.21}$$

The decoder LSTM then takes h as the initial state and reconstructs a sequence $\hat{X}$:

$$\hat{X} = \text{LSTM}_{dec}(h). \tag{5.22}$$

Training proceeds by minimizing the reconstruction loss:

$$\mathcal{L}_{AE} = \frac{1}{T} \sum_{t=1}^{T} \|x_t - \hat{x}_t\|^2 , \tag{5.23}$$

where $\hat{x}_t$ is the reconstructed value at time step t.

Architecture Used in the Estimator:

The LSTM-AE head in our design consists of a single encoder-decoder pair, each composed of 64 LSTM units. A RepeatVector layer bridges the encoder output to the decoder input, ensuring dimensional compatibility. The latent code h serves as a compact representation of the radar signal's spatiotemporal characteristics and is passed to the fusion layer along with outputs from the other heads, as shown in figure 5.3.

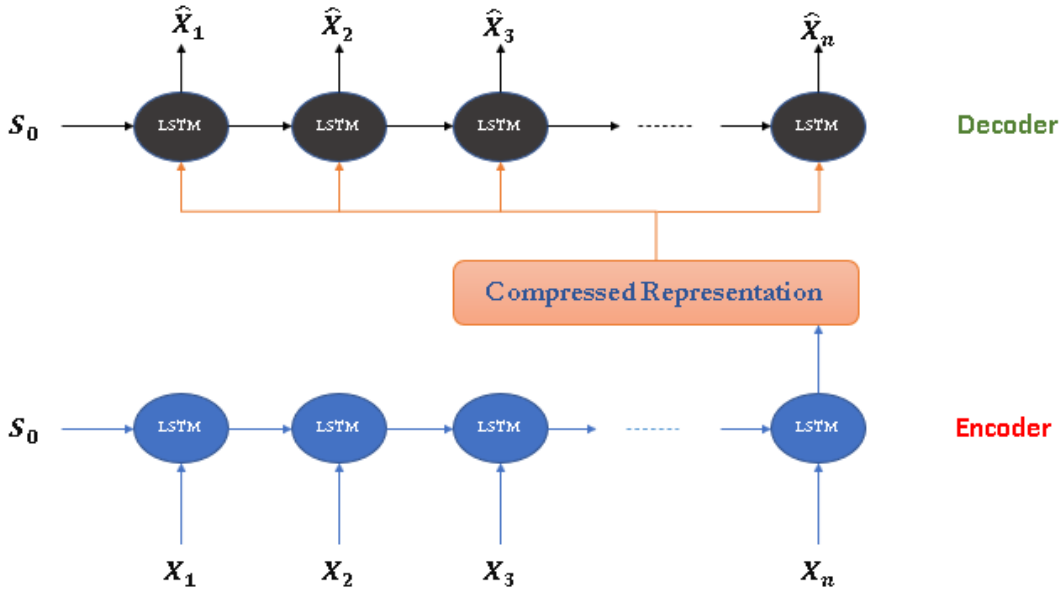

FIGURE 5.3: LSTM Autoencoder architecture.

Interpretation and Role:
The LSTM-AE serves multiple roles in our estimator:

- *Unsupervised Pre-training:* Allows the network to extract stable features without needing ground truth parameters.
- *Compression and Generalization:* Reduces dimensionality while retaining salient clutter dynamics.
- *Fusion Input:* Provides an abstracted vector summarizing the entire input sequence, which is essential for joint learning across different estimator heads.

Advantages in Clutter Modeling:
LSTM-AEs have shown superior capabilities in capturing noise-resilient patterns, particularly beneficial when the clutter returns contain overlapping distributions, e.g., due to mixture of land echoes and sea clutter. Their ability to learn compact but expressive representations makes them ideal for downstream regression tasks, as confirmed in our experimental evaluation (see Section 5.5).

Relation to Stacked Architectures:
To further enhance representational capacity, LSTM-AEs can be stacked. In a stacked LSTM-AE, multiple encoder-decoder layers are trained sequentially, with the latent code from one serving as input to the next. In our case, a two-layer stacked LSTM-AE was tested, and performance gains were observed in reducing the variance of the final parameter estimates.

In conclusion, both Bidirectional LSTM and LSTM Autoencoder architectures serve complementary functions within the proposed multi-headed estimation framework. BLSTMs enrich temporal modeling with bidirectional context, while LSTM-AEs provide unsupervised compressed features that increase the robustness and generalization of the overall system. These capabilities, combined with CNN and CNN-LSTM heads, result in a high-performance estimator capable of tackling the intricate problem of parameter estimation in correlated Pareto Type II radar clutter.

5.4.2 Data Generation and Preprocessing

The performance of any deep learning-based estimator hinges critically on the quantity, quality, and diversity of training data. For parameter estimation in SIRV-modeled radar clutter, no open-access datasets are available that offer both the empirical values of the clutter parameters (λ, η) and the correlation properties of the environment. Therefore, a synthetic data generation strategy based on theoretical models becomes necessary. The process must simulate clutter sequences that realistically capture both the second-order (correlation) and first-order (marginal PDF) statistics of Pareto Type II distributed sea clutter.

5.4.2.1 SIRV-Based Clutter Modeling

As detailed in [10],the SIRV is a compound model widely used to represent radar clutter exhibiting heavy-tailed statistics. The SIRV model expression is mentionned in equation 5.2. For correlated clutter, $\mathbf{M}$ is typically Toeplitz and Hermitian, with entries defined by a correlation coefficient ρ such that:

$$[\mathbf{M}]_{i,j} = \rho^{|i-j|}, \quad \text{for } i, j = 1, \ldots, m. \tag{5.24}$$

The texture τ follows an inverse-gamma distribution corresponding to a Pareto Type II marginal for the intensity (or amplitude) envelope:

$$p_\tau(\tau) = \frac{1}{\Gamma(\lambda)} \cdot \frac{b^{-\lambda}}{\tau^{\lambda+1}} \exp\left(-\frac{1}{b\tau}\right), \tag{5.25}$$

When marginalizing over τ, the resulting distribution of $\mathbf{z}$ is heavy-tailed and captures the impulsive behavior observed in radar returns from sea clutter. This statistical representation is a core assumption in the performance analysis of the GLRT-LTD detector, which relies on the precise knowledge (or estimation) of λ and η for accurate GLRT-CFAR detection performance [50].

5.4.2.2 Synthetic Dataset Generation

To train our estimator, we synthetically generate a dataset of radar returns under varying conditions of spikiness (controlled by λ), local power scaling (via η), and correlation (via ρ). The complete process is summarized below.

Parameter Space and Simulation Conditions

- Number of pulses per sample: $N = 4, 8, and 16$
- Sequence length (window size): $m = 24$
- Monte Carlo trials: $n = 1000$
- Shape parameter λ: uniformly sampled in $[1.1, 8]$
- Scale parameter η: uniformly sampled in $[1, 12]$
- Correlation coefficient ρ: sampled in $\{0.01, 0.1, 0.3, 0.6, 0.9, 0.99\}$

Generation Process The generation of one SIRV radar return sample proceeds as follows:

Algorithm 3 SIRV-Based Data Simulation for Training

Require: Number of pulses N, window size m, trials n, parameter ranges for λ, η, and ρ
Ensure: Dataset $\mathcal{D}$ of radar returns with labels

Initialize $\mathcal{D} \leftarrow \emptyset$
for $i = 1$ to n **do**
  Sample $\lambda_i \sim \mathcal{U}(1.1, 8)$
  Sample $\eta_i \sim \mathcal{U}(1, 12)$
  Sample $\rho_i \in \{0.01, 0.1, 0.3, 0.6, 0.9, 0.99\}$
  Compute $\mathbf{M}_i = \text{Toeplitz}(\rho_i^{|i-j|})$
  $\mathbf{L}_i \leftarrow \text{Cholesky}(\mathbf{M}_i)$
  for $j = 1$ to N **do**
    $\tau_j \sim \text{InverseGamma}(\lambda_i, \eta_i)$
    $\mathbf{x}_j \sim \mathcal{CN}(0, \mathbf{I}_m)$
    $\mathbf{c}_j = \sqrt{\tau_j} \cdot \mathbf{L}_i \cdot \mathbf{x}_j$
  end for
  Form matrix $\mathbf{C}_i = [\mathbf{c}_1, \ldots, \mathbf{c}_N]$
  Flatten $\mathbf{C}_i$ into vector z_i (real + imag parts)
  Append (z_i, λ_i, η_i) to $\mathcal{D}$
end for
return $\mathcal{D}$

5.4.2.3 Dataset Partitioning and Normalization

The complete dataset $\mathcal{D}$, typically consisting of 250,000 samples, is split into:

- 80% training set
- 20% validation set

This synthetic dataset enables supervised learning of the mapping:

$$\mathcal{F} : z_i \mapsto (\hat{\lambda}_i, \hat{\eta}_i)$$

via DL networks described in the next section.

5.4.3 Architecture Details

The proposed estimator is structured as a *multi-headed deep learning model* designed to combine the strengths of diverse deep architectures. Each head in the model either serves to extract features from the input radar return sequences or to perform intermediate predictions of the desired parameters. The outputs of these heads are fused and processed by a final regressor block that produces the final estimates of the Pareto Type II clutter parameters.

5.4.3.1 Overall Design

The architecture is composed of the following branches:

- **Head 1: LSTM Autoencoder (LSTM-AE)** Provides a compressed temporal feature representation of the radar sequence.
- **Head 2: 1D Convolutional Neural Network (1D-CNN)** Extracts local spatial patterns and amplitudinal variations.

- **Head 3: Bidirectional LSTM (BLSTM)** Captures both forward and backward temporal dependencies.
- **Head 4: 1D CNN followed by LSTM (CNN-LSTM)** Performs hierarchical feature extraction followed by temporal modeling.

The output from each head is either a feature vector or an intermediate regression output. These outputs are concatenated with the normalized original signal and passed to a fusion layer consisting of:

- A **Dense layer** for dimensionality alignment,
- A **final LSTM layer** for time-aware fusion,
- Two **fully connected layers** for regression to (λ, η).

5.4.3.2 Head 1: LSTM Autoencoder (LSTM-AE)

The LSTM Autoencoder acts as a temporal feature extractor. It is composed of:

- **Encoder**: A stack of two LSTM layers with 64 and 32 units, followed by a bottleneck (compressed latent representation).
- **Decoder**: Mirrors the encoder with two LSTM layers, reconstructing the original sequence from the latent vector.

During training, the autoencoder is trained in an unsupervised way to minimize the reconstruction error. After pretraining, only the encoder part is retained and connected to the fusion layer.

5.4.3.3 Head 2: 1D-CNN

This branch is composed of:

- **Conv1D Layer 1**: 64 filters, kernel size = 9, activation = ReLU.
- **Conv1D Layer 2**: 64 filters, kernel size = 7, activation = ReLU.
- **MaxPooling1D**: Pool size = 5.
- **Flatten + Dropout**.

This head captures localized structure in the radar sequence, suitable for identifying short-duration spikes or high amplitude clutter returns.

5.4.3.4 Head 3: Bidirectional LSTM (BLSTM)

To leverage full temporal context, this head includes:

- **BLSTM layer**: 128 units, which processes the sequence in both forward and backward directions.
- **Dense layer**: 64 units, activation = ReLU, followed by Dropout.

This architecture is ideal for modeling temporal dependencies in both directions, especially important for detecting transient patterns in correlated radar returns.

5.4.3.5 Head 4: CNN-LSTM Hybrid

Combining the advantages of CNNs (local pattern recognition) and LSTMs (temporal modeling), this head includes:

- **Conv1D Layer 1**: 32 filters, kernel size = 7, activation = ReLU.
- **Conv1D Layer 2**: 32 filters, kernel size = 5, activation = ReLU.
- **MaxPooling1D**: Pool size = 3.
- **LSTM Layer**: 32 units.
- **Dense Layer**: 2 units (intermediate predictions of λ, η).

This head is a strong predictive branch that provides a fast approximation of the final targets, which improves convergence of the whole network during training.

5.4.3.6 Fusion and Output Block

The fusion mechanism concatenates:

1. The latent feature vector from LSTM-AE encoder.
2. The flattened CNN features.
3. The BLSTM hidden state.
4. The CNN-LSTM prediction vector.
5. The normalized radar input vector.

This combined vector is passed to:

- **LSTM Fusion Layer**: 64 units.
- **Dense Layer 1**: 128 units, activation = ReLU.
- **Dense Output Layer**: 2 units (final regression of $\hat{\lambda}$ and $\hat{\eta}$).

5.4.3.7 Regularization and Optimization

To ensure generalization and prevent overfitting:

- Dropout layers (rate = 0.3) are inserted after major hidden layers.
- Batch normalization layers are used after each Dense block.
- The optimizer used is **Adam** with a learning rate of 1×10^{-4}.
- Loss function: **Mean Squared Error (MSE)**.

5.4.3.8 Summary of the Proposed Estimator

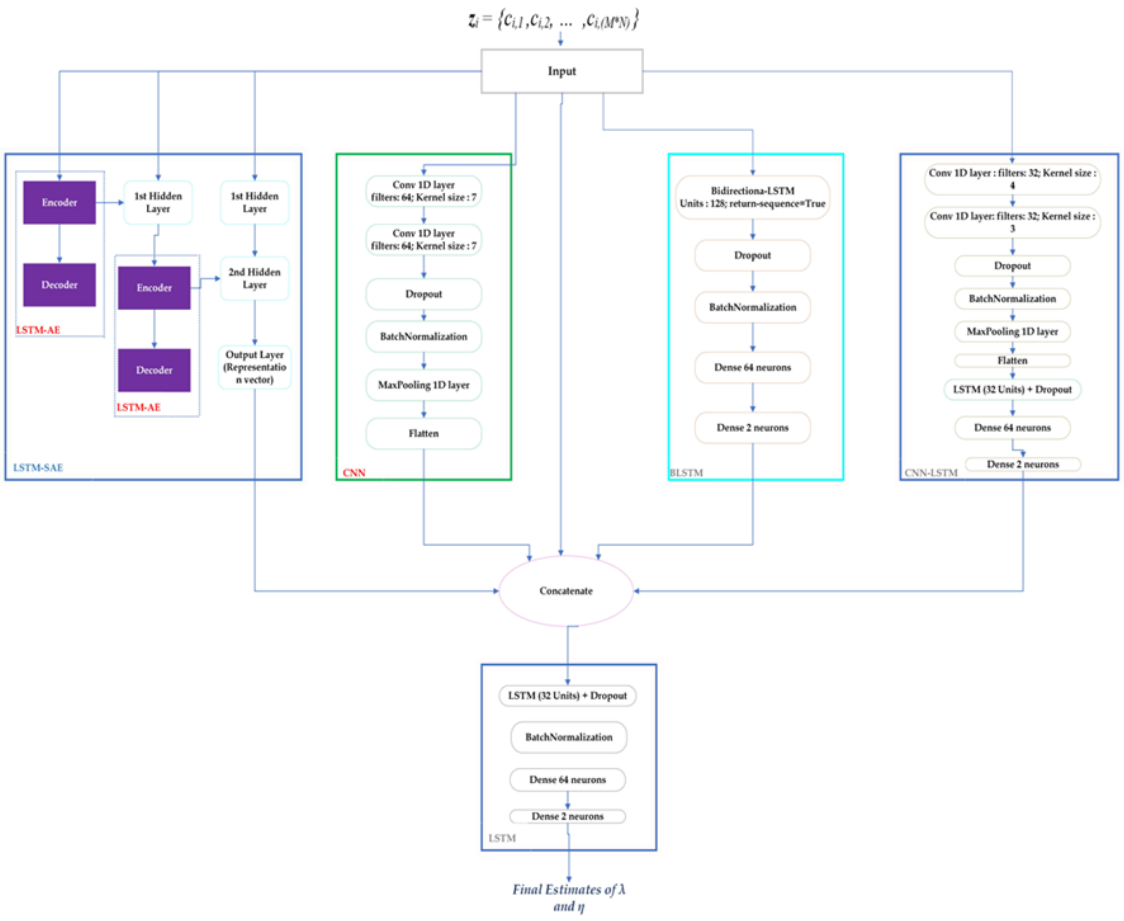

FIGURE 5.4: Architecture of the proposed multi-headed deep learning estimator for (λ, η).

The proposed estimator's multi-headed structure ensures that the complementary properties of different deep learning blocks are jointly exploited to enhance estimation accuracy. As will be shown in Section 5.5, this architecture outperforms classical and single-headed DL estimators, particularly under conditions of high clutter correlation and low signal-to-noise ratio.

5.4.4 Experimental Setup and Performance Evaluation Protocol

The evaluation of the proposed multi-headed deep learning-based estimator was conducted through a rigorous experimental protocol, designed to benchmark its performance against established classical estimators such as FOME, [z log(z)], and MLE. This section provides a consolidated description of the performance evaluation framework, leveraging the SIRV-based synthetic dataset previously introduced in Section 5.4.2 along with simulation Parameters and Data Partitioning

5.4.4.1 Model Training and Optimization

The neural estimator was trained using the Adam optimizer with an initial learning rate of 1×10^{-4}, and a Mean Squared Error (MSE) loss function. Batch size was set to 128, and early stopping was implemented with a patience of 10 epochs to prevent overfitting.

Regularization was achieved through dropout layers (rate = 0.3) and batch normalization, integrated throughout the architecture to ensure robustness and improve convergence. The final multi-headed deep learning-based estimator is obtained after different trials and errors process. Different architectures were considered with diverse models, layers, activation functions, and optimizers. The elaboration of the proposed intelligent estimator was validated after the comparison with several deep learning models. Table 5.1 summarises the overall mean square error (MSE) in the training and validation phases along with the configuration of the different architecture designs. Dropout and batch normalisation layers were incorporated for the above models to increase the generalization and improve prediction performances.

As depicted from table 5.1, it is clearly seen that the proposed DL-based estimator exhibits better performance than the other topologies for both the training and the validation datasets. The Adam optimizers exhibit the best performance in reason of its convergence rapidity along with the RMSprop with near-optimal performance.

TABLE 5.1: Comparison of different architectures configuration and MSE accuracy

Layers/Blocks	**Layers / Blocks Description**	**Number of neurons / filters / units**	**Activation function**	**Optimisers**	**Training accuracy (MSE)**	**Validation accuracy (MSE)**
Stacked LSTM	4 LSTM Layers + 2 Dense Layers	LSTM: 128, 64, 64, 32	Sigmoid	SGD	1.11	1.24
		Dense: 64, 2	ReLU	RMSprop	1.08	1.19
BLSTM	BLSTM + 2 Dense Layers	BLSTM: 128 units	Sigmoid	Adam	1.05	1.14
		Dense: 100, 2	Tanh	SGD	1.33	1.87
			ReLU	RMSprop	1.28	1.85
CNN-LSTM	2 CNN Layers + 2 LSTM Layers + 2 Dense Layers	Conv1D: 64, 64	Sigmoid	SGD	0.98	1.12
		LSTM: 32, 32	Tanh	RMSprop	0.86	0.89
		Dense: 64, 2	ReLU	Adam	0.86	0.87
Multi-Headed	CNN+ BLSTM+ CNN-LSTM + LSTM+ Dense	See Figure 5.4	ReLU	RMSprop	0.39	0.59
				Adam	0.31	0.34

5.4.4.2 Performance Metrics

To assess the accuracy and stability of the estimator, the following metrics were used:

- **MSE** between the true and estimated values of the shape (λ) and scale (η) parameters.
- **Bias**, computed as the mean deviation from ground truth values.
- **Standard deviation**, to quantify the estimator's sensitivity to random variations.
- **Runtime performance**, measured as the average inference time per prediction (averaged over 100 runs).

5.4.4.3 Baseline Estimators for Comparison

The following classical estimators were used for performance comparison:

- **FOME** [33],
- **[z log(z)] Estimator** [73],
- **MLE** [34].

All baseline methods assume i.i.d. clutter and do not consider correlation effects, thereby serving as references for evaluating robustness and generalization of the proposed deep learning-based solution under realistic, correlated clutter scenarios.

5.4.4.4 Computational Environment

Experiments were carried out on a computer equipped with:

- Intel Core i7-11700K CPU @ 3.6GHz,
- 16 GB RAM,
- NVIDIA RTX 2070 GPU (8 GB),
- TensorFlow 2.12.0 and Keras 2.12.0 (Python 3.10).

This configuration ensured adequate computational resources for training and evaluating the multi-headed model within reasonable time constraints. All training was conducted offline, and prediction runtime was recorded during inference-only execution phases.

5.5 Performance Assessment

With regards to the Deep Learning-based estimator presented above, we proceed in this section to carry out numerical simulations to evaluate the effectiveness of the proposed estimation method and confirm its applicability. For this purpose, we compare the results obtained using the multi-headed deep learning estimator against the existing FOME, [z log(z)], and the MLE methods by means of the MSE metric test. The dataset used for the test task is constructed following the same approach used for the training database. The window size is set to m=24 with N= 4,8 and 16 integrated pulses, and we considered different values of the correlation coefficient from data which tends to be uncorrelated to strongly correlated data (ρ = 0.01,0.1, 0.3, 0.6, 0.9,0.99) to apprehend the effect of the correlation on the estimation task. Pareto samples are generated for a range of [1.1-6] of λ so that different states of the clutter spikiness are included. In the overall test phase, the estimates of λ are averaged over n=1000 Monte Carlo trails, and

the computations were executed on an Intel® Core™ i5-3230M CPU @ 2.60 GHz RAM 8.0 GB.

As previously stated, for each value of λ between [1.1,6], we generate an m by N matrix with a fixed correlation coefficient. After that, the matrix is flattened to produce a single sample vector. This process is repeated 1000 times for each value of the couple (λ, η), resulting in a 50000*384 matrix in the test dataset.

Firstly, for low correlation coefficient (ρ=0.1), figure 5.5 illustrates the MSE curves of the FOME, [z log(z)], MLE, and the DL-based estimator for a window size of 24 and 16 integrated pulses. For very spiky clutter, comparable results are obtained for the four estimators with slightly enhanced performance of the DL-based estimator for $\lambda \leq 1.3$. As the spikiness of the clutter decreases, the proposed method exhibits improved performance compared to the other estimators with instabilities of the MLE and its closed-form counterpart [z log(z)] estimators. For the parameter η, it is clearly seen that the multi-headed deep learning attains the lowest error for $\eta < 8$ with overlapped curves of the other classical estimators. For large values of η, the MLE gives similar results as the DL and [z log(z)] estimators.

As the correlation coefficient increase (ρ=0.3), the estimation error of the [z log(z)], MLE, and the FOME increases while the DL estimator maintains its superiority for high values of the shape parameter as depicted from the figure 5.6 with a slight superiority of the [z log(z)] for λ =1.4. Note that for ρ=0.3 and large values of η, similar results to those obtained for ρ=0.1 are observed, and globally, precise estimation accuracy is achieved using the proposed estimation approach . For ρ =0.6, it is clearly depicted from the figure 5.7 that the proposed estimator achieves improved performance for the overall range of the shape and scale parameters. For highly correlated data ρ =0.9, the estimation accuracy of the multi-headed deep learning-based approach exhibits better performance than the classical approaches as illustrated in figure 5.8.

For a more informative illustration, figure 5.9 and 5.10 illustrate the bias of the shape and scale parameters for *ρ=0.1* and ρ=0.9 respectively. As expected, the results obtained from the bias criterion appears to be identical to those obtained by the MSE. Regardless of the correlation coefficient, for large shape parameter values, the DL estimator performance exceeds the log and moments based estimator. On the other hand, the bias confirm the superiority of the DL approach in estimating the scale parameter in the overall studied cases.

It is worth noticed that the Multi-headed deep learning approach estimates both the shape and scale parameters separately but simultaneously while the estimation of the scale parameter for the [z log(z)] and FOME depends on the estimate of the shape parameter. By contrast, for the case of the MLE, the shape parameter is estimated from the scale parameter. Thus, it is important to compare the results of the proposed approach against those obtained using moment of order half. figure 5.11. shows improved results of the multi-headed DL estimator for η >1.8 for uncorrelated data. On the other hand, for correlated pulses, figure 5.12. illustrates the effectivness of the proposed approach to estimate the scale parameter for the overall range of η values. Consequently, the estimation of one parameter from another using the existing approaches engenders imprecise results, leading to a considerable errors impeding the application of such estimators in real radar scenarios.

In order to visualise the impact of the correlation on the estimation task, MSE curves of the underlying estimators with ρ = 0.01 and ρ = 0.99 are gathered in the same figure. For *m=24* and *N = 16*, it is clearly seen in figure 5.16 that, the more the data are correlated, the less are the estimation stability and accuracy of the classical statistical estimators. Here, the DL based estimator attains the lowest MSE values and ensures the stability and the accuracy for both cases with enhanced results for the uncorrelated data.

With reduced number of pulses (=8, 4), the estimation performance degrades for the overall estimators compared to the results obtained for large number of pulses as depicted in figure 5.17 and 5.15 . Therefor, the proposed approach keeps its superiority for large and small number of pulses which approve its applicability in various situations.

The estimation accuracy depends highly on the clutter spikiness, the number of pulses, and the correlation between the samples. From the results obtained previously, it is worth stated that the classical FOME, [z log(z)] and MLE approaches are well suited for spiky uncorrelated echos and degraded performances are observed for strongly correlated data with high values of the shape parameter. The particularity of the proposed Multi-headed deep learning based estimator resides in its ability to estimate accurately and simultaneously the shape and the scale parameters for both correlated and uncorrelated data with reduced computational complexity.

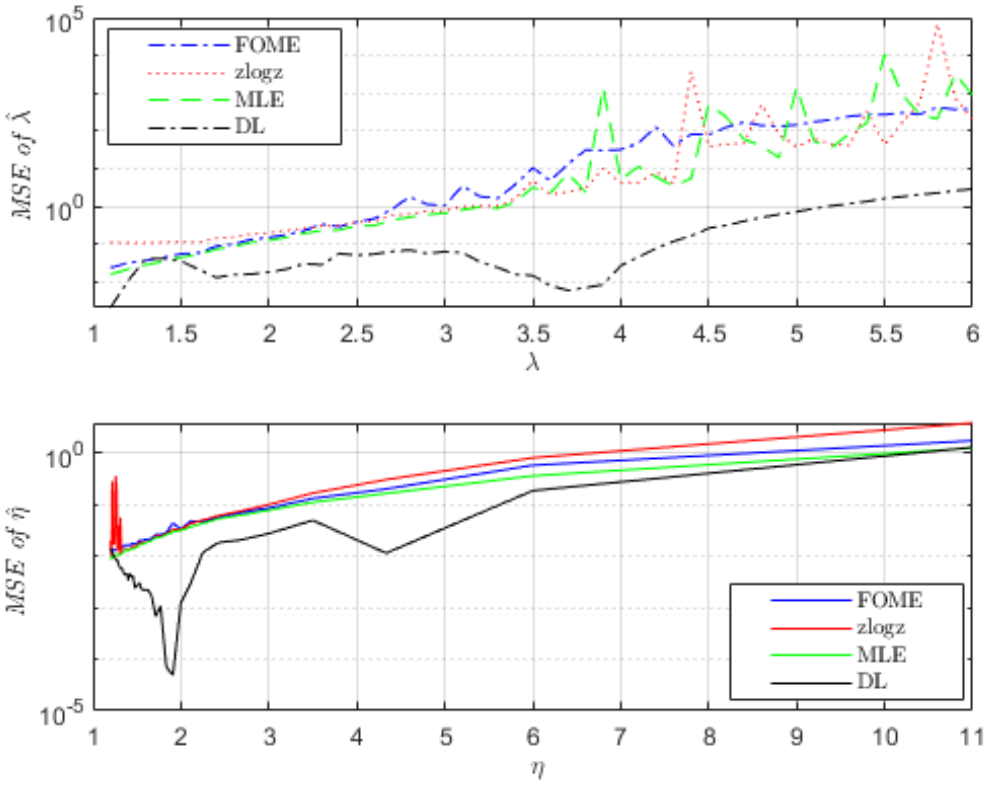

FIGURE 5.5: MSE of the estimates of $\boldsymbol{\lambda}$ and $\boldsymbol{\eta}$ for $\boldsymbol{\rho}$=0.1, m=24,and N=16

5.5.1 Computation Time and Efficiency

One of the critical factors in radar applications is real-time viability. The proposed DL estimator, though heavier in architecture, offers forward inference without iterative optimization. Nonetheless, for correlated echoes, the [z log(z)] method exhibits degraded performance, which again limit its applicability. The compromise between the time execution and the accuracy performance of the proposed intelligent algorithm confirms its superiority against the conventional FOME, [z log(z)] and MLE methods.

The estimation accuracy depends highly on the clutter spikiness, the number of pulses, and the correlation between the samples. From the results obtained previously, it is worth stating that the classical FOME, [z log(z)] and MLE approaches are well suited for spiky uncorrelated echoes and degraded performance is observed for strongly correlated data with high values of the shape parameter as depicted from table 5.2. The particularity of the proposed multi-headed deep learning-based estimator resides in its ability to estimate accurately and simultaneously the shape and the scale parameters for both correlated and uncorrelated data with reduced computational complexity.

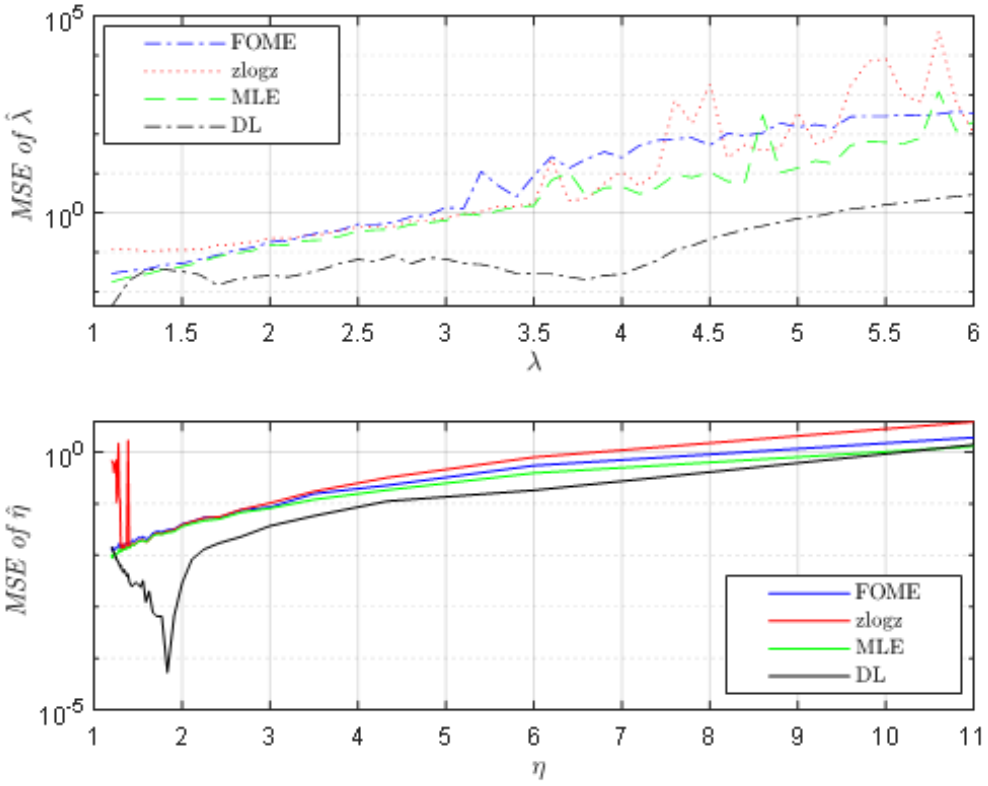

FIGURE 5.6: MSE of the estimates of $\boldsymbol{\lambda}$ and $\boldsymbol{\eta}$ for $\boldsymbol{\rho}$=0.3, m=24, and N=16

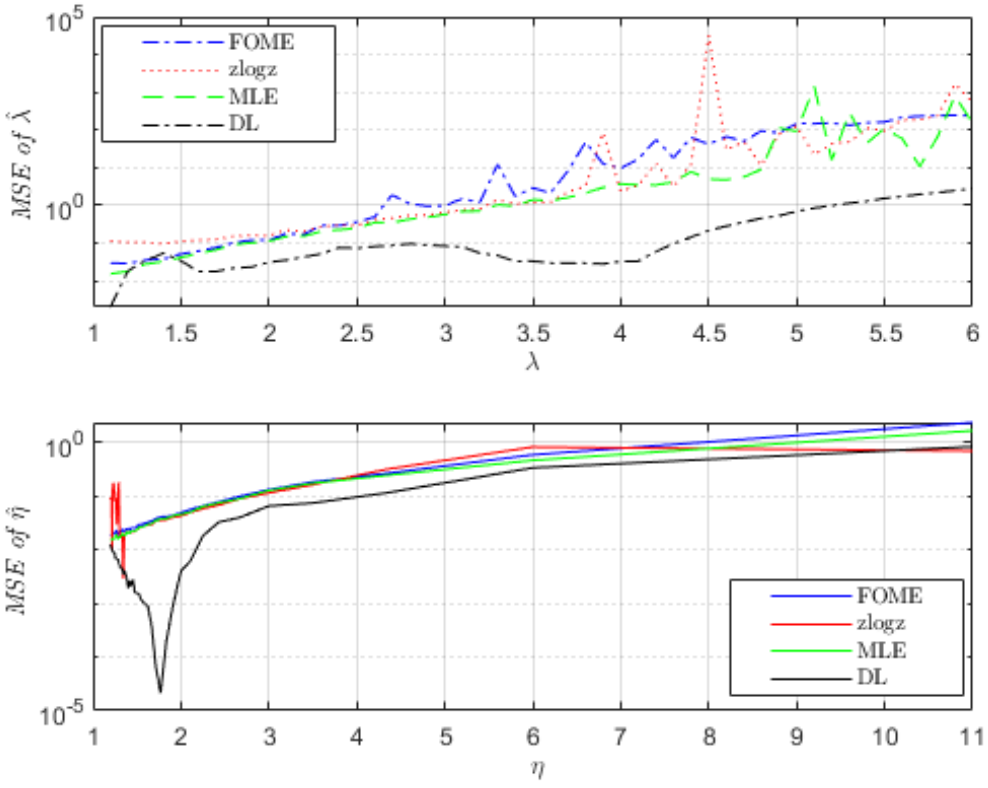

FIGURE 5.7: MSE of the estimates of $\boldsymbol{\lambda}$ and $\boldsymbol{\eta}$ for $\boldsymbol{\rho}$=0.6, m=24, and N=16

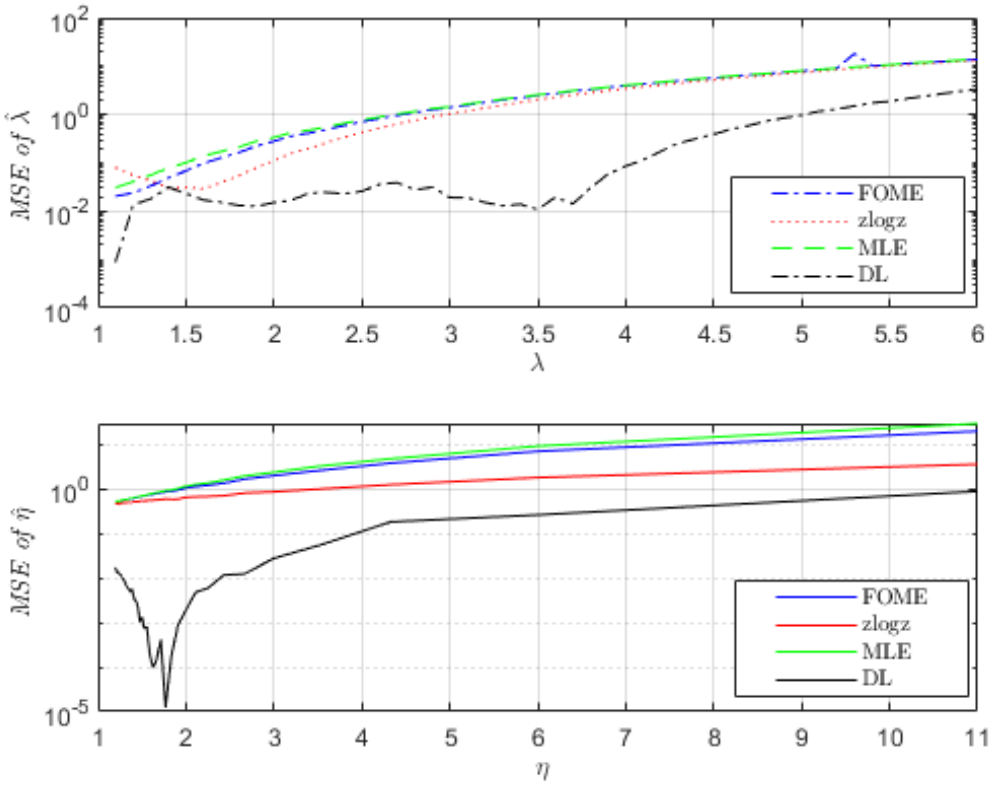

FIGURE 5.8: MSE of the estimates of $\boldsymbol{\lambda}$ and $\boldsymbol{\eta}$ for $\boldsymbol{\rho}$=0.9, m=24, and N=16

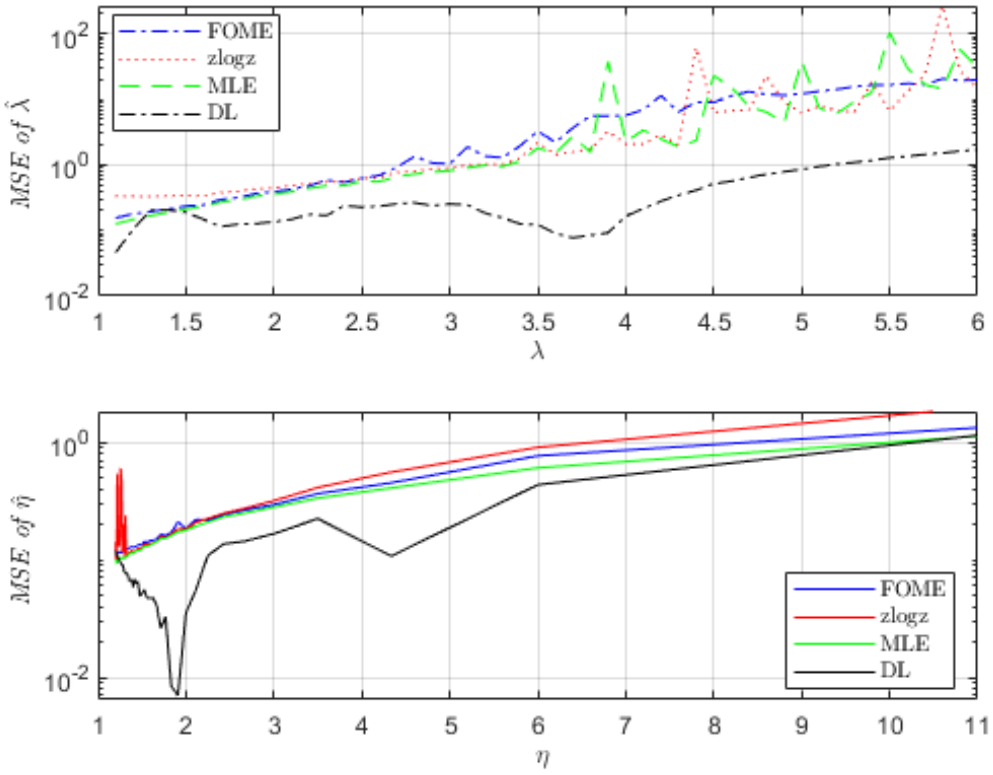

FIGURE 5.9: Bias of estimated $\boldsymbol{\lambda}$ and $\boldsymbol{\eta}$ for $\boldsymbol{\rho}$=0.1, m=24, and N=16

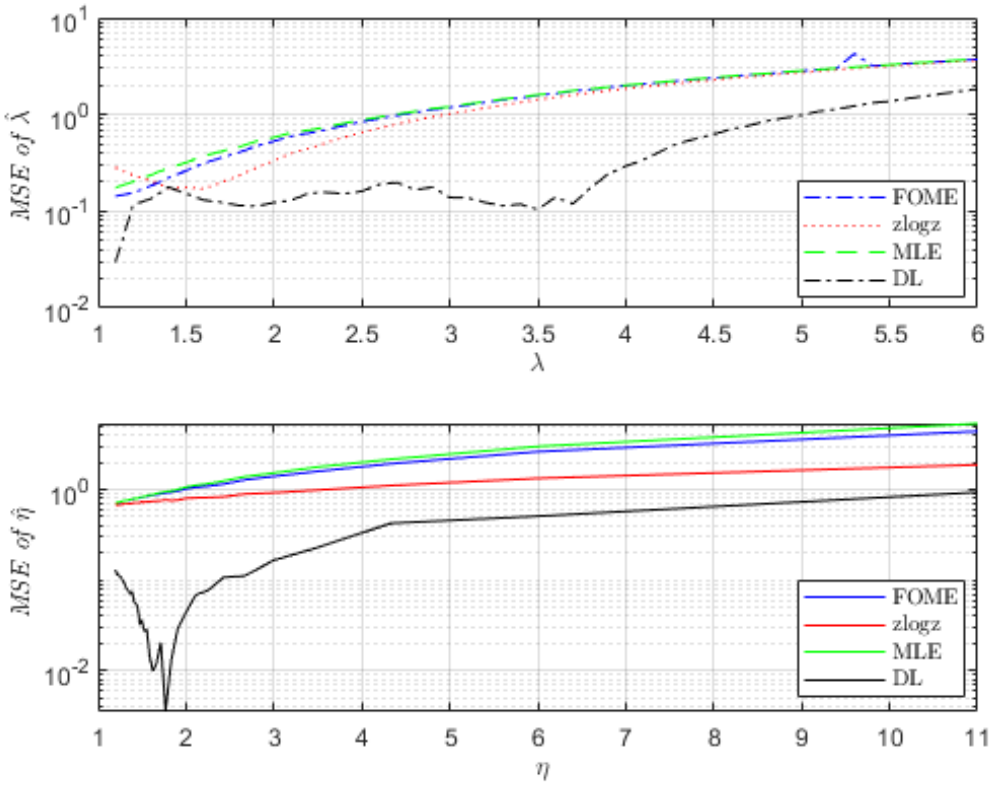

FIGURE 5.10: Bias of estimated $\boldsymbol{\lambda}$ and $\boldsymbol{\eta}$ for $\boldsymbol{\rho}$=0.9, m=24, and N=16

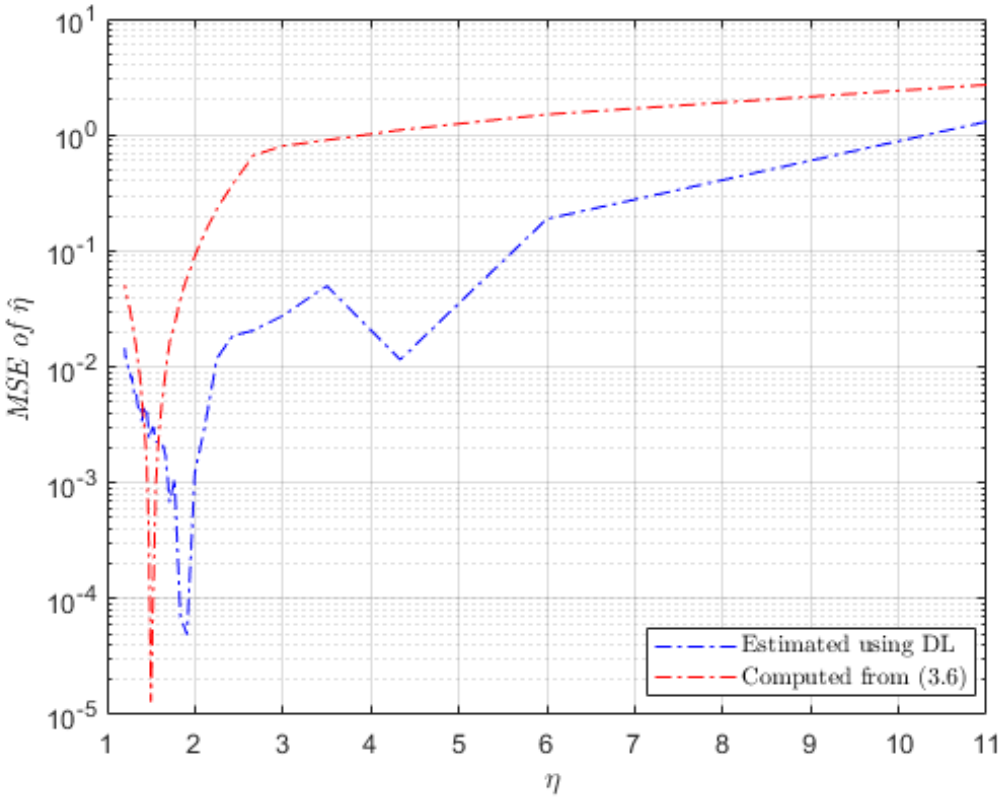

FIGURE 5.11: MSE comparison of estimated $\boldsymbol{\eta}$ for $\boldsymbol{\rho}$=0.1, m=24, and N=16

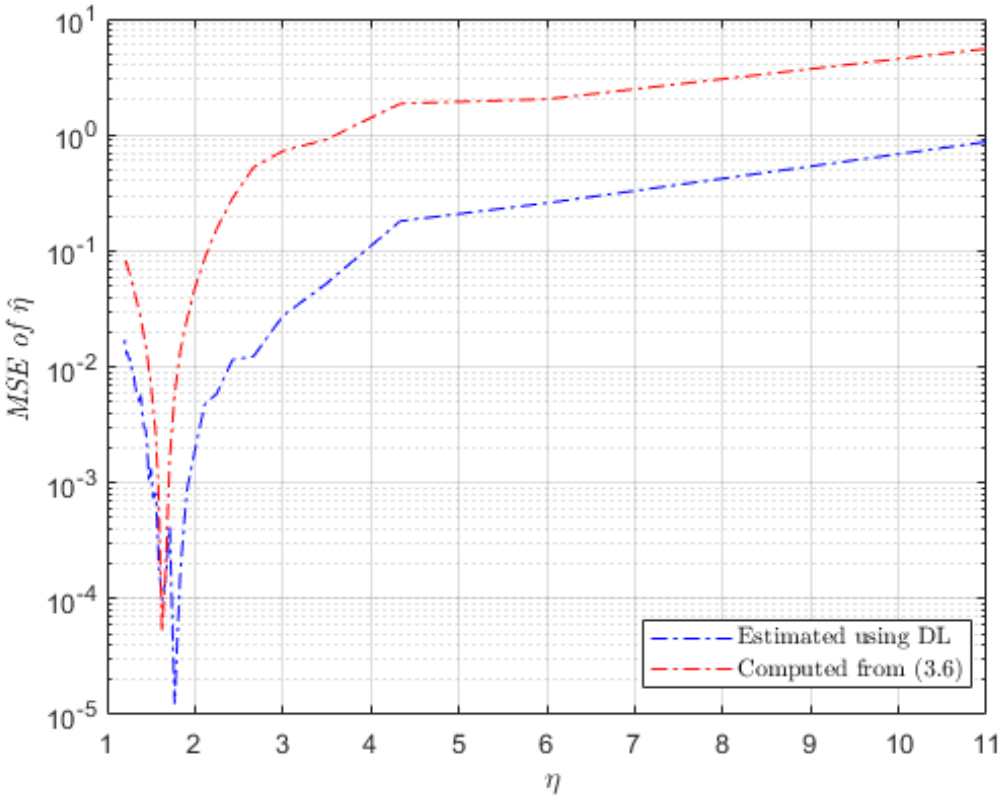

FIGURE 5.12: MSE comparison of estimated $\boldsymbol{\eta}$ for $\boldsymbol{\rho}$=0.9, m=24, and N=16

TABLE 5.2: Average elapsed time in (s) for different values of λ, 100 runs and different estimation method

Estimation approach / **Shape parameter**	**MLE**	**FOME**	**DLE**	**[z log(z)]**
$1.1 \leq \lambda < 3$	6.1753	4.334	34.502×10^{-3}	15.375×10^{-3}
$3 \leq \lambda < 4.5$	6.4641	4.869	35.925×10^{-3}	17.781×10^{-3}
$4.5 \leq \lambda < 6$	8.7823	6.211	37.638×10^{-3}	19.224×10^{-3}

5.5.1.1 Summary

Overall, the multi-headed DL estimator outperforms conventional approaches across all key dimensions:

- **Accuracy:** Achieves the lowest MSE across varying λ, η, and ρ.
- **Robustness:** Demonstrates consistent performance regardless of clutter spikiness or correlation.
- **Efficiency:** Fast inference compatible with near real-time requirements.
- **Generalization:** Maintains high accuracy for both low and high pulse counts.

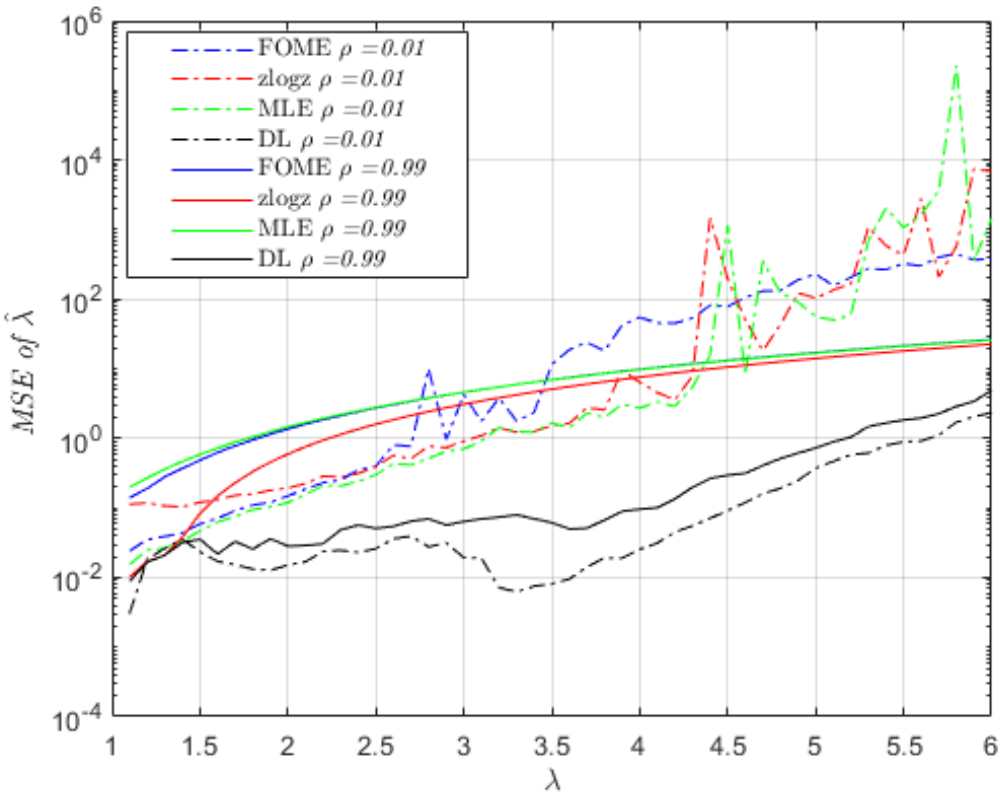

FIGURE 5.13: MSE comparison of estimated $\boldsymbol{\lambda}$ for $\boldsymbol{\rho}$=0.01(dashed lines) and $\boldsymbol{\rho}$=0.99 (solid lines) with m=24, and N=16

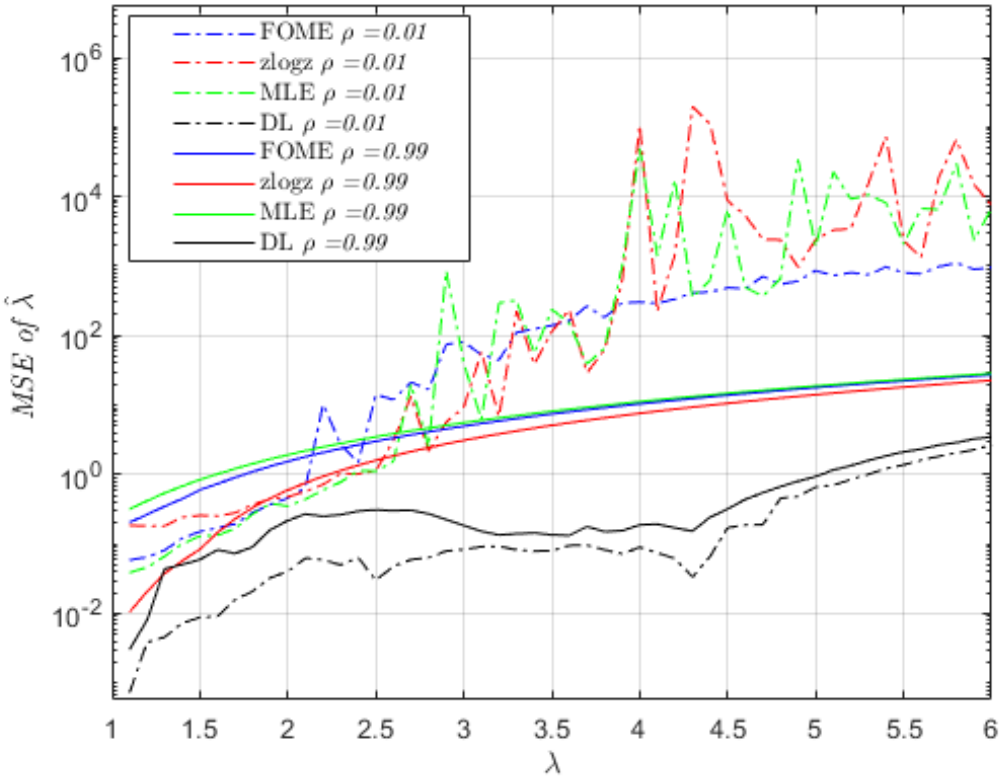

FIGURE 5.14: MSE comparison of estimated $\boldsymbol{\lambda}$ for $\boldsymbol{\rho}$=0.01(dashed lines) and $\boldsymbol{\rho}$=0.99 (solid lines) with m=24, and N=8

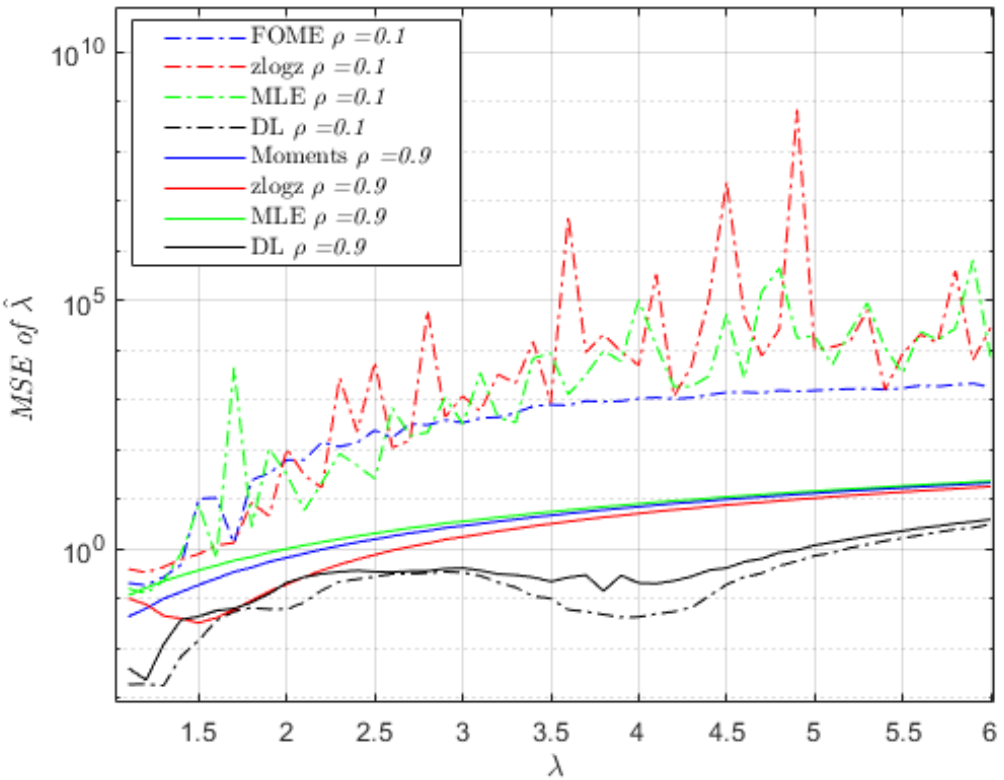

FIGURE 5.15: MSE comparison of estimated $\boldsymbol{\lambda}$ for $\boldsymbol{\rho}$=0.1(dashed lines) and $\boldsymbol{\rho}$=0.9 (solid lines) with m=24, and N=4

These outcomes validate the potential of deep learning-based estimators in complex radar environments where traditional methods fall short due to structural assumptions and iterative complexity.

5.5.2 CFAR Analysis and GLRT-LTD Detection Performance

To further evaluate the applicability of the proposed multi-headed deep learning-based estimator, we investigate its impact on the performance of the GLRT-LTD (Generalized Likelihood Ratio Test - Linear Threshold Detector) in correlated Pareto Type II clutter. This analysis follows the same simulation setup as in the reference study: a coherent processing interval of $m = 16$ pulses, $N = 24$ reference cells, and varying correlation coefficients $\rho = 0.01$ (weakly correlated) and $\rho = 0.9$ (highly correlated).

In Order to assess detection performance of the GLRT-LTD, inspection of the CFAR property is necessary. Figures 5.16 and 5.17 display the probability of false alarm (P_{FA}) as a function of the shape parameter λ and threshold factor T under uncorrelated and correlated scenarios, respectively. Parameter estimation is performed using the [z log(z)] method.

As shown in Figure 5.16 for $\rho = 0.01$, the adaptive GLRT-LTD detector preserves the CFAR texture property. A value of $T = 8.55$ dB corresponds to a $P_{FA} = 10^{-3}$. Figure 5.17 confirms similar CFAR behavior in the highly correlated case ($\rho = 0.9$), demonstrating that the GLRT-LTD maintains its false alarm regulation capability across varying degrees of correlation.

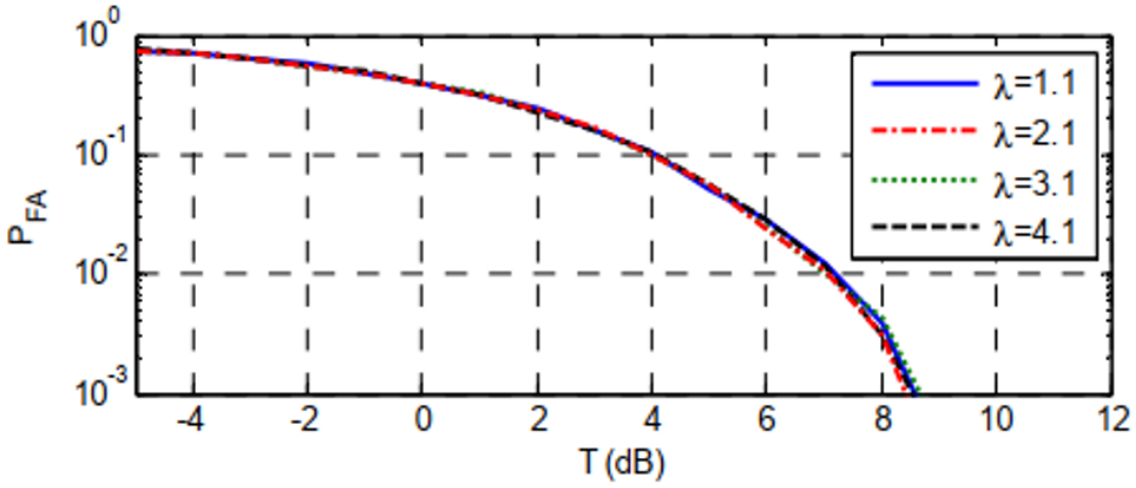

FIGURE 5.16: CFAR texture property of the GLRT-LTD detector for $m = 16$, $\rho = 0.01$, and $N = 24$.

As shown in Fig. 5.18, under the weak correlation case ($\rho = 0.01$), the DL-based estimator closely follows the detection performance of traditional estimators such as MLE and [z log(z)]. This indicates that in nearly uncorrelated clutter scenarios, the DL model retains its robustness and yields detection probabilities.

In contrast, for strongly correlated clutter ($\rho = 0.9$), as shown in Fig. 5.19, the DL-based estimator also outperforms classical estimators, This is attributed to its capability to estimate the shape and scale parameters with higher accuracy under correlation, which in turn enhances the detection capability.

These observations validate the effectiveness of integrating the DL estimator into the GLRT-LTD framework, enabling enhanced detection performance without compromising statistical robustness, particularly in challenging correlated environments.

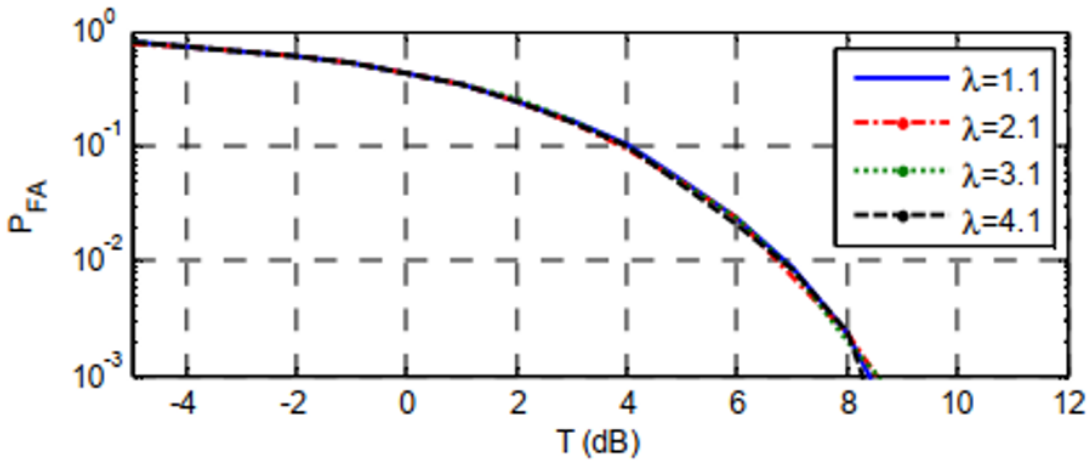

FIGURE 5.17: CFAR texture property of the GLRT-LTD detector for $m = 16$, $\rho = 0.9$, and $N = 24$.

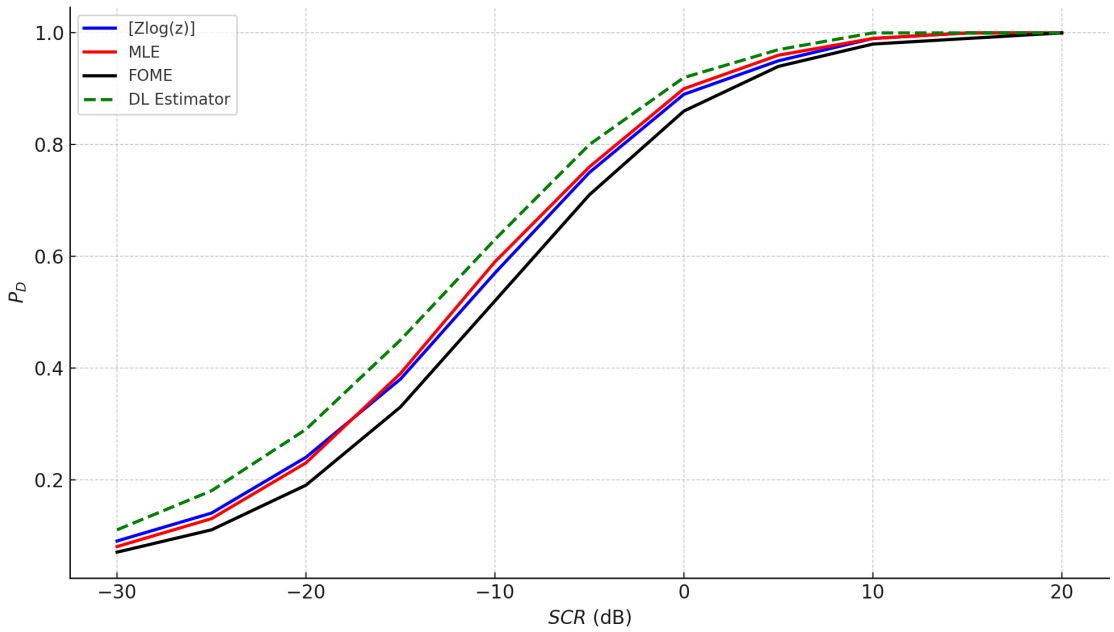

FIGURE 5.18: Detection performance of the GLRT-LTD detector with $\lambda = 1.1$, $m = 16$, $\rho = 0.01$, and $N = 24$

5.6 Conclusion and Future Directions

This chapter has addressed the challenging problem of accurately estimating the shape (λ) and scale (η) parameters of Pareto Type II clutter, especially under the realistic conditions of correlated radar returns. Traditional clutter parameter estimation techniques, such as the FOME, [z log(z)] estimator, and MLE have demonstrated reasonable effectiveness in the context of uncorrelated and highly spiky clutter. However, their performance significantly degrades as the correlation in clutter increases or as the shape parameter grows, leading to instability, high bias, or slow convergence.

To overcome these limitations, this work proposed a novel multi-headed deep learning-based

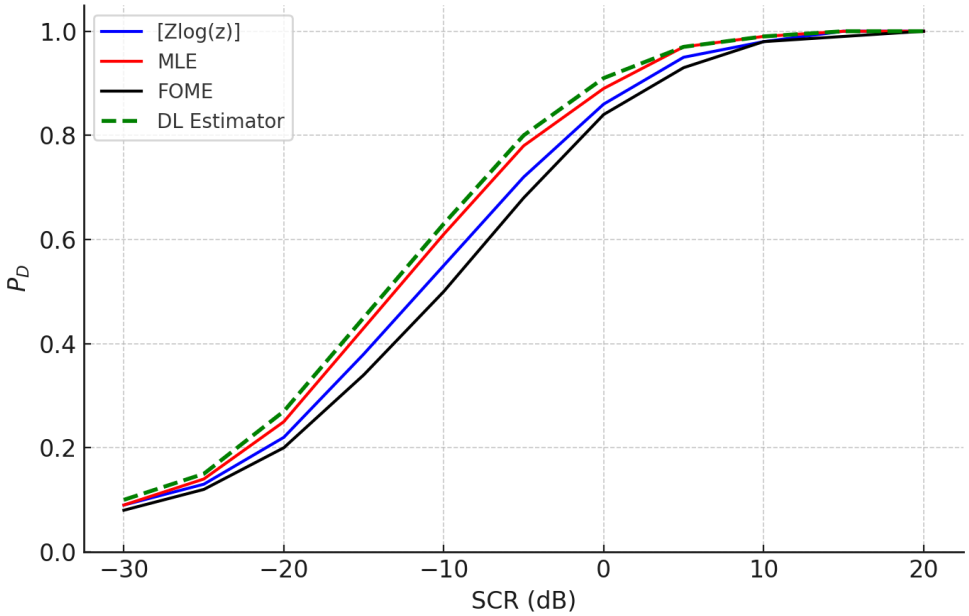

FIGURE 5.19: Detection performance of the GLRT-LTD detector with $\lambda = 1.1$, $m = 16$, $\rho = 0.9$, and $N = 24$, including the DL-based estimator.

estimator, leveraging both supervised and semi-supervised learning paradigms. The model architecture combines diverse deep neural network modules, including LSTM Autoencoders, CNNs, BLSTMs, and hybrid CNN-LSTM layersin a parallel structure, enabling it to capture intricate temporal, spectral, and statistical features of radar returns. Each component contributes distinct yet complementary insights, and their outputs are fused into a joint prediction block designed to simultaneously estimate the Pareto clutter parameters.

A synthetic dataset, derived from the SIRV model, was generated to simulate a wide variety of clutter environments, ranging from nearly i.i.d. spiky returns to strongly correlated heavy-tailed clutter. Numerical experiments validated the effectiveness of the proposed estimator. Specifically, the results showed that:

- The deep learning-based estimator consistently outperformed classical methods across all evaluated metrics, including MSE and bias, particularly in high-correlation scenarios and for high values of λ.
- Unlike classical estimators that rely on sequential estimation of λ and η, the DL-based approach performs simultaneous and independent prediction, thereby reducing error propagation and improving robustness.
- The inference speed of the proposed estimator is well suited for real-time systems. While the execution time slightly exceeds that of closed-form estimators like [z log(z)], it remains significantly faster than iterative methods such as MLE, with the added benefit of stability and adaptability.

Future Directions

While the results are promising, several key areas for future investigation and development remain:

- **Threshold Estimation for CFAR Detectors:** A critical extension involves integrating a direct DL-based mapping from estimated clutter parameters to the optimal detection threshold. This would support a full end-to-end learning-based CFAR detector that adapts in real time to environmental statistics.
- **Joint Estimation under AGN:** The presence of AGN, especially in low SNR regimes, distorts clutter statistics and hampers estimation accuracy. Future models should explicitly incorporate noise robustness mechanismspotentially via denoising autoencoders or noise-aware loss functions.
- **Transfer Learning for Operational Data:** Training on simulated data presents generalization challenges. Future work could explore fine-tuning the model using transfer learning on real radar datasets or through domain adaptation methods to bridge the simulation-to-reality gap.

In conclusion, this work demonstrates the power of deep learning to bridge the limitations of classical statistical estimation in radar systems. The proposed multi-headed framework, grounded in the physics of SIRV clutter and informed by the capabilities of modern DL architectures, offers a robust, fast, and generalizable solution for parameter estimation in correlated Pareto Type II clutter environments. It paves the way for the next generation of intelligent, adaptive CFAR detectors ready to operate under the most demanding real-world conditions.

6 Improved Distributed GO-CFAR and SO-CFAR Detectors Using Moth Flame Algorithm

Contents

6.1 Introduction

Radar detection in cluttered environments requires a careful balance between sensitivity to targets and immunity to false alarms. To address this, CFAR algorithms are extensively deployed in both centralized and distributed radar systems. Classical CFAR techniques, such as CA-CFAR, GO-CFAR, SO-CFAR, and Ordered-Statistics CFAR (OS-CFAR), achieve their functionality by adjusting detection thresholds based on local statistics derived from reference cells surrounding the cell under test (CUT) [42].

Distributed radar architectures are increasingly adopted to improve spatial coverage and robustness through cooperation among multiple sensors. In such systems, CFAR processing is typically performed locally at each sensor, followed by binary decision fusion at a central node using rules such as AND, OR, or majority voting [51]. However, distributed operation introduces new degrees of freedom (e.g., sensor-dependent reference window sizes and threshold multipliers), and the global CFAR constraint must be satisfied at the fusion center. As a result, locally tuned parameters can become globally suboptimal when sensors operate under non-identical conditions (e.g., different M_i, different local SNRs, or non-uniform interference) [41].

This chapter formulates the tuning of distributed GO-CFAR and SO-CFAR threshold multipliers as a multivariable optimization problem: determine the set $\{\alpha_i\}_{i=1}^{n}$ that maximizes

the global probability of detection while enforcing a prescribed global probability of false alarm. Due to the nonlinear and coupled expressions of P_d and P_f under fusion, analytical optimization is generally intractable; hence, population-based metaheuristics are an attractive solution strategy [44, 55, 56].

In this context, the Moth-Flame Optimization (MFO) algorithm [82] is employed to optimize distributed GO-CFAR and SO-CFAR parameters. This chapter introduces a novel framework for optimizing distributed GO-CFAR and SO-CFAR detectors using the MFO algorithm. The objective is to jointly optimize the local CFAR scaling factors and the configuration of the fusion center to achieve superior global detection performance. Unlike previous approaches that tune detectors individually or without considering the global objective function, the proposed method adopts a centralized learning strategy with decentralized application, ensuring that each sensor's configuration contributes constructively to the network-wide detection outcome.

The main interest of using a population-based optimizer in this setting is not to replace analytical CFAR design, but to address the *coupled* and *multivariable* nature of distributed threshold tuning under fusion rules. In low-dimensional and homogeneous configurations, several metaheuristics may converge to nearly identical solutions. However, in non-identical sensor configurations—where each sensor has its own M_i and operating conditions—the optimization landscape becomes more complex and algorithmic exploration/exploitation behavior becomes more influential on the final detection performance.

Key contributions of this chapter include:

- Neyman–Pearson constrained formulation for tuning distributed CFAR multipliers $\{\alpha_i\}$ at the network level.
- An MFO-based optimization framework for distributed GO-CFAR and SO-CFAR detectors in Gaussian disturbance.
- A comparative study against Grey Wolf Optimization (GWO) and Biogeography-Based Optimization (BBO) using the same experimental settings and reported results.
- Detailed simulation studies showing the effectiveness of the optimized GO-CFAR and SO-CFAR detectors under different radar configurations and clutter statistics.

The remainder of this chapter is organized as follows. Section 6.2 introduces the distributed CFAR detection model, including the local GO-CFAR/SO-CFAR formulations and the fusion rules at the decision center. Section 6.3 formulates the network-level threshold optimization problem under the Neyman–Pearson criterion and defines the adopted objective function. Section 6.4 presents the Moth–Flame Optimization (MFO) algorithm and its implementation for distributed CFAR parameter tuning. Section 6.5 reports the simulation results for identical and non-identical configurations and compares MFO with GWO and BBO. Section 6.6 discusses the results and practical considerations, and Section 6.7 concludes the chapter with directions for future work.

6.2 Distributed CFAR Detection Model

6.2.1 Centralized GO-CFAR and SO-CFAR (Gaussian Disturbance)

GO-CFAR and SO-CFAR are effective CFAR algorithms used to address heterogeneous scenarios such as clutter edges and multiple-target interference. The general decision rule compares the CUT amplitude with an adaptive threshold derived from the reference window statistics.

For Gaussian clutter, the adaptive threshold T is computed using two reference windows U and V (each of size $M/2$). A detection is declared if the CUT exceeds the threshold scaled by a

factor α, as illustrated in Figure 6.1. The goal is to maintain a constant false alarm probability (typically 10^{-3} to 10^{-8}).

Under the assumption that reference cells contain i.i.d. Gaussian clutter samples $x_1, x_2, \ldots, x_M$, the expressions for the centralized GO-CFAR P_{FA} and P_D are [40, 41]:

$$\begin{cases} P_{FA}^{GO} &= 2(1+\alpha)^{-M/2} - 2\sum\limits_{i=0}^{M/2-1}\binom{M/2+i-1}{i}(2+\alpha)^{-(M/2+i)} \\ P_D^{GO} &= 2\left(1+\dfrac{\alpha}{1+\mathrm{SNR}}\right)^{-M/2} - 2\sum\limits_{i=0}^{M/2-1}\binom{M/2+i-1}{i}\left(2+\dfrac{\alpha}{1+\mathrm{SNR}}\right)^{-(M/2+i)} \end{cases} \tag{6.1}$$

Similarly, for centralized SO-CFAR:

$$\begin{cases} P_{FA}^{SO} &= 2(2+\alpha)^{-M/2}\sum\limits_{i=0}^{M/2-1}\binom{M/2+i-1}{i}(2+\alpha)^{-i} \\ P_D^{SO} &= 2\left(2+\dfrac{\alpha}{1+\mathrm{SNR}}\right)^{-M/2}\sum\limits_{i=0}^{M/2-1}\binom{M/2+i-1}{i}\left(2+\dfrac{\alpha}{1+\mathrm{SNR}}\right)^{-i} \end{cases} \tag{6.2}$$

The centralized GO-CFAR and SO-CFAR expressions are briefly recalled to characterize the *local* detector performance (P_{FA_i}, P_{D_i}), which is then embedded into the distributed fusion model to obtain global (P_f, P_d).

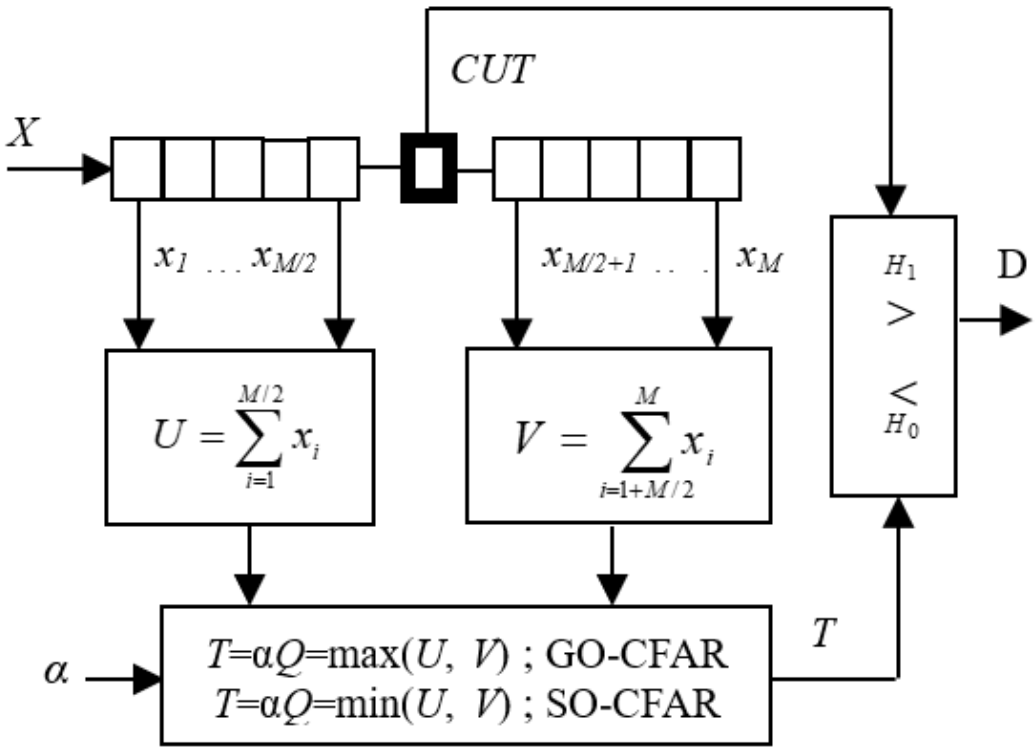

FIGURE 6.1: Centralized GO-CFAR and SO-CFAR processors.

6.2.2 Distributed CFAR Architecture and Fusion Rules

Distributed CFAR systems comprise n local CFAR processors, each producing a binary decision $D_i \in \{0, 1\}$ that is transmitted to a fusion center where a fusion rule is applied (Figure 6.2). This architecture enhances robustness to local variations and improves coverage, but requires careful tuning of local multipliers α_i.

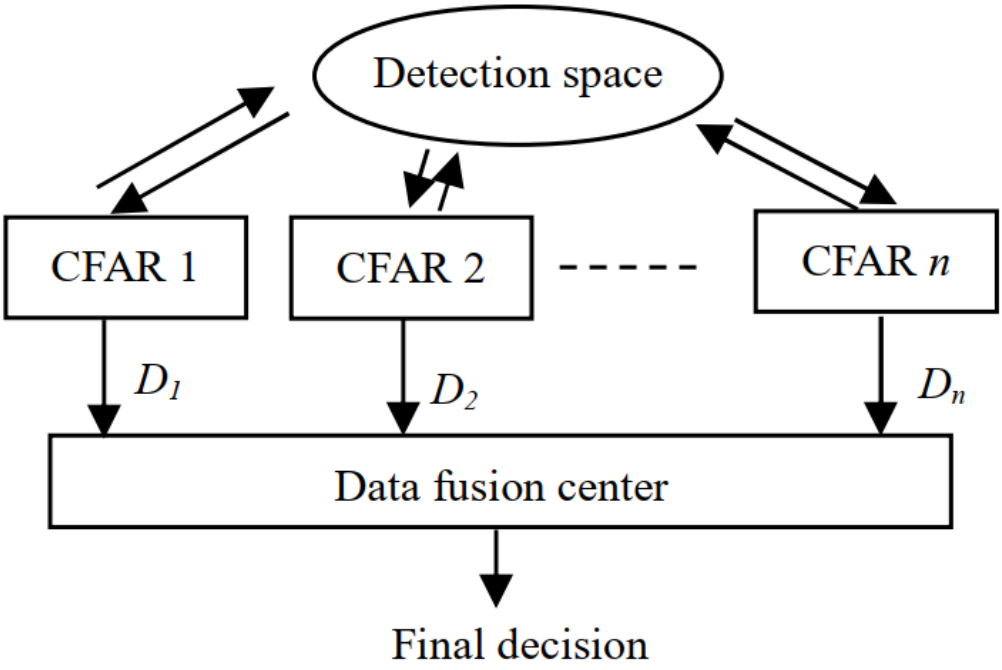

FIGURE 6.2: Distributed CFAR processor system .

For sensor i with local window size M_i, local multiplier α_i, and local SNR S_i, the local probabilities for GO-CFAR and SO-CFAR follow the same structure as the centralized expressions.

Fusion rules :

AND fusion rule: all sensors must detect the target:

$$\begin{cases} P_f &= \prod_{i=1}^{n} P_{FA_i} \\ P_d &= \prod_{i=1}^{n} P_{D_i} \end{cases} \tag{6.3}$$

OR fusion rule: at least one sensor detects the target:

$$\begin{cases} P_f &= 1 - \prod_{i=1}^{n} (1 - P_{FA_i}) \\ P_d &= 1 - \prod_{i=1}^{n} (1 - P_{D_i}) \end{cases} \tag{6.4}$$

MAJORITY fusion rule: for $n = 3$:

$$\begin{cases} P_f &= P_{FA_1}P_{FA_2} + P_{FA_1}P_{FA_3} + P_{FA_2}P_{FA_3} - 2P_{FA_1}P_{FA_2}P_{FA_3} \\ P_d &= P_{D_1}P_{D_2} + P_{D_1}P_{D_3} + P_{D_2}P_{D_3} - 2P_{D_1}P_{D_2}P_{D_3} \end{cases} \tag{6.5}$$

Because P_f and P_d are nonlinear and coupled across sensors through the fusion rule, best detection performance requires *joint* optimization of $\{\alpha_i\}_{i=1}^n$, especially in the non-identical case (different M_i and/or different S_i). This motivates the optimization framework described next.

6.3 Threshold Control and Optimization Objectives

The global design objective follows the Neyman–Pearson principle: maximize the global detection probability while enforcing a prescribed global false alarm probability α_0. Formally, the distributed threshold selection problem can be written as

$$\max_{\alpha_1,\ldots,\alpha_n} P_d^{(global)}(\alpha_1,\ldots,\alpha_n;\mathbf{S}) \quad \text{s.t.} \quad P_f^{(global)}(\alpha_1,\ldots,\alpha_n) = \alpha_0, \tag{6.6}$$

where $\mathbf{S} = [S_1,\ldots,S_n]$ denotes the local SNR values at each sensor.

The global probability of detection $P_d^{(global)}$ depends on the local SNR values $\{S_i\}$, i.e., $P_d^{(global)} = P_d^{(global)}(\alpha_1,\ldots,\alpha_n; S_1,\ldots,S_n)$. In practice, the SNR may be unknown and time-varying

- **Average (expected) detection optimization:**

$$\max_{\alpha_1,\ldots,\alpha_n} \mathbb{E}_{\mathbf{S}}\left[P_d^{(global)}(\alpha_1,\ldots,\alpha_n;\mathbf{S})\right] \quad \text{s.t.} \quad P_f^{(global)}(\alpha_1,\ldots,\alpha_n) = \alpha_0. \tag{6.7}$$

- **Robust (worst-case) detection optimization:**

$$\max_{\alpha_1,\ldots,\alpha_n} \min_{\mathbf{S}\in\mathcal{S}} P_d^{(global)}(\alpha_1,\ldots,\alpha_n;\mathbf{S}) \quad \text{s.t.} \quad P_f^{(global)}(\alpha_1,\ldots,\alpha_n) = \alpha_0. \tag{6.8}$$

The constrained optimization problem is implemented using a penalty-based formulation consistent with the Neyman–Pearson criterion. The fitness function minimized by the metaheuristic algorithms is defined as

$$\mathrm{Fit}(\boldsymbol{\alpha}) = \left|1 - P_d^{(global)}(\boldsymbol{\alpha})\right| + \frac{1}{\alpha_0}\left|P_f^{(global)}(\boldsymbol{\alpha}) - \alpha_0\right|. \tag{6.9}$$

The first term promotes maximization of the global detection probability (since minimizing $|1 - P_d|$ drives P_d toward unity). The second term penalizes any deviation from the prescribed false alarm probability. The scaling factor $1/\alpha_0$ ensures strong enforcement of the CFAR constraint, particularly for small values of $\alpha_0 \in \{10^{-4}, 10^{-6}, 10^{-8}\}$.

This formulation corresponds to a weighted penalty approach where

$$w_1 = 1, \qquad w_2 = \frac{1}{\alpha_0},$$

thus prioritizing strict adherence to the global false alarm requirement during the search process.

6.3.1 Search Domain and Experimental Configuration

The optimization process uses the following configuration:

- Search domain: $\alpha_i \in [0.001, 10.001]$,
- Desired global false alarm levels: $\alpha_0 \in \{10^{-4}, 10^{-6}, 10^{-8}\}$,
- Window sizes: identical case ($M = 24$) and non-identical case ($M_1 = 16, M_2 = 24, M_3 = 32$),
- SNR setting for reported tables: $S = 20$ dB,
- Fusion rule: primarily OR.

Because $P_d^{(global)}$ and $P_f^{(global)}$ are nonlinear and coupled across sensors through the fusion rule, analytical optimization is intractable. This motivates the use of population-based metaheuristics.

6.4 Moth-Flame Optimization (MFO) Method

The Moth-Flame Optimization (MFO) algorithm is a population-based metaheuristic introduced by Mirjalili in 2015 [82]. In this chapter, each moth represents a candidate threshold vector

$$\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_n],$$

while flames represent the best solutions found so far.

MFO updates candidate solutions using a logarithmic spiral mechanism that balances global exploration and local exploitation. The position update rule of the i^{th} moth around the j^{th} flame is defined as

$$\vec{M}_i^{\ell+1} = \vec{D}_i \, e^{bt} \cos(2\pi t) + \vec{F}_j, \tag{6.10}$$

where $\vec{M}_i^{\ell+1}$ denotes the updated position at iteration $\ell + 1$, $\vec{F}_j$ is the corresponding flame position, $\vec{D}_i = \vec{F}_j - \vec{M}_i^{\ell}$ is the distance vector, b controls the spiral shape, and $t \in [-1, 1]$ is a random parameter.

To progressively shift from exploration to exploitation, the number of flames decreases with the iteration index according to

$$\text{Flames}_\ell = \text{round}\left(N_m - \frac{\ell(N_m - 1)}{T}\right), \tag{6.11}$$

where N_m is the population size and T is the maximum number of iterations.

6.4.1 Algorithmic Parameters Used in This Work

The experiments reported in this chapter use:

- Population size: $N_m = 250$ moths,
- Maximum iterations: $T = 500$,
- Search dimension: $n = 3$ (number of sensors),
- Bounds: $\alpha_i \in [0.001, 10.001]$.

6.4.2 Computational Complexity

Each iteration evaluates the fitness of all N_m moths. Since computing the global (P_d, P_f) involves n local detectors, the overall computational complexity scales as

$$\mathcal{O}(N_m\, T\, n).$$

With $N_m = 250$, $T = 500$, and $n = 3$, the total number of fitness evaluations is 1.25×10^5. Due to the closed-form expressions of GO-CFAR and SO-CFAR probabilities, each evaluation involves lightweight algebraic operations; consequently, the computational burden remains moderate (typically a few seconds per optimization run in MATLAB), making the approach suitable for offline calibration or periodic re-optimization.

6.4.3 Implementation Overview

Algorithm 4 MFO for CFAR Threshold Selection

Require: Population size N_m, iterations T, bounds of α_i
Ensure: Optimal threshold vector $\boldsymbol{\alpha}^\star$
1: Initialize moth population randomly within bounds
2: **for** $\ell = 1$ to T **do**
3: Update number of flames using Eq. (6.11)
4: Evaluate the fitness $\mathrm{Fit}(\boldsymbol{\alpha})$ for all moths
5: Sort moths by fitness and update flame positions
6: Update moth positions using Eq. (6.10)
7: **end for**
8: **return** Best flame as $\boldsymbol{\alpha}^\star$

6.5 Distributed CFAR Detection Comparisons

This section reports the numerical results of the MFO, GWO, and BBO optimization of distributed GO-CFAR and SO-CFAR detectors under Gaussian disturbance and Swerling I target assumptions.

6.5.1 Identical Distributed GO-CFAR Detectors (OR Fusion Rule)

Table 6.1 reports the scale factors and detection probabilities for identical GO-CFAR detectors with $M = 24$, $S = 20$ dB, and OR fusion rule, for different numbers of sensors ($n = 3, 5, 7$) and false alarm constraints. The results show that all three metaheuristic algorithms converge to very similar threshold values and detection probabilities, particularly for stringent false alarm constraints (10^{-6} and 10^{-8}). For moderate false alarm levels (10^{-4}), MFO provides slightly higher detection probabilities compared to GWO and BBO.

It is also observed that the global detection probability increases significantly as the number of sensors grows. For example, at $\alpha_0 = 10^{-4}$, P_d increases from approximately 0.90 for $n = 3$ to more than 0.99 for $n = 7$. This behavior is consistent with the OR fusion rule, which benefits from the increased spatial diversity provided by additional sensors.

TABLE 6.1: Scale factors found for identical GO-CFAR detectors using MFO, GWO and BBO algorithms with $M = 24$, $S = 20$ dB and OR fusion rule.

Optimiz-tool	α_0	$n = 3$	$n = 5$	$n = 7$
MFO algorithm	10^{-4}	$t_i = 0.9593$ $P_d = 0.9033$	$t_i = 1.0190$ $P_d = 0.9729$	$t_i = 1.0589$ $P_d = 0.9920$
	10^{-6}	$t_i = 1.5530$ $P_d = 0.7457$	$t_i = 1.6265$ $P_d = 0.8839$	$t_i = 1.6758$ $P_d = 0.9450$
	10^{-8}	$t_i = 2.2771$ $P_d = 0.5496$	$t_i = 2.3665$ $P_d = 0.7131$	$t_i = 2.4264$ $P_d = 0.8130$
GWO algorithm	10^{-4}	$t_i = 0.9588$ $P_d = 0.8993$	$t_i = 1.0190$ $P_d = 0.9730$	$t_i = 1.0594$ $P_d = 0.9923$
	10^{-6}	$t_i = 1.5530$ $P_d = 0.7457$	$t_i = 1.6265$ $P_d = 0.8839$	$t_i = 1.6758$ $P_d = 0.9450$
	10^{-8}	$t_i = 2.2771$ $P_d = 0.5496$	$t_i = 2.3665$ $P_d = 0.7131$	$t_i = 2.4264$ $P_d = 0.8130$
BBO algorithm	10^{-4}	$t_i = 0.9587$ $P_d = 0.8993$	$t_i = 1.0190$ $P_d = 0.9730$	$t_i = 1.0594$ $P_d = 0.9923$
	10^{-6}	$t_i = 1.5530$ $P_d = 0.7457$	$t_i = 1.6265$ $P_d = 0.8839$	$t_i = 1.6758$ $P_d = 0.9450$
	10^{-8}	$t_i = 2.2771$ $P_d = 0.5496$	$t_i = 2.3665$ $P_d = 0.7130$	$t_i = 2.4264$ $P_d = 0.8130$

6.5.2 Non-Identical Distributed GO-CFAR Detectors (OR Fusion Rule)

Table 6.2 presents the results for non-identical GO-CFAR detectors with $M_1 = 16$, $M_2 = 24$, and $M_3 = 32$. In this configuration, the optimization problem becomes multi-dimensional since each detector has an independent threshold multiplier.

Compared to the identical case, small but noticeable performance differences among optimization algorithms appear. MFO consistently achieves the highest detection probability across all false alarm levels, though the improvements remain moderate. For instance, at $\alpha_0 = 10^{-4}$, MFO achieves $P_d = 0.8954$, slightly outperforming GWO and BBO.

The results indicate that as the dimensionality of the search space increases (due to non-identical parameters), the explorationexploitation balance of the optimizer becomes more relevant. MFO demonstrates improved robustness in such multivariable configurations.

TABLE 6.2: Optimal scale factors of non-identical GO-CFAR detectors using MFO, GWO and BBO algorithms for $M_1 = 16$, $M_2 = 24$, $M_3 = 32$, $S = 20$ dB and OR fusion rule.

Optimiz-tool	α_0	α_1	α_2	α_3
MFO algorithm	10^{-4}	1.5678	0.9606	0.6929
		$P_d = 0.8954$		
	10^{-6}	2.6895	1.5635	1.0872
		$P_d = 0.7348$		
	10^{-8}	4.2176	2.2918	1.5409
		$P_d = 0.5332$		
GW algorithm	10^{-4}	1.5797	0.9661	0.6844
		$P_d = 0.8953$		
	10^{-6}	2.7077	1.5472	1.0915
		$P_d = 0.7348$		
	10^{-8}	4.2341	2.2798	1.5431
		$P_d = 0.5331$		
BBO algorithm	10^{-4}	1.5421	1.0394	0.6684
		$P_d = 0.8935$		
	10^{-6}	2.8239	1.4972	1.1017
		$P_d = 0.7328$		
	10^{-8}	4.1116	2.2574	1.6278
		$P_d = 0.5282$		

6.5.3 Non-Identical Distributed SO-CFAR Detectors (OR Fusion Rule)

Table 6.3 reports the optimized thresholds for the distributed SO-CFAR architecture under the same non-identical configuration.

In this case, performance differences among optimizers become more pronounced. MFO provides a clear improvement in detection probability over GWO and BBO for all false alarm levels. For $\alpha_0 = 10^{-4}$, MFO achieves $P_d = 0.9462$, compared to 0.8623 and 0.8305 for GWO and BBO, respectively.

The distributed SO-CFAR architecture exhibits superior detection performance compared to GO-CFAR under the same conditions. This behavior is consistent with the inherent robustness of SO-CFAR against multiple target interference and heterogeneous clutter transitions.

TABLE 6.3: Optimal thresholds of non-identical SO-CFAR detectors using MFO, GWO and BBO algorithms for $M_1 = 16$, $M_2 = 24$, $M_3 = 32$, $S = 20$ dB and OR fusion rule.

Optimiz-tool	α_0	α_1	α_2	α_3
MFO Algorithm	10^{-4}	1.8409	0.8748	0.7274
		$P_d = 0.9462$		
	10^{-6}	4.4052	2.3883	1.4785
		$P_d = 0.7058$		
	10^{-8}	8.6902	4.4542	2.6894
		$P_d = 0.3607$		
GWO algorithm	10^{-4}	2.9511	1.4911	0.9906
		$P_d = 0.8623$		
	10^{-6}	6.0710	2.6642	1.6490
		$P_d = 0.6228$		
	10^{-8}	12.9761	4.3028	2.4673
		$P_d = 0.3480$		
BBO Algorithm	10^{-4}	3.5400	1.3062	1.3300
		$P_d = 0.8305$		
	10^{-6}	5.4373	2.8432	1.8135
		$P_d = 0.6069$		
	10^{-8}	10.1671	5.2553	2.7756
		$P_d = 0.3039$		

6.6 Discussion and Practical Considerations

The reported results indicate that performance differences among metaheuristics methods are modest for low-dimensional cases (e.g., identical detectors), but become more visible as the optimization dimensionality increases (non-identical detectors). This is consistent with the intuition that a more complex coupled search space benefits from stronger exploration/exploitation balance.

Fusion-rule effect. The OR fusion rule tends to benefit more from optimization because it amplifies the contribution of sensors with higher local detection performance. Conversely, the AND rule is conservative and can suppress gains from optimizing individual detectors, explaining why optimizers often converge to similar outcomes under AND.

Operational limitation. The reported formulation assumes known (or fixed) local SNR values (as in the tables). In practice, SNR is uncertain and can be time-varying. A practical implementation approach is to perform offline optimization for a grid of SNR hypotheses and store the resulting $\{\alpha_i\}$ in a lookup table, or to use periodic re-optimization when environmental conditions vary slowly.

Modeling limitation. This chapter reports results under Gaussian disturbance. Extending the same optimization framework to compound-Gaussian or heavy-tailed sea clutter is a key direction for future work.

6.7 Conclusion and Future Works

This chapter addressed the optimization of distributed GO-CFAR and SO-CFAR detectors using bio-inspired metaheuristics, with a focus on the Moth-Flame Optimization (MFO) algorithm. The distributed threshold tuning problem was cast as a multivariable optimization task under a global false alarm constraint, and MFO was applied to compute the local threshold multipliers.

The proposed MFO method was compared against the Grey Wolf Optimization (GWO) and Biogeography-Based Optimization (BBO) algorithms. The reported results (Tables 6.1–6.3) indicate that MFO provides comparable or slightly improved detection performance, with a clearer advantage in non-identical configurations where the search space becomes higher-dimensional.

Future directions:

- **Extension to non-Gaussian clutter:** extend the distributed optimization framework to compound-Gaussian and heavy-tailed clutter models that better represent maritime and heterogeneous radar environments.
- **Deep learning-based surrogate optimization:** replace iterative metaheuristic search with a learning-based predictor capable of estimating near-optimal threshold multipliers $\{\alpha_i^\star\}$ directly from observed data. A neural network can be trained offline using synthetic scenarios (varying M_i, fusion rules, clutter statistics, and SNR hypotheses), where labels are generated by the optimization procedure. At runtime, the trained model provides low-latency inference of $\{\alpha_i\}$ with negligible computational cost compared to population-based optimization. Constraint-aware learning can be incorporated by adding a penalty term enforcing $P_f^{(global)} \approx \alpha_0$ during training or by projecting network outputs onto the feasible CFAR set.

7 General Conclusion and Perspectives

Contents

7.1 Overview and Context

The primary goal of this dissertation was to develop and evaluate advanced statistical and deep learning-based techniques for the modeling, Parameter estimation, and detection of maritime radar clutter, particularly under non-Gaussian and correlated conditions. Over the past several decades, the accurate estimation of clutter parameters and the design of robust CFAR detectors have remained central to radar research. Yet, the increased complexity of operational environments characterized by strong correlations, heterogeneous clutter, and rapidly changing sea states has necessitated a paradigm shift from purely statistical models toward hybrid data-driven approaches. This work contributes to that transition by integrating deep learning architectures, evolutionary algorithms, and model selection techniques into the radar detection pipeline.

The scientific contributions presented in this dissertation encompass four main pillars:

1. **Model Selection via Cross-Validation:** We introduced a data-centric model selection framework leveraging cross-validation to determine the most appropriate clutter distribution under varying environmental conditions. This approach aligns with practical radar deployment scenarios where the clutter statistics may deviate from theoretical expectations.

2. **Modeling of Sea Clutter Using SIRV Frameworks:** We adopted and extended the SIRV model to capture the statistical complexity of radar clutter distributions. This model accounts for both the speckle (fast-varying Gaussian component) and texture (slowly-varying modulator), enabling the representation of compound distributions such as the K-distribution and Pareto Type II.

3. **Data-Driven Clutter Parameter Estimation:** Through the design of CNN-LSTM and multi-headed deep learning architectures, we proposed novel estimators for shape and scale parameters of K-distributed and Pareto Type II clutter. These models were trained using synthetically generated datasets and evaluated against traditional estimators such as MLE, FOME, and [zlog(z)].

4. **CFAR Optimization:** Using MFO , we addressed the challenge of distributed CFAR parameter tuning, optimizing scale factors for GO-CFAR and SO-CFAR under OR , AND, and MAJORITY fusion rules. The results demonstrated that MFO provides superior detection probability compared to the GWO and BBO.

7.2 Key Findings and Contributions

7.2.1 Cross-Validation for Clutter Model Selection

Model selection, often overlooked in radar signal processing, was rigorously explored in Chapter 3. By using cross-validated CCDF fitting across various range cells, we demonstrated that no single clutter model is universally optimal. Instead, adaptive selection strategies are required. For instance, the K-distribution proved superior for high-resolution, speckle-dominated scenarios, while Pareto Type II was more appropriate for low-resolution or texture-rich conditions. These insights informed the design of both the parameter estimators and CFAR architectures.

7.2.2 Deep Learning in Clutter Parameter Estimation

Chapters 4 and 5 demonstrated that deep learning (DL) models, especially those that incorporate temporal modeling (LSTM, BLSTM) and spatial feature extraction (CNN), outperform conventional statistical estimators in both accuracy and robustness. By leveraging many-to-one and many-to-many sequence regression paradigms, our models were able to predict clutter parameters directly from raw amplitude sequences, bypassing the need for handcrafted features. This finding is particularly relevant in non-stationary or highly impulsive environments, where methods like IOM or MLE exhibit strong biases.

We introduced a novel architecture that combines LSTM-AE, CNN, BLSTM, and CNN-LSTM branches to harness complementary strengths of various neural blocks. The fusion of intermediate representations and predictions yielded a unified model capable of generalizing across multiple correlation levels, spikiness regimes, and SNR conditions. Notably, this model achieved the lowest MSE across all tested scenarios while maintaining low inference latency, making it suitable for real-time radar systems.

7.2.3 Metaheuristic-Based CFAR Threshold Optimization

In Chapter 6, we optimized CFAR scale factors for multiple sensor configurations using metaheuristic algorithms. The MFO algorithm consistently outperformed GWO and BBO in terms of convergence speed and detection accuracy, especially under non-identical sensor windows and heterogeneous clutter. These findings underscore the importance of intelligent optimization in next-generation radar fusion frameworks.

This research aligns closely with the objectives of cognitive radar systems capable of perceiving, learning, and adapting to their operational environment. By embedding deep learning models within the radar signal chain, we move toward detectors that dynamically adjust to clutter statistics, improving target detection rates without human intervention.

7.3 Future Research Directions

7.3.1 Threshold Estimation with DL Models

A promising extension of this work lies in the direct estimation of CFAR detection thresholds using DL models. Instead of estimating (λ, η) and computing α_i separately, future models could be trained end-to-end to map clutter sequences directly to threshold values corresponding to desired P_{FA} levels.

7.3.2 Transfer Learning and Online Adaptation

Given the variability of maritime environments, models trained offline may experience domain shift in the field. Transfer learning approachessuch as fine-tuning with limited labeled data or employing domain adaptation techniquescould allow for online adaptation without full retraining.

7.3.3 Hybrid Statistical-Deep Fusion Models

Rather than treating classical and DL models as alternatives, hybrid schemes could leverage the analytical insights of statistical models with the adaptability of deep networks. For instance, a moment-based estimate could serve as an initialization or regularization term in a DL loss function, enhancing generalization in low-data regimes.

7.3.4 Data Augmentation and Simulation Engines

The quality of DL models is inherently tied to the quality and diversity of training data. While this work used SIRV-based synthetic data, future developments could incorporate electromagnetic simulations or generative models (e.g., GANs) to create richer training environments.

7.4 Final Reflections

The work presented in this dissertation is both timely and foundational. As radar systems evolve from rule-based decision-making to adaptive, data-driven architectures, the techniques developed herein will play a central role in ensuring their robustness, reliability, and efficiency. By bridging theoretical modeling with modern machine learning, we have laid the groundwork for intelligent radar systems capable of operating in the most challenging cluttered environments.

Nonetheless, numerous open questions remain. How can these estimators be scaled to multi-target scenarios? What is the best strategy for updating model weights in resource-constrained environments? How do we ensure fairness and reproducibility across deployments?

As radar technology continues to advance, these questions will guide the next wave of innovations. The path forward is both challenging and excitingbut the tools developed in this dissertation provide a strong foundation for navigating it.

Bibliography

[1] M. I. Skolnik, *Introduction to radar systems*, vol. 3. New York: McGraw-Hill, 1980.

[2] S. Watts, "Asv mark i the first airborne maritime surveillance radar," *IEEE Aerospace and Electronic Systems Magazine*, vol. 34, no. 8, pp. 50–61, 2019.

[3] S. Watts and L. Rosenberg, "Challenges in radar sea clutter modelling," *IET Radar, Sonar & Navigation*, vol. 16, no. 9, pp. 1403–1414, 2022.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[5] G. Lampropoulos, A. Drosopoulos, N. Rey, *et al.*, "High resolution radar clutter statistics," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 1, pp. 43–60, 1999.

[6] K. Ward, C. Baker, and S. Watts, "Maritime surveillance radar. i. radar scattering from the ocean surface," in *IEE Proceedings F - Radar and Signal Processing*, vol. 137, pp. 51–62, IET, 1990.

[7] U. S. Hashmi, S. Akbar, R. Adve, P. W. Moo, and J. Ding, "Artificial intelligence meets radar resource management: A comprehensive background and literature review," *IET Radar, Sonar & Navigation*, vol. 17, no. 2, pp. 153–178, 2023.

[8] M. Rangaswamy, P. Chakravarthi, D. Weiner, L. Cai, and H. Wang, "Signal detection in correlated gaussian and non-gaussian radar clutter," tech. rep., 1993.

[9] S. Haykin, "Signal processing: where physics and mathematics meet," *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 6–7, 2001.

[10] K. J. Sangston and K. R. Gerlach, "Coherent detection of radar targets in a non-gaussian background," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 2, pp. 330–340, 1994.

[11] E. Conte and M. Longo, "Characterisation of radar clutter as a spherically invariant random process," in *IEE Proceedings F (Communications, Radar and Signal Processing)*, vol. 134, pp. 191–197, IET, 1987.

[12] B. Cantrell, "Radar detection in non-gaussian, correlated clutter," Tech. Rep. NRL Report 9015, Naval Research Laboratory, 1986.

[13] P. P. Gandhi and S. A. Kassam, "Analysis of cfar processors in nonhomogeneous background," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-24, no. 4, pp. 427–445, 1988.

[14] M. Greco, P. Stinco, F. Gini, and M. Rangaswamy, "Impact of sea clutter nonstationarity on disturbance covariance matrix estimation and cfar detector performance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 3, pp. 1502–1513, 2010.

[15] A. Farina, F. Gini, M. Greco, and L. Verrazzani, "High resolution sea clutter data: statistical analysis of recorded live data," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 144, no. 3, pp. 121–130, 1997.

[16] W. Melvin and J. Scheer, "Principles of modern radar: Advanced techniques. new york," 2013.

[17] E. Jakeman and P. Pusey, "A model for non-rayleigh sea echo," *IEEE Transactions on Antennas and Propagation*, vol. 24, no. 6, pp. 806–814, 1976.

[18] B. Yonel, E. Mason, and B. Yazici, "Deep learning for passive synthetic aperture radar," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 90–103, 2017.

[19] M. Greco, F. Bordoni, and F. Gini, "X-band sea-clutter nonstationarity: influence of long waves," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 2, pp. 269–283, 2004.

[20] F. Gini, G. B. Giannakis, M. Greco, and G. T. Zhou, "Time-averaged subspace methods for radar clutter texture retrieval," *IEEE Transactions on Signal Processing*, vol. 49, no. 9, pp. 1886–1898, 2001.

[21] F. Gini and M. Greco, "Texture modelling, estimation and validation using measured sea clutter data," *IEE Proceedings-Radar, Sonar and Navigation*, vol. 149, no. 3, pp. 115–124, 2002.

[22] M. Greco, P. Stinco, and F. Gini, "Identification and analysis of sea radar clutter spikes," *IET Radar, Sonar & Navigation*, vol. 4, no. 2, pp. 239–250, 2010.

[23] E. Jakeman and P. Pusey, "Significance of k distributions in scattering experiments," *Physical Review Letters*, vol. 40, no. 9, p. 546, 1978.

[24] D. Schleher, "Radar detection in weibull clutter," *IEEE Transactions on aerospace and electronic systems*, no. 6, pp. 736–743, 2007.

[25] S. Watts, "Radar detection prediction in k-distributed sea clutter and thermal noise," *IEEE Transactions on Aerospace and Electronic Systems*, no. 1, pp. 40–45, 2007.

[26] G. V. Weinberg, "An investigation of the pareto distribution as a model for high grazing angle clutter.," tech. rep., 2011.

[27] E. Ollila, D. E. Tyler, V. Koivunen, and H. V. Poor, "Compound-gaussian clutter modeling with an inverse gaussian texture distribution," *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 876–879, 2012.

[28] K. J. Sangston, F. Gini, M. V. Greco, and A. Farina, "Structures for radar detection in compound gaussian clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 2, pp. 445–458, 2002.

[29] L. Rosenberg and S. Watts, *Radar sea clutter: modelling and target detection*. IET, 2021.

[30] V. Kezys and S. Haykin, "International conference on radar meteorology," in *Proceedings of the 26th International Conference on Radar Meteorology*, pp. 354–356, American Meteorological Society.

[31] T. H. Kerbaa, A. Mezache, and H. Oudira, "Model selection of sea clutter using cross validation method," *Procedia Computer Science*, vol. 158, pp. 394–400, 2019.

[32] S. Bocquet, L. Rosenberg, and C. H. Gierull, "Parameter estimation for a compound radar clutter model with trimodal discrete texture," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 10, pp. 7062–7073, 2020.

[33] D. R. Iskander and A. M. Zoubir, "Estimation of the parameters of the k-distribution using higher order and fractional moments [radar clutter]," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 4, pp. 1453–1457, 1999.

[34] I. Chalabi and A. Mezache, "Estimators of compound gaussian clutter with log-normal texture," *Remote Sensing Letters*, vol. 10, no. 7, pp. 709–716, 2019.

[35] D. Blacknell and R. Tough, "Parameter estimation for the k-distribution based on [zlog(z)]," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 148, no. 6, pp. 309–312, 2001.

[36] M. Sahed, A. Mezache, and T. Laroussi, "A novel [z log (z)]-based closed form approach to parameter estimation of k-distributed clutter plus noise for radar detection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 1, pp. 492–505, 2015.

[37] M. Sahed and A. Mezache, "Closed-form fractional-moment-based estimators for k-distributed clutter-plus-noise parameters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 4, pp. 2094–2100, 2017.

[38] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1d convolutional neural networks and applications: A survey," *Mechanical systems and signal processing*, vol. 151, p. 107398, 2021.

[39] H. M. Finn and R. S. Johnson, "Adaptive detection mode with threshold control as a function of spatially sampled clutter level estimates," *Radio Corporation of America Review*, vol. 29, no. 3, pp. 414–465, 1968.

[40] G. V. Trunk, "Range resolution of targets using automatic detection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 14, no. 5, pp. 750–755, 1978.

[41] V. G. Hansen and J. H. Sawyers, "Detectability loss due to greatest-of-selection in a cell averaging cfar," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 16, no. 1, pp. 115–118, 1980.

[42] H. Rohling, "Radar cfar thresholding in clutter and multiple target situations," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 19, no. 4, pp. 608–621, 1983.

[43] M. Barkat and P. K. Varshney, "Decentralized cfar signal detection," *IEEE Transactions on Aerospace and Electronic Systems*, pp. 141–149, 1989.

[44] R. S. Blum and J. Qiao, "Threshold optimization for distributed order-statistic cfar signal detection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 1, pp. 368–377, 1996.

[45] F. Gini and F. Lombardini, "Decentralized cfar detection with binary integration in weibull clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 2, pp. 396–407, 1997.

[46] W. Liu, Y. Lu, and J. S. Fu, "Data fusion of multiradar system by using genetic algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 2, pp. 601–612, 2002.

[47] J. D. Moore and N. B. Lawrence, "Comparison of two cfar methods used with square law detection of swerling i targets," in *International Radar Conference*, pp. 403–409, 1980.

[48] M. Weiss, "Analysis of some modified cell-averaging cfar processors in multiple-target situations," *IEEE Transactions on Aerospace and Electronic Systems*, no. 1, pp. 102–114, 2007.

[49] F. Gini and M. Greco, "Covariance matrix estimation for cfar detection in correlated heavy-tailed clutter," *Signal Processing*, vol. 82, no. 12, pp. 1847–1859, 2002.

[50] K. J. Sangston, F. Gini, and M. S. Greco, "Coherent radar target detection in heavy-tailed compound-gaussian clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 1, pp. 64–77, 2012.

[51] G. V. Weinberg, S. D. Howard, and C. Tran, "Bayesian framework for detector development in pareto distributed clutter," *IET Radar, Sonar & Navigation*, vol. 13, no. 9, pp. 1548–1555, 2019.

[52] M. Greco, F. Gini, and M. Diani, "Robust cfar detection of random signals in compound-gaussian clutter plus thermal noise," *IEE Proceedings-Radar, Sonar and Navigation*, vol. 148, no. 4, pp. 227–232, 2001.

[53] F. Gini and M. Greco, "Suboptimum approach to adaptive coherent radar detection in compound-gaussian clutter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 3, pp. 1095–1104, 1999.

[54] F. Gini, F. Lombardini, and P. K. Varshney, "On distributed signal detection with multiple local free parameters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 4, pp. 1457–1466, 1999.

[55] A. Gouri, A. Mezache, and H. Oudira, "Distributed ca-cfar and os-cfar detectors mentored by biogeography based optimization tool," *International Journal of Information Science and Technology*, vol. 3, no. 3, pp. 20–29, 2019.

[56] H. Oudira, A. Gouri, and A. Mezache, "Optimization of distributed cfar detection using grey wolf algorithm," *Procedia Computer Science*, vol. 158, pp. 74–83, 2019.

[57] T. H. Kerbaa, A. Mezache, and H. Oudira, "Improved decentralized so-cfar and go-cfar detectors via moth flame algorithm," in *2022 International Conference of Advanced Technology in Electronic and Electrical Engineering (ICATEEE)*, pp. 1–5, IEEE, 2022.

[58] P. Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proceedings of ICML workshop on unsupervised and transfer learning*, pp. 37–49, JMLR Workshop and Conference Proceedings, 2012.

[59] S. Haykin, W. Stehwien, C. Deng, P. Weber, and R. Mann, "Classification of radar clutter in an air traffic control environment," *Proceedings of the IEEE*, vol. 79, no. 6, pp. 742–772, 1991.

[60] A. Coluccia and G. Ricci, "Radar detection in k-distributed clutter plus thermal noise based on knn methods," in *2019 IEEE Radar Conference (RadarConf)*, pp. 1–5, IEEE, 2019.

[61] D. Roy, S. Srivastava, A. Kusupati, P. Jain, M. Varma, and A. Arora, "One size does not fit all: Multi-scale, cascaded rnns for radar classification," *ACM Transactions on Sensor Networks*, vol. 17, no. 2, pp. 1–27, 2021.

[62] J. Zhao, R. Jiang, X. Wang, and H. Gao, "Robust cfar detection for multiple targets in k-distributed sea clutter based on machine learning," *Symmetry*, vol. 11, no. 12, p. 1482, 2019.

[63] G. Wang, H. Ding, C. Wang, and N. Liu, "Estimation of sea clutter distribution parameters using deep neural network," in *Artificial Intelligence in China*, pp. 326–333, Singapore: Springer, 2020.

[64] M. Zhang, B. Li, H. Liu, and C. Zhao, "Federated learning for radar gesture recognition based on spike timing-dependent plasticity," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 2, pp. 2379–2393, 2024.

[65] H. Akaike, "Statistical predictor identification," *Annals of the institute of Statistical Mathematics*, vol. 22, no. 1, pp. 203–217, 1970.

[66] H. Akaike, "A new look at the statistical model identification," *IEEE transactions on automatic control*, vol. 19, no. 6, pp. 716–723, 2003.

[67] H. AKAIKE, "Comments on on model structure testing in system identification," *International Journal of Control*, vol. 27, no. 2, pp. 323–324, 1978.

[68] A. Hirotugu, "An objective use of bayesian models," *Annals of the Institute of Statistical Mathematics*, vol. 29, no. 1, pp. p9–20, 1977.

[69] C. M. Hurvich and C.-L. Tsai, "Regression and time series model selection in small samples," *Biometrika*, vol. 76, no. 2, pp. 297–307, 1989.

[70] R. Rao and Y. Wu, "A strongly consistent procedure for model selection in a regression problem," *Biometrika*, vol. 76, no. 2, pp. 369–374, 1989.

[71] P. M. Djuric, "Using the bootstrap to select models," in *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 3729–3732, IEEE, 1997.

[72] S. Watts, "The performance of cell-averaging cfar systems in sea clutter," in *Record of the IEEE 2000 International Radar Conference [Cat. No. 00CH37037]*, pp. 398–403, IEEE, 2000.

[73] S. Bocquet, "Simulation of correlated pareto distributed sea clutter," in *2013 International Conference on Radar*, pp. 258–261, IEEE, 2013.

[74] A. Mezache, A. Bentoumi, and M. Sahed, "Parameter estimation for compound-gaussian clutter with inverse-gaussian texture," *IET Radar, Sonar & Navigation*, vol. 11, no. 4, pp. 586–596, 2017.

[75] S. Bocquet, "Parameter estimation for pareto and k distributed clutter with noise," *IET Radar, Sonar & Navigation*, vol. 9, no. 1, pp. 104–113, 2015.

[76] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[77] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[78] A. Farina, A. Russo, F. Scannapieco, and S. Barbarossa, "Theory of radar detection in coherent weibull clutter," in *IEE Proceedings F - Communications, Radar and Signal Processing*, vol. 134, pp. 174–190, IET, 1987.

[79] R. Tough and K. Ward, "The correlation properties of gamma and other non-gaussian processes generated by memoryless nonlinear transformation," *Journal of Physics D: Applied Physics*, vol. 32, no. 23, p. 3075, 1999.

[80] E. Conte, M. Di Bisceglie, M. Longo, and M. Lops, "Canonical detection in spherically invariant noise," *IEEE Transactions on Communications*, vol. 43, no. 2/3/4, pp. 347–353, 1995.

[81] T. H. Kerbaa, A. Mezache, F. Gini, and M. S. Greco, "Cnn-lstm based approach for parameter estimation of k-clutter plus noise," in *2020 IEEE Radar Conference (RadarConf20)*, pp. 1–6, IEEE, 2020.

[82] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowledge-based systems*, vol. 89, pp. 228–249, 2015.