



2020 SSPD

Online



2020 SENSOR SIGNAL PROCESSING FOR DEFENCE CONFERENCE (SSPD)

Virtual Conference
15th and 16th September 2020

Welcome

Programme

Keynote Speakers

Invited Speakers

Technical Committee

SSPD2021 Flyer

IEEE Catalog Number: CFP20SPD-ART

ISBN: 978-1-7281-3810-7

IEEE Catalog Number: CFP20SPD-USB

ISBN: 978-1-7281-3809-1

2020 Sensor Signal Processing for Defence Conference (SSPD)

Copyright © 2020 by the Institute of Electrical and Electronics Engineers, Inc. All rights reserved.

Copyright and Reprint Permissions

Abstracting is permitted with credit to the source. Libraries are permitted to photocopy beyond the limit of U.S. copyright law for private use of patrons those articles in this volume that carry a code at the bottom of the first page, provided the per-copy fee indicated in the code is paid through Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923.

For other copying, reprint or republication permission, write to IEEE Copyrights Manager, IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08854. All rights reserved.

IEEE Catalog Number:	CFP20SPD-ART (Xplore)
ISBN:	978-1-7281-3810-7(Xplore)
IEEE Catalog Number:	CFP20SPD-USB (USB)
ISBN:	978-1-7281-3809-1 (USB)

Printed copies of this publication are available from:

Curran Associates, Inc
57 Morehouse Lane
Red Hook, NY 12571 USA
Phone: (845) 758-0400
Fax: (845) 758-2633
E-mail: curran@proceedings.com

Produced by IEEE eXpress Conference Publishing

For information on producing a conference proceedings and receiving an estimate, contact conferencepublishing@ieee.org
<http://www.ieee.org/conferencepublishing>

SSPD Conference 2020 - Welcome

Dear Colleagues,

We warmly welcome you to this year's SSPD Conference, our first virtual conference. This event is the 9th conference of the Sensor Signal Processing for Defence series and provides a chance to present, listen to and discuss the latest scientific findings in signal processing for defence.

We are privileged to have our two keynote speakers, Professor Vivek Goyal from Boston University and Dr. Daniel Sternlicht from the U.S Naval Surface Warfare Center, Panama City. The SSPD 2020 conference also welcomes our invited speakers; Professor Paul White from the University of Southampton; Professor Athina Petropulu from Rutgers University; Professor Sean Gong from Queen Mary University of London; and Dr. Paul Thomas from Dstl.

A welcome also extends to our military and industrial speakers and the presenters of scientific papers presenting their novel research through live oral presentations and pre-recorded sessions. We look forward to some interesting debate and discussion throughout the conference.

We would like to take this opportunity to thank the speakers, reviewers, session chairs and the technical committee for their contribution to this event.

We hope you enjoy our conference.

Mike Davies

Chair, SSPD 2020

Technical sponsorship is provided by IEEE Signal Processing Society and the IEEE Aerospace and Electronic Systems Society. Proceedings will be submitted to the Xplore Digital Library. The conference is organised by the University Defence Research Collaboration (UDRC) in Signal Processing, sponsored by the Defence Science and Technology Laboratory (Dstl) and the Engineering and Physical Sciences Research Council (EPSRC).



Technical Co-Sponsor

IEEE
Signal Processing Society



EPSRC
Pioneering research
and skills



Programme - SSPD2020

Tuesday 15th September 2020

Session 1 - Tracking, Detection and Localisation - Chair Mike Davies, University of Edinburgh

9:30 Introduction and Welcome to day 1 – Mike Davies

9:40 – 10:10 Invited Speaker: Soup in the Souk: A 'Bazaar' Approach to Defence Signal Processing, Paul Thomas, Dstl

10:10 – 10:35 Subspace Perturbation Bounds with an Application to Angle of Arrival Estimation using the MUSIC Algorithm, Connor Delaosa¹, Jennifer Pestana¹, Stephan Weiss¹, Ian K. Proudler¹,
¹University of Strathclyde

10:35 – 11:00 A Gaussian Process based Method for Multiple Model Tracking, Mengwei Sun¹, Mike E. Davies¹, Ian Proudler², James R. Hopgood¹,¹University of Edinburgh, ²University of Strathclyde

11:00 – 11:25 Narrowband Angle of Arrival Estimation Exploiting Graph Topology and Graph Signals, Ian K. Proudler¹, Vladimir Stankovic¹, and Stephan Weiss¹,¹University of Strathclyde

Close 11:25

Session 2 – Radar and Defence Panel- Chair - Stephen Ablett, Dstl

13:00 Introduction and Welcome to Session 2 – Chair Stephen Ablett

13:00 – 14:00 Defence Panel Discussion: Machine learning methods for defence and security face fundamental challenges which will inhibit their uptake, Chair Jordi Barr, Dstl

14:00 – 14:30 Invited Speaker, Dual Function Radar-Communication Systems, Athina Petropulu, Rutgers University

14:30 – 14:55 Information-Theoretic Compressive Measurement Design for Micro-Doppler Signatures, Fraser K. Coutts¹, John Thompson¹, Bernard Mulgrew¹,¹University of Edinburgh

14:55 – 15:20 Identification of Radar Emitter Type with Recurrent Neural Networks, Sabine Apfeld¹, Alexander Charlish¹, Gerd Ascheid² ¹Fraunhofer, ²RWTH Aachen University

15:20 Close

Session 3 Imaging - Chair - Stephen McLaughlin, Heriot-Watt University

16:00 Introduction and Welcome to Session 3 – Stephen McLaughlin

16:00 – 17:00 Academic Keynote Speaker, One Click At A Time: Photon- And Electron-Level Modeling For Improved Imaging, Vivek Goyal, University of Boston

17:00 – 17:25 Fast Surface Detection Using Single-Photon Detection Events, Abderrahim Halimi¹, Andrew Wallace¹, Gerald S. Buller¹, Stephen McLaughlin¹, Heriot-Watt University

17:25 – 17:50 Robust depth imaging in adverse scenarios using single-photon Lidar and beta-divergences, Quentin Legros¹, Stephen McLaughlin¹, Yoann Altmann¹, Sylvain Meignen², Mike E. Davies³, ¹Heriot-Watt University, ²University Grenoble Alpes, ³University of Edinburgh

17:50 Close

Wednesday 16th September 2020

Session 4 Machine Learning – Chair – Neil Robertson, Queen’s University Belfast

9:30 Introduction and Welcome to day 2 – Neil Robertson

9:40 – 10:00 Invited Speaker: Search & Learn: From User Guided Search to Federated Zero-Shot Learning, Sean Gong, Queen Mary University of London

10:10 – 10:35 Optimising Network Architectures for Provable Adversarial Robustness, Henry Gouk¹, Timothy M. Hospedales¹, ¹University of Edinburgh

10:35 – 11:00 Tail of Distribution GAN (TailGAN): Generative-Adversarial-Network-Based Boundary Formation, Nikolaos Dionelis¹, Mehrdad Yaghoobi¹, Sotirios A. Tsaftaris¹, ¹University of Edinburgh

11:00 -11:25 Electrical device classification using deep learning, Richard O Lane¹, Steven P Lindsay¹, ¹QinetiQ

Close 11:25

Session 5 – Mix of Signal Processing – James Hopgood, University of Edinburgh

13:00 Introduction and Welcome to Session 5 – James Hopgood

13:00 – 14:00 Spotlight presentations

14:00 – 14:30 Spotlight Questions and Answers.

Multimodal Learning for Early Detection of Explosive Sounds using Relative Spectral Distribution, Vishwajeet Shukla¹, Mayank Singour¹, ¹Samsung Research Institute

Detection and Identification of Radar Waveforms in Electronic Warfare context, Antoine Foucault¹, Cedric Cornu¹, Ali Khenchaf², Fabrice Comblet², ¹Thales Defense Mission Systems, ²Lab STICC, ENSTA Bretagne, CNRS.

Robust Source Number Estimation Based on Denoising Preprocessing, Koichi Ichige¹, Shohei Hamada¹, ¹Yokohama National University, Katsuhisa Kashiwagi², Nobuya Arakawa², Ryo Saito², ²Murata Manufacturing Co., Ltd.

Learning Entropy of Adaptive Filters via Clustering Techniques, Ivo Bukovsky¹, Gejza Dohnal¹, Pavel Steinbauer¹, Ondrej Budik¹, Kei Ichiji², Homma Noriyasu², ¹Czech Technical University in Prague ²Tohoku University.

Classifying LPI signals with transfer learning on CNN architectures, Bunlong Lay¹, Alexander Charlish¹, ¹Fraunhofer FKIE.

14:30 – 14:55 Approximate LASSO Model Predictive Control for Resource Constrained Systems, Yun Wu¹, Joao F. C. Mota¹, Andrew M. Wallace¹, ¹Heriot-Watt University

14:55 – 15:20 Extraction of Analytic Eigenvectors From a Parahermitian Matrix, Stephan Weiss¹, Ian K. Proudler¹, Fraser K. Coutts², and Julian Deeks³, ¹University of Strathclyde, ²University of Edinburgh, ³Dstl

15:20 Close

Session 6 Underwater Signal Processing - Chair Gary Heald, Dstl

16:00 Introduction and Welcome to Session 6 – Gary Heald

16:00 – 17:00 Defence Keynote Speaker, Sensing and Automation in the Future Maritime Environment, Daniel D. Sternlicht, U.S. Naval Surface Warfare Center, Panama City Division

17:00 – 17:30 Invited Speaker, A Whistle Stop Tour of Processing for Delphinid Vocalisations, Paul White, University of Southampton

17:30 – 17:40 Closing remarks

Keynote Speakers

Dr. Vivek Goyal

Vivek Goyal received the M.S. and Ph.D. degrees in electrical engineering from the University of California, Berkeley, where he received the Eliahu Jury Award for outstanding achievement in systems, communications, control, or signal processing. He was a Member of Technical Staff at Bell Laboratories, a Senior Research Engineer for Digital Fountain, and the Esther and Harold E. Edgerton Associate Professor of Electrical Engineering at MIT. He was an adviser to 3dim Tech, winner of the 2013 MIT \$100K Entrepreneurship Competition Launch Contest Grand Prize, and consequently with Nest Labs 2014-2016. He is now an Associate Professor of Electrical and Computer Engineering at Boston University.



Dr. Goyal is a Fellow of the IEEE and of the OSA. He was awarded the IEEE Signal Processing Society (SPS) Magazine Award in 2002, the IEEE Int. Conf. on Image Processing Best Paper Award in 2014, the IEEE SPS Best Paper Award in 2017 and 2019, and an NSF CAREER Award. Work he supervised won awards at the IEEE Data Compression Conf. in 2006 and 2011, the IEEE Sensor Array and Multichannel Signal Processing Workshop in 2012, the IEEE Int. Conf. on Imaging Processing in 2018, and the IEEE Int. Conf. Computational Photography in 2018, as well as five MIT thesis awards. He is a co-author of Foundations of Signal Processing (Cambridge University Press, 2014).

ABSTRACT: ONE CLICK AT A TIME: PHOTON- AND ELECTRON-LEVEL MODELING FOR IMPROVED IMAGING

Using detectors with single-photon sensitivity enables lidar systems to form depth and reflectivity images at very long ranges, which has improved surveillance and reconnaissance capabilities. Initially, our interest was in exploiting inhomogeneous Poisson processes models and the typical structure of natural scenes to achieve extremely high photon efficiency through first-photon imaging and related methods. However, modeling at the level of individual photons does not merely give advantages when signals are weak. It is also central to withstanding high levels of ambient light and mitigating the effects of detector dead time, which ordinarily create high bias in high-flux imaging. Our sensor signal processing advances thus potentially improve lidar performance in settings with very high dynamic range of optical flux, such as autonomous navigation. Furthermore, modeling at the level of individual incident particles and emitted secondary electrons leads to improvements in focused ion beam microscopy that apply uniformly over all dose levels.

Key related paper identifiers: [10.1126/science.1246775](https://doi.org/10.1126/science.1246775) [10.1109/TSP.2015.2453093](https://doi.org/10.1109/TSP.2015.2453093)
[10.1038/ncomms12046](https://doi.org/10.1038/ncomms12046) [10.1109/TSP.2017.2706028](https://doi.org/10.1109/TSP.2017.2706028) [10.1126/science.aat2298](https://doi.org/10.1126/science.aat2298)
[10.1109/TSP.2019.2914891](https://doi.org/10.1109/TSP.2019.2914891) [10.1016/j.ultramic.2020.112948](https://doi.org/10.1016/j.ultramic.2020.112948)

Dr. Daniel D. Sternlicht

Dr. Daniel D. Sternlicht is the Distinguished Scientist for Littoral Sensing Technologies at the U.S. Naval Surface Warfare Center Panama City Division (NSWC PCD), where he serves as technical expert in sensing technologies relevant to the full spectrum of littoral warfare systems; provides subject matter expertise across the Naval Research and Development Establishment; and leads new developments in maritime reconnaissance and surveillance for Navy and Marine Corps missions. During a career that includes scientific research and program and line management, Dr. Sternlicht’s pioneering work in through-the-sensor environmental characterization, multi-sensor fusion, automated change detection, and advanced techniques for localization and classification of underwater munitions, combined with his leadership in developing and transitioning state-of-the art sensing systems into the Fleet and scholarship in the historical development of U.S. Navy sensors, has led to international recognition as an authority in maritime reconnaissance and surveillance technologies.



Dr. Sternlicht received the B.A. degree in Biology from the University of Pennsylvania, Philadelphia, the M.S. degree in Electrical Engineering from the University of Hawaii, Manoa, and the Ph.D. degree in Electrical Engineering and Applied Ocean Science from the University of California, San Diego and Scripps Institution of Oceanography. He lectures regularly at civilian and military colleges and universities, has chaired numerous technical conference sessions, was guest editor for the IEEE Journal of Oceanic Engineering 2009 Special Issue on Synthetic Aperture Sonar, and is Executive Co-Chair for the upcoming 2022 MTS/IEEE OCEANS technology conference. In 2013 Dr. Sternlicht received the Department of the Navy Meritorious Civilian Service Award, and in 2018 he was promoted to the position of Senior Scientist & Technology Manager (SSTM).

Abstract: SENSING AND AUTOMATION IN THE FUTURE MARITIME ENVIRONMENT

In this emerging era of great power competition, the goal of outpacing potential adversaries in the development of military technology takes on a new urgency. Evolving capabilities in sensing and automation are driven by a trade space that includes range and lethality versus close engagement and survivability; finders versus hidiers; connection/aggregation/centralization versus disconnection/disaggregation/decentralization; and planning and judgement versus reaction and autonomy. Cooperative networks of offboard systems will be essential to future maritime operations – where the balance between maintaining control with full communications and accepting the risk of acting without (or with limited) communications will continue to evolve with technology. This paper discusses developments in advanced sensors and automation that will be key to realizing a networked force of manned and offboard systems with the ability to sense, comprehend, communicate, predict, plan and take appropriate action in the future maritime environment.

Invited Speakers

Professor Paul White

Paul White is Professor of Statistical Signal Processing within Engineering and Physical Sciences at the University of Southampton.

"Sound plays a vital role in the lives of marine mammals; understanding how they use acoustics provides inspiration for man-made systems, whilst monitoring the impact of man-made noise is important for their conservation."

Paul is Professor of Statistical Signal Processing in the Institute of Sound and Vibration Research (ISVR).

Having obtained a BSc in Mathematics in 1985, Paul moved along the south coast to Southampton to undertake his PhD at the University of Southampton, becoming a lecturer in ISVR in 1988 and was awarded his Chair in 2004.



ABSTRACT: A WHISTLE STOP TOUR OF PROCESSING FOR DELPHINID VOCALISATIONS

Passive Acoustic Monitoring (PAM) of the ocean has increased dramatically over the last decade. PAM being the civilian equivalent to passive sonar. The availability of relatively low cost, reliable, long-endurance underwater acoustic recorders has meant that an increasing amount of acoustic data is being collected. One of the key motivators for this technology is to study marine mammals, be that to: monitor for them to activities (e.g. sonar transmissions) to mitigate impacts, explore animal distributions or to estimate species abundance. Having collected the large datasets there is a need to develop tools to assist with the analysis – traditional analysis methods rely upon manual inspection of data in near-real time, which becomes unfeasibly demanding when deployments can last months.

Delphinids (oceanic dolphins) are a species group which produce three general classes of vocalisations: echolocation clicks, burst pulses and whistles. Classifying delphinids to the species level, based on acoustic data remains a significant challenge, but it is believed that whistles (frequency modulated chirps) provide the most promising route to acoustic classification. Real datasets recorded from these animals consists of a complicated mixture of multiple rapidly frequency modulated narrowband signals mixed with broadband and impulsive noise sources. This paper will discuss some of the processing challenges associated with analysing these whistles with the ultimate goal of classifying them. In particular, we shall consider the problems of detecting the whistles, extracting them and then solutions to the multi-target tracking problem that arises when trying to isolate whistles from individuals.

Professor Athina P. Petropulu

Athina P. Petropulu is a distinguished Professor at the Electrical and Computer Engineering (ECE) Department at Rutgers, having served as chair of the department during 2010-2016. Prior to joining Rutgers she was a Professor of ECE at Drexel University (1992-2010). She held Visiting Scholar appointments at SUPELEC, Universite' Paris Sud, Princeton University and University of Southern California. Dr. Petropulu's research interests span the area of statistical signal processing, wireless communications, signal processing in networking, physical layer security, and radar signal processing.



Her research has been funded by various government industry sponsors including the National Science Foundation (NSF), the Office of Naval research, the US Army, the National Institute of Health, the Whitaker Foundation, Lockheed Martin and Raytheon. Dr. Petropulu is Fellow of IEEE and AAAS and recipient of the 1995 Presidential Faculty Fellow Award given by NSF and the White House. She is President-Elect for the IEEE Signal Processing Society for 2020-2021. She has served as Editor-in-Chief of the IEEE Transactions on Signal Processing (2009-2011), IEEE Signal Processing Society Vice President-Conferences (2006-2008), and is currently member-at-large of the IEEE Signal Processing Board of Governors. She was the General Chair of the 2005 International Conference on Acoustics Speech and Signal Processing (ICASSP-05), Philadelphia PA, and is General Co-Chair of the 2018 IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC). She is recipient of the 2005 IEEE Signal Processing Magazine Best Paper Award, and the 2012 IEEE Signal Processing Society Meritorious Service Award. She was Distinguished Lecturer for the Signal Processing Society for 2017-2018, and is currently Distinguished Lecturer for the IEEE Aerospace & Electronics Systems Society.

ABSTRACT: DUAL FUNCTION RADAR-COMMUNICATION SYSTEMS

Automotive radars for advanced driver assistance systems and autonomous driving are required to have high angle discrimination capability and small package size so that they can be easily integrated into vehicles. Unlike conventional phase arrays whose resolution is proportional to their size, multi-input multi-output (MIMO) radar can meet both high resolution and small size requirements. This is because MIMO radar can synthesize virtual arrays with large apertures using only small number of transmit and receive antennas. Even with the help of MIMO radar technology, however, the cost of synthesizing a large virtual uniform linear array (ULA) with half wavelength element spacing can be very high. One way to further reduce the cost without sacrificing angle resolution is to use virtual sparse linear arrays (SLAs), e.g., use a thinned receive ULA. SLA operating as a MIMO radar can properly deploy the reduced number of transmit and receive antennas, such that the element spacing of the corresponding virtual array is larger than half wavelength, while its aperture is the same as that of a ULA with half wavelength element spacing. Prior approaches have focused on optimal sparse array design, or use of interpolation techniques for filling the holes in the synthesized SLA before applying digital beamforming for angle finding. In this talk, we present a new approach, where we use matrix completion to complete the corresponding virtual ULA before estimating the target angle. In particular, we show that for a small number of targets within the same range-Doppler cell, the Hankel matrix constructed by subarrays of the virtual ULA is low-rank, and thus under certain conditions, can be completed based on the SLA measurements. We derive the coherence properties of the Hankel matrix so that it can be completed via nuclear norm minimization methods. We also demonstrate via examples the effect of various SLA topologies on the identifiability of the Hankel matrix.

Professor Sean Gong

Shaogang (Sean) Gong is Professor of Visual Computation at Queen Mary University of London and a Turing Fellow of the Alan Turing Institute of Data Science and Artificial Intelligence. He established the Queen Mary Computer Vision Laboratory in 1993 and has enjoyed immensely working with PhD students and postdoctoral researchers. His research is in Computer Vision and Machine Learning (Google Scholar and DBLP), with a focus on Object Recognition, Action Recognition, and Video Analysis. His brief bio and publications with pdf download.



ABSTRACT: SEARCH AND LEARN: FROM USER GUIDED SEARCH TO FEDERATED ZERO-SHOT LEARNING

Deep learning has been successful for many computer vision tasks due to the availability of shared and centralised large sized training data. However, increasing awareness of privacy concerns poses new challenges to deep learning, especially for human subject related recognition such as person reidentification (Re-ID). Moreover, existing person search methods predominantly assume the availability of at least one-shot imagery sample of the queried person. This assumption is limited in circumstances where only a brief textual (or verbal) description of the target person is available.

In this talk, I will describe challenges and recent progress on deep learning for text attribute based person search without any query image, and decentralised learning from non-shared private training data distributed at multiple user-cites of independent multidomain labels for person re-identification. Both problems require solving generalised Zero-Shot Learning.

Dr. Paul Thomas

Paul is a Fellow in the UK MOD's Defence Science and Technology Laboratory (Dstl) and Visiting Professor of Sensor Fusion and Autonomy at the University of Loughborough.

With 20+ years' experience in the area of sensor fusion, signal processing and autonomy for defence applications, Paul now holds the role of Principal Advisor for ISR Fusion and Processing and Lead Technical Reviewer for Underpinning Data Science. Paul serves on the Technical Programme Committees for the ISIF Fusion and IET Intelligent Signal Processing (ISP) conferences.



ABSTRACT: SOUP IN THE SOUK: A 'BAZAAR' APPROACH TO DEFENCE SIGNAL PROCESSING

You are currently experiencing an earthquake in software and algorithm development. Open Source threatens to shake down the cathedrals of corporate software projects and flatten the chapels of private code repositories. Replacing them with the ad-hoc, organic, collaborative energy of the bazaar. This is, in equal measure, an opportunity for the signal processing algorithm developer and the defence industry as a whole.

This talk describes the development of Stone Soup, the open source framework for tracking and state estimation. We describe the development process of an open source project and experiences gained while creating critical mass. We discuss how structure is key to enabling engagement and how design is important even in an amorphous project. Stone soup is already transforming the culture of the academic community and we discuss how it can bring changes to the way defence industry exploits signal processing innovations.

SSPD2020 Conference Committee

General Chairs

- Mike Davies - University of Edinburgh
- Stephen McLaughlin - Heriot-Watt University
- Jordi Barr - Dstl
- Gary Heald - Dstl

Publicity and Local Arrangements Chair

- Janet Forbes - University of Edinburgh

Technical Programme Committee

- Abderrahim Halimi - Heriot-Watt University
- Alasdair Hunter - Dstl
- Alessio Balleri - Cranfield University
- Andreas Ahrens - Hochschule Wismar
- Andrew Baird - Dstl
- Andrew Wallace - Heriot-Watt University
- Andy Stove - Stove Specialties
- Antonio de Maio - University of Naples "Federico II"
- Athanasios Gkelias - Imperial College London
- Augusto Aubry - Universita degli studi di Napoli
- Bernard Mulgrew - University of Edinburgh
- Brian Barber - Dstl
- Bruno Clercx - Imperial College London
- Carmine Clemente - University of Strathclyde
- Chris Baker - University of Birmingham
- Christoph Wasserzier - Fraunhofer
- Christos Ilioudis - University of Strathclyde
- Cristian Rusu - Dublin University
- David - Nethercott Dstl
- David Garren - Naval Postgraduate School
- Duncan Williams - Dstl
- Fan Liu - University College London
- Francis Watson - Thales
- Francisco Javier Aparicio Navarro - De Montfort University
- George Jacob - Dstl
- Harvey Alison - Leonardo
- Hugh Griffiths – University College London
- Ian Proudler - Loughborough University
- Ioannis Kaloskampis - Cardiff University
- Ivo Bukovsky - Czech Technical University in Prague
- James Hopgood - University of Edinburgh

- Joao Mota - Heriot-Watt University
- John Buck - University of Massachusetts Dartmouth
- John Thompson - University of Edinburgh
- Jose Vazquez - Seebyte
- Julian Deeks - Dstl
- Keith Brown - Heriot-Watt University
- Keith Thompson -University of Strathclyde
- Kin Leung - Imperial College
- Krishnaprasad Nambur Ramamohan - Delft University of Technology
- Laura Anitori - TNO
- Mahesh Banavar - Clarkson University
- Maria Greco - University of Pisa
- Mathini Sellathurai -Heriot-Watt University
- Mehrdad Yaghoobi - University of Edinburgh
- Murat Uney - NATO CMRE
- Neil Cade - Leonardo
- Oliver Sims - Leonardo
- Rodrigo de Lamare - Pontifical Catholic University of Rio de Janeiro
- Sami Aldalahmeh - Al-Zaytoonah University of Jordan
- Simon Maskell - University of Liverpool
- Stephen Ablett - Dstl
- Vladimir Stankovic - University of Strathclyde
- Wei Dai - Imperial College London
- Wenwu Wang - University of Surrey
- Wolfgang Koch - Fraunhofer
- Yoann Altmann - Heriot-Watt University

SSPD 2021

Sensor Signal Processing for Defence Conference

Important Dates:

Submission of Papers: 18 April 2021

Notification of Paper Acceptance: 1 July 2021

Final version of Paper Due: 30 July 2021

Date of conference: 14 to 15 September 2021

Location: University of Edinburgh



International Conference in Sensor Signal Processing for Defence: from Sensor to Decision

The Sensor Signal Processing for Defence Conference is organised by the University Defence Research Collaboration (UDRC) in Signal Processing. SSPD 2021 aims to bring together researchers from academia, industry and government organisations interested in Signal Processing for Defence.

Papers are solicited from the following areas:-

- Array Signal Processing
- Image Processing
- Radar, Sonar and Acoustic
- Multimodal Signal Processing
- Multi-Target Tracking
- Signal Acquisition and Sensor Management
- Multiple-input and multiple-output (MIMO)
- Deep Learning, Machine Learning
- Information/Data Analysis
- Data Fusion
- Source Separation
- Anomaly Detection
- Distributed Signal Processing
- Low Size Weight & Power Solutions
- Target Detection and Identification
- Electro-Optic Sensing

All submitted papers will be peer reviewed. Technical sponsorship is provided by the IEEE Signal Processing Society and the IEEE Aerospace and Electronic Systems Society and proceedings will be submitted to the Xplore Digital Library.

www.sspdconference.org



SSPD
Conference



Technical Co-Sponsor



Table of Contents

Session 1: Tracking, Detection and Localisation

Subspace Perturbation Bounds with an Application to Angle of Arrival Estimation using the MUSIC Algorithm..... 1
Connor Delaosa, Jennifer Pestana, Stephan Weiss, and Ian K. Proudler

A Gaussian Process Based Method for Multiple Model Tracking..... 6
Mengwei Sun, Mike E. Davies, Ian Proudler, and James R. Hopgood

Narrowband Angle of Arrival Estimation Exploiting Graph Topology and Graph Signals 11
Ian K. Proudler, Vladimir Stankovic, and Stephan Weiss

Session 2: Radar

Information-Theoretic Compressive Measurement Design for Micro-Doppler Signatures 16
Fraser K. Coutts, John Thompson, and Bernard Mulgrew

Identification of Radar Emitter Type with Recurrent Neural Networks..... 21
Sabine Apfeld, Alexander Charlish, and Gerd Ascheid

Session 3: Imaging

Fast Surface Detection using Single-Photon Detection Events 26
Abderrahim Halimi, Andrew Wallace, Gerald S. Buller, and Stephen McLaughlin

Robust Depth Imaging in Adverse Scenarios using Single-Photon Lidar and Beta-Divergences 31
Q. Legros, S. McLaughlin, Y. Altmann, S. Meignen, and Mike E. Davies

Session 4: Machine Learning

Optimising Network Architectures for Provable Adversarial Robustness 36
Henry Gouk and Timothy M. Hospedales

Tail of Distribution GAN (TailGAN): Generative-Adversarial-Network-Based Boundary Formation..... 41
Nikolaos Dionelis, Mehrdad Yaghoobi, and Sotirios A. Tsaftaris

Electrical Device Classification using Deep Learning 46
R.O. Lane

Session 5: Other Signal Processing

Multimodal Learning for Early Detection of Explosive Sounds using Relative Spectral Distribution 51
Vishwajeet Shukla and Mayank Singour

Detection of Linear Frequency Modulation, Phase-Coded and Multicarrier Radar Waveforms in Electronic Warfare Context 56
Antoine Foucalt, Cédric Cornu, Ali Khenchaf, and Fabrice Comblet

Robust Source Number Estimation Based on Denoising Preprocessing 61
Koichi Ichige, Shohei Hamada, Katsuhisa Kashiwagi, Nobuya Arakawa, and Ryo Saito

Learning Entropy of Adaptive Filters via Clustering Techniques 66
Ivo Bukovsky, Gejza Dohnal, Pavel Steinbauer, Ondrej Budik, Kei Ichiji, and Homma Noriyasu

Classifying LPI Signals with Transfer Learning on CNN Architectures..... 71
Bunlong Lay and Alexander Charlish

Approximate LASSO Model Predictive Control for Resource Constrained Systems	76
<i>Yun Wu, João F. C. Mota, and Andrew M. Wallace</i>	
Extraction of Analytic Eigenvectors from a Parahermitian Matrix.....	81
<i>Stephan Weiss, Ian K. Proudler, Fraser K. Coutts, and Julian Deeks</i>	
Session 6: Underwater Signal Processing	
Sensing and Automation in the Future Maritime Environment	86
<i>Daniel D. Sternlicht</i>	

Subspace Perturbation Bounds with an Application to Angle of Arrival Estimation using the MUSIC Algorithm

Connor Delaosa¹, Jennifer Pestana², Stephan Weiss¹, Ian K. Proudler¹

¹Dept. Electronic & Electrical Engineering, University of Strathclyde, Glasgow G1 1XW, Scotland

²Dept. Mathematics & Statistics, University of Strathclyde, Glasgow G1 1XH, Scotland

{connor.delaosa,jennifer.pestana,stephan.weiss,ian.proudler}@strath.ac.uk

Abstract—This paper explores how angle of arrival (AoA) estimation using the multiple signal classification (MUSIC) algorithm is affected by estimation errors in the space-time covariance matrix. In particular, we explore how this estimation error perturbs the signal-plus-noise and noise-only subspaces of the matrix, and how this subsequently affects the performance of MUSIC for AoA estimation. This subspace perturbation is shown to depend on the space-time covariance matrix itself, the sample size over which it is estimated, as well as the distance of the smallest signal-related eigenvalue to the noise floor. We link a bound on this perturbation to a bound on MUSIC performance, and demonstrate its utility for AoA estimation in simulations.

Index Terms—space-time covariance matrix, parahermitian matrix, cross-correlation sequence, estimation, angle of arrival, MUSIC

I. INTRODUCTION

In broadband array processing a space-time covariance matrix forms the basis for describing the second order statistics of the data recorded from M sensor measurements. These measurements, $\mathbf{x}[n] \in \mathbb{C}^M$, can be used to calculate the space-time covariance matrix as $\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^H[n-\tau]\} \in \mathbb{C}^{M \times M}$, where $\mathcal{E}\{\cdot\}$ is the expectation operator and $\{\cdot\}^H$ is the Hermitian transpose operator. In narrowband signal processing, which captures the phase shifts between sensor elements, it suffices to use the instantaneous covariance matrix $\mathbf{R}[0]$ only. If we move to the broadband domain then explicit time delay information must be preserved, and requires the inclusion of the lag component τ into the space-time covariance matrix $\mathbf{R}[\tau]$, which contains auto and -cross-correlation sequences. This space-time covariance matrix may need to be estimated such as in the situations of limited availability of data samples or when data must be restricted to a small time frame to assume stationarity. The estimation and the resulting perturbations of this matrix has been explored in the past [1]–[4].

In the broadband case, the multiple signal classification (MUSIC) algorithm [5] uses the space-time covariance matrix to estimate the angle of arrival (AoA) of a particular source. If these matrices are estimated from finite data, we can use

perturbation theory to gain a measure of how much this result is perturbed. Therefore, an understanding of how this algorithm performs under estimation is important. From [2], [6] we know the variance of the estimation error, and subsequently how eigenvalues and eigenspaces are perturbed as a result of the sample size, N . In this paper, we are concerned with how this perturbation of $\mathbf{R}[\tau]$ impacts the MUSIC algorithm and in particular how the algorithm degrades as a function of N .

In the past, papers have looked at the noise and signal subspaces crucial for the performance of MUSIC. In particular, [7], [8] show that when using the singular value decomposition (SVD) and the root-MUSIC algorithm, respectively, subspace methods for a small fixed sample size severely degrade as the signal-to-noise ratio (SNR) decreases greatly. In order to improve the direction of arrival (DoA) estimates for subspace methods in poor conditions, various papers modify the MUSIC algorithm in different ways such as introducing: a new estimator [9], an innovative MUSIC algorithm to decrease the threshold of SNR before degradation [10], or an iterative algorithm to reduce unwanted terms in the sample spatially correlated covariance matrix [11]. Under the influence of varying modelling errors [12] investigates the MUSIC estimator showing its performance degradation for higher error terms. In contrast, in this paper we look to extend [7], [8] to the broadband case where we use an eigenvalue decomposition (EVD) to obtain the subspaces as we vary N . We apply perturbation theory in order to understand the degradation of the traditional MUSIC algorithm for a fixed SNR and a range of sample sizes with the aim of establishing a link between a bound of this perturbation and a bound on the performance of MUSIC.

This paper is laid out as follows; in Sec. II the data model and the MUSIC algorithm based on ideal quantities are stated. Sec. III demonstrates the construction of an estimated space-time covariance matrix, the estimation error between this and the ideal case and how this error can lead to eigenvalues and eigenspaces being perturbed. In Sec. IV we relate the perturbation of subspaces to the MUSIC algorithm estimated from limited data and therefore derive a bound to characterise this relationship. We then, in Sec. V, verify the theory through simulation and test the performance of the bound and related

This work was supported in parts by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/S000631/1 and the MOD University Defence Research Collaboration in Signal Processing, Dstl, and a John Anderson Research Award by the University of Strathclyde.

quantities.

II. ANGLE OF ARRIVAL ESTIMATION

A. Data Model

Let $\mathbf{x}[n] \in \mathbb{C}^M$ be a data vector recorded from M sensor measurements $x_m[n]$, $m = 1, \dots, M$. This array is illuminated by K sources in the far-field, and while we assume no loss in gain, these source signals $s_k[n]$, $k = 1, \dots, K$, arrive at the different sensors with time delays $\tau_{k,m}$. Therefore, the propagation environment can be described by a broadband steering vector $\mathbf{a}_k[n]$,

$$\mathbf{a}_k[n] = \frac{1}{\sqrt{M}} \begin{bmatrix} f[n - \tau_{k,1}] \\ \vdots \\ f[n - \tau_{k,M}] \end{bmatrix}, \quad (1)$$

with $f[n - \tau]$ implementing a fractional delay filter [13], [14]. For the time delay $\tau_{k,m} = \frac{1}{cT_s} \mathbf{k}_k^T \mathbf{r}_m$, c denotes the propagation speed in the medium, T_s is the sampling period, \mathbf{k}_k is the normal vector to the wavefront of the k th source signal and \mathbf{r}_m is the location of the m th sensor in three-dimensional space. Using the steering vector, a model for the received data is

$$\mathbf{x}[n] = \sum_{k=1}^K \mathbf{a}_k[n] s_k[n] + \mathbf{v}[n], \quad (2)$$

where $\mathbf{v}[n]$ is spatially and temporally uncorrelated Gaussian noise with covariance $\mathcal{E}\{\mathbf{v}[n]\mathbf{v}^H[n - \tau]\} = \sigma_v^2 \mathbf{I} \delta[\tau]$ and \mathbf{I} is an $M \times M$ identity matrix.

The cross-spectral density (CSD) matrix, $\mathbf{R}(z)$, originates from the z -transform of $\mathbf{R}[\tau]$ i.e. $\mathbf{R}(z) = \sum_{\tau} \mathbf{R}[\tau] z^{-\tau}$, or in short $\mathbf{R}(z) \bullet \circ \mathbf{R}[\tau]$, where $\bullet \circ$ denotes a transform pair. In almost all cases $\mathbf{R}(z)$ admits a parahermitian matrix EVD (PhEVD) [15], [16] such that

$$\mathbf{R}(z) = \mathbf{Q}(z) \mathbf{\Lambda}(z) \mathbf{Q}^P(z). \quad (3)$$

Here $\mathbf{Q}(z) \in \mathbb{C}^{M \times M}$ is paraunitary i.e. $\mathbf{Q}(z) \mathbf{Q}^P(z) = \mathbf{I}$ where $\{\cdot\}^P$ denotes the parahermitian operator of a given matrix such that $\mathbf{R}(z) = \mathbf{R}^P(z) = \mathbf{R}^H(z^{-1})$. The quantity $\mathbf{Q}(z)$ is analytic [15] in most cases [17] and contains the eigenvectors while $\mathbf{\Lambda}(z) = \text{diag}\{\lambda_1(z), \dots, \lambda_M(z)\}$ holds the M eigenvalues.

Given the time domain model in (2), the CSD matrix for this scenario is

$$\mathbf{R}(z) = \sum_{k=1}^K \mathbf{a}_k(z) S_k(z) \mathbf{a}_k^P(z) + \sigma_v^2 \mathbf{I}, \quad (4)$$

where $S_k(z)$ is the power spectral density (PSD) of the k th source signal $s_k[n]$. This signal $s_k[n]$ can be tied to an uncorrelated zero-mean unit-variance Gaussian signal $u_k[n]$ via an innovation filter [18] $b_k[n] \bullet \circ B_k(z)$, such that $S_k(z) = B_k(z) B_k^P(z)$.

B. MUSIC based on Ideal Space-Time Covariance Matrices

Since the factors in (3) are analytic in some region of convergence including the unit circle, in the subsequent analysis we restrict ourselves to evaluating on the unit circle i.e. $z = e^{j\Omega}$. With knowledge of the number of sources, K , we partition, $\mathbf{Q}(z)|_{z=e^{j\Omega}}$ into two subspaces such that the K columns of $\mathbf{Q}_s(e^{j\Omega})$ span the K -dimensional signal-plus-noise subspace, while the columns of $\mathbf{Q}_n(e^{j\Omega})$ form the basis for its complement, the $M - K$ -dimensional noise-only subspace. Hence

$$\mathbf{R}(e^{j\Omega}) = \begin{bmatrix} \mathbf{Q}_s(e^{j\Omega}) & \mathbf{Q}_n(e^{j\Omega}) \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_s(e^{j\Omega}) & 0 \\ 0 & \mathbf{\Lambda}_n(e^{j\Omega}) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{Q}_s^H(e^{j\Omega}) \\ \mathbf{Q}_n^H(e^{j\Omega}) \end{bmatrix}, \quad (5)$$

where $\mathbf{\Lambda}_s(e^{j\Omega})$ contains the K eigenvalues associated with the signal-plus-noise subspace, and $\mathbf{\Lambda}_n(e^{j\Omega})$ contains the remaining $M - K$ eigenvalues.

With these quantities now defined, it can be stated that the steering vectors, $\mathbf{a}_{\varphi, \vartheta}(e^{j\Omega}) = [\mathbf{a}_1(e^{j\Omega}), \dots, \mathbf{a}_K(e^{j\Omega})]^T$ for azimuth φ and elevation ϑ , contribute to the signal-plus-noise subspace, $\mathbf{Q}_s(e^{j\Omega})$. These therefore reside within the nullspace of the noise subspace, $\mathbf{Q}_n(e^{j\Omega})$, i.e. the vector $\mathbf{Q}_n^H(e^{j\Omega}) \mathbf{a}_{\varphi, \vartheta}(e^{j\Omega})$ tends toward a zero vector if $\mathbf{a}_{\varphi, \vartheta}(e^{j\Omega})$ is the steering vector of a source. As a result, we can scan the noise-only subspace with steering vectors determined by a range of angles $\{\varphi, \vartheta\}$ and frequencies Ω such that

$$\xi_{\text{MUSIC}}(e^{j\Omega}, \varphi, \vartheta) = \frac{1}{\mathbf{a}_{\varphi, \vartheta}^P(e^{j\Omega}) \mathbf{Q}_n(e^{j\Omega}) \mathbf{Q}_n^P(e^{j\Omega}) \mathbf{a}_{\varphi, \vartheta}(e^{j\Omega})}, \quad (6)$$

which is also known as the ideal MUSIC spectrum [5]. In practice, this metric will be computed using estimated quantities, resulting in perturbations. The next two sections will therefore investigate the estimation errors and analyse the subspace perturbation of the noise-only subspace and its effect on MUSIC.

III. COVARIANCE ESTIMATION AND SUBSPACE PERTURBATIONS

A. Space-Time Covariance Estimation

Instead of determining $\mathbf{R}[\tau]$ via an expectation operation, in practice it has to be estimated in time over a finite window of snapshots, say N . Given the data vector $\mathbf{x}[n]$ for $n = 0, \dots, (N - 1)$, a sample space-time covariance matrix $\hat{\mathbf{R}}[\tau]$ can be calculated e.g. via an estimator [19]

$$\hat{\mathbf{R}}[\tau] = \begin{cases} \frac{1}{N-|\tau|} \sum_{n=0}^{N-|\tau|-1} \mathbf{x}[n + \tau] \mathbf{x}^H[n], & \tau \geq 0 \\ \frac{1}{N-|\tau|} \sum_{n=0}^{N-|\tau|-1} \mathbf{x}[n] \mathbf{x}^H[n - \tau], & \tau < 0 \end{cases}. \quad (7)$$

Because this estimator is unbiased, for the estimation error

$$\mathbf{E}[\tau] = \hat{\mathbf{R}}[\tau] - \mathbf{R}[\tau], \quad (8)$$

we find that its power, $\mathcal{E}\{\|\mathbf{E}[\tau]\|_{\mathbb{F}}^2\}$, equals the variance of $\hat{\mathbf{R}}[\tau]$, which has been shown to depend on both the ground truth $\mathbf{R}[\tau]$ and the sample size N [19].

B. Perturbation of Eigenspaces

With the PhEVD $\hat{\mathbf{R}}(z) = \hat{\mathbf{Q}}(z)\hat{\mathbf{\Lambda}}(z)\hat{\mathbf{Q}}^{\text{P}}(z)$ we partition the eigenvalues and eigenvectors similarly to (5). The eigenvalues in $\hat{\mathbf{\Lambda}}(e^{j\Omega}) = \text{blockdiag}\{\hat{\mathbf{\Lambda}}_{\text{s}}(e^{j\Omega}) \hat{\mathbf{\Lambda}}_{\text{n}}(e^{j\Omega})\}$ are split into a K -element diagonal matrix containing the eigenvalues corresponding to the signal-plus-noise-subspace and an $M - K$ diagonal matrix of values that form the noise floor. The eigenvectors are then defined as $\hat{\mathbf{Q}}(e^{j\Omega}) = [\hat{\mathbf{Q}}_{\text{s}}(e^{j\Omega}) \hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega})]$, such that the subspaces are given by $\hat{\mathcal{U}}_{\text{s,n}}(e^{j\Omega}) = \text{range}\{\hat{\mathbf{Q}}_{\text{s,n}}(e^{j\Omega})\}$, with the ideal subspaces being denoted similarly. With these quantities, we assess the difference between the ground-truth noise-only subspace $\mathcal{U}_{\text{n}}(e^{j\Omega})$ and its estimated version $\hat{\mathcal{U}}_{\text{n}}(e^{j\Omega})$ as

$$\text{dist}\{\hat{\mathcal{U}}_{\text{n}}(e^{j\Omega}), \mathcal{U}_{\text{n}}(e^{j\Omega})\} = \|\hat{\mathbf{P}}_{\text{n}}(e^{j\Omega}) - \mathbf{P}_{\text{n}}(e^{j\Omega})\|_2, \quad (9)$$

where $\hat{\mathbf{P}}_{\text{n}}(e^{j\Omega}) = \hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega})\hat{\mathbf{Q}}_{\text{n}}^{\text{H}}(e^{j\Omega})$ and $\mathbf{P}_{\text{n}}(e^{j\Omega}) = \mathbf{Q}_{\text{n}}(e^{j\Omega})\mathbf{Q}_{\text{n}}^{\text{H}}(e^{j\Omega})$ are projection operators into the noise-only subspaces. To find bounds on the eigenspace perturbations we must first define the spectral distance δ between the signal-plus-noise and noise-only eigenvalues as

$$\delta = \min_{\substack{\lambda_1 \in \mathbf{\Lambda}_{\text{s}}(e^{j\Omega}) \\ \lambda_2 \in \mathbf{\Lambda}_{\text{n}}(e^{j\Omega})}} |\lambda_1 - \lambda_2| > 0; \quad (10)$$

in other words, δ is the absolute minimum difference between the signal-plus-noise and noise-only eigenvalues.

If the estimation error [6, Sec. 7.2] satisfies

$$\|\mathbf{E}(e^{j\Omega})\|_2 < \delta/5, \quad (11)$$

we can state that the subspace distance is bounded as follows [2], [6]:

$$\text{dist}\{\hat{\mathcal{U}}_{\text{n}}(e^{j\Omega_0}), \mathcal{U}_{\text{n}}(e^{j\Omega_0})\} \leq \frac{4}{\delta} \|\mathbf{E}_{\text{sn}}(e^{j\Omega_0})\|_2. \quad (12)$$

Here $\mathbf{E}_{\text{sn}}(e^{j\Omega_0})$ is extracted from the following partitioning of a similarity transform of the estimation error matrix $\mathbf{E}(e^{j\Omega_0}) = \mathbf{E}(e^{j\Omega})|_{e^{j\Omega}=e^{j\Omega_0}}$ with $\mathbf{E}(e^{j\Omega}) \bullet \text{---} \mathbf{E}[\tau]$,

$$\mathbf{Q}^{\text{H}}(e^{j\Omega_0})\mathbf{E}(e^{j\Omega_0})\mathbf{Q}(e^{j\Omega_0}) = \begin{bmatrix} \underbrace{\mathbf{E}_{\text{s}}(e^{j\Omega_0})}_{K} & \underbrace{\mathbf{E}_{\text{sn}}^{\text{H}}(e^{j\Omega_0})}_{M-K} \\ \underbrace{\mathbf{E}_{\text{sn}}(e^{j\Omega_0})}_{K} & \underbrace{\mathbf{E}_{\text{n}}(e^{j\Omega_0})}_{M-K} \end{bmatrix}. \quad (13)$$

From the quantities defined, we can state that the subspace perturbation depends on: the ground truth covariance matrix, $\mathbf{R}(e^{j\Omega})$, the sample size N via $\mathbf{E}_{\text{sn}}(e^{j\Omega_0})$, and the distance δ i.e. by how much the ‘signal eigenvalue’ differs from the noise floor. We can now use this in Sec. IV to analyse the MUSIC algorithm.

IV. MUSIC BASED ON SAMPLE COVARIANCE MATRICES

Following Secs. II and III-A, where the ideal MUSIC algorithm and the estimated quantities are introduced, we formulate the estimated MUSIC cost function in Sec. IV-A below. The link between its inverse and the subspace perturbation bound is then investigated in Sec. IV-B.

A. Estimation of MUSIC

Using the noise-only subspace $\hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega_0})$, we obtain

$$\begin{aligned} \hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi, \vartheta) &= \mathbf{a}^{\text{H}}(e^{j\Omega_0})\hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega_0})\hat{\mathbf{Q}}_{\text{n}}^{\text{H}}(e^{j\Omega_0})\mathbf{a}(e^{j\Omega_0}) \\ &= \|\hat{\mathbf{Q}}_{\text{n}}^{\text{H}}(e^{j\Omega_0})\mathbf{a}_{\varphi, \vartheta}(e^{j\Omega_0})\|_2^2, \end{aligned} \quad (14)$$

which defines the MUSIC spectrum derived from estimated quantities. Ideally, (14) should be zero at the angles of arrival but, due to the use of the estimated covariance matrix, it can deviate from zero by an unknown amount. Since the MUSIC metric in (14) is based on the estimated subspaces we can now relate this to Sec. III-B and describe the relationship between $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)$ and the perturbation of subspaces.

B. Perturbation Bounds of the MUSIC Metric

In the following, we analyse the inverse MUSIC spectrum in (14) in the direction of arrival of the source, $\{\varphi_0, \vartheta_0\}$, only. For notational brevity, we omit subscripts from the steering vector.

Since $\hat{\mathbf{Q}}_{\text{n}}^{\text{H}}(e^{j\Omega_0})\hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega_0}) = \mathbf{I} \in \mathbb{R}^{K \times K}$ (14) becomes

$$\begin{aligned} \hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0) &= \mathbf{a}^{\text{H}}(e^{j\Omega_0})\hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega_0})\hat{\mathbf{Q}}_{\text{n}}^{\text{H}}(e^{j\Omega_0}) \\ &\quad \cdot \hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega_0})\hat{\mathbf{Q}}_{\text{n}}^{\text{H}}(e^{j\Omega_0})\mathbf{a}(e^{j\Omega_0}) \\ &= \|\hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega_0})\hat{\mathbf{Q}}_{\text{n}}^{\text{H}}(e^{j\Omega_0})\mathbf{a}(e^{j\Omega_0})\|_2^2. \end{aligned} \quad (15)$$

Since $\mathbf{Q}_{\text{n}}^{\text{H}}(e^{j\Omega_0})\mathbf{a}(e^{j\Omega_0}) = \mathbf{0}$, we can write (15) such that

$$\begin{aligned} \hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0) &= \|\hat{\mathbf{Q}}_{\text{n}}(e^{j\Omega_0})\hat{\mathbf{Q}}_{\text{n}}^{\text{H}}(e^{j\Omega_0})\mathbf{a}(e^{j\Omega_0}) - \\ &\quad - \mathbf{Q}_{\text{n}}(e^{j\Omega_0})\mathbf{Q}_{\text{n}}^{\text{H}}(e^{j\Omega_0})\mathbf{a}(e^{j\Omega_0})\|_2^2 \\ &\leq \|\hat{\mathbf{P}}_{\text{n}}(e^{j\Omega_0}) - \mathbf{P}_{\text{n}}(e^{j\Omega_0})\|_2^2 \|\mathbf{a}(e^{j\Omega_0})\|_2^2 \\ &= \text{dist}\{\hat{\mathcal{U}}_{\text{n}}(e^{j\Omega_0}), \mathcal{U}_{\text{n}}(e^{j\Omega_0})\}^2. \end{aligned} \quad (16)$$

Here we have that $\|\mathbf{a}(e^{j\Omega_0})\|_2 = 1$ by design and using (9) with $\hat{\mathcal{U}}_{\text{n}}(e^{j\Omega_0})$ and $\mathcal{U}_{\text{n}}(e^{j\Omega_0})$.

Using (12), we can relate $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)$ to the error matrix in (13) and the eigenvalue distance, δ , in (10) such that

$$\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0) \leq \frac{16}{\delta^2} \|\mathbf{E}_{\text{sn}}(e^{j\Omega_0})\|_2^2, \quad (17)$$

whenever (11) is satisfied. Thus we can link the inverse MUSIC spectrum at the source AoA to the derived eigenspace perturbation in (16) and to its bound via (17).

V. RESULTS AND SIMULATIONS

A. Scenario

As an example, we look at the case of a single source ($K = 1$) impinging on an array of $M = 3$ sensors from an angle of arrival of $\varphi_0 = -60^\circ$ and no elevation i.e. $\vartheta_0 = 0^\circ$. The innovation filter $b_1[n]$ of Sec. II-A has a passband response, where the normalised passband frequencies are defined as $\Omega_{\text{pass}} = [0.2\pi \ 0.9\pi]$. This filter is excited by a zero-mean unit-variance uncorrelated complex circularly symmetric Gaussian source. The sample space-time covariance matrix, $\hat{\mathbf{R}}[\tau]$, is estimated from N snapshots of data $\mathbf{x}[n]$ for

$n = 0, \dots, N - 1$ samples and with the support value, τ adjusted to be optimal [4]. For the simulations below, we vary the sample size from 200 to 10000 samples in intervals of 100 samples. Each parameter setting is tested over an ensemble of 10^4 simulations so that we can analyse the statistics of the results. We consider only one fixed frequency at $\Omega_0 = \frac{\pi}{4}$.

B. MUSIC and Subspace Perturbations

For the following results, we are interested in how the quantities in (11) and (17) perform, and investigate the validity of these for different N . For clarity, we convert (17) into a ratio such that

$$\gamma(e^{j\Omega_0}) = \hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0) / \frac{16}{\delta^2} \|\mathbf{E}_{\text{sn}}(e^{j\Omega_0})\|_2^2, \quad (18)$$

which satisfies $\gamma(e^{j\Omega_0}) \leq 1$ if (11) is satisfied.

As discussed previously, the subspace distance is related to a condition specified in (11). We first look at the proportion of total simulations that satisfy (11) as a function of sample size this is shown in Fig. 1 where we see this proportion increases as the sample size, N , increases which is to be expected. From this, it is important to note that often for smaller sample size we fail to satisfy the condition for the simulations carried out in this paper. We denote the simulations that pass this condition by the $\{\cdot\}^{(\text{pass})}$ superscript.

In Fig. 2 we investigate the distribution of (18) for the simulations that pass (11) and from this result we verify that if (11) is satisfied then (17) is always satisfied. Since we often do not satisfy the condition in (11) we look at the distribution of $\gamma(e^{j\Omega_0})$ for all simulations to see the overall performance of (18) and, therefore, test if (17) is useful independently of the condition. The result from Fig. 3 shows that as we increase N more simulations satisfy these bounds, with the median of $\gamma(e^{j\Omega_0})$ almost constant. For the curve given by the 95th percentile, it can be seen that at $N \geq 400$ samples $\gamma(e^{j\Omega_0})$ crosses 10^0 and satisfies (17) and (18) for at least 95% of simulation runs. For $N < 400$ we notice that for many of the trials, the 95th percentile curve is relatively close to one.

In Fig. 4 it can be seen that even though the distribution in Fig. 3 has 95th percentiles less than one, we still have a small percentage of simulations that fail to satisfy (17) for small sample sizes. As we increase the sample size to $N = 2000$ this percentage tends to 0 showing that all simulations satisfy the bound regardless of the condition in (11).

C. Extraction of the Angle of Arrival

In Sec. IV-B, we were concerned by how $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)$ is affected at the ideal/expected angle of arrival. We now investigate the extraction of the angle of arrival from the inverse MUSIC spectra based on estimates.

If estimated quantities are used then the inverse MUSIC spectrum can vary and where we would expect the angle of arrival to be extracted i.e. at the global minima of $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)$, could shift. However, as we have discussed, the use of more data can result in more accurate estimates. In this example, we extract an angle of arrival,

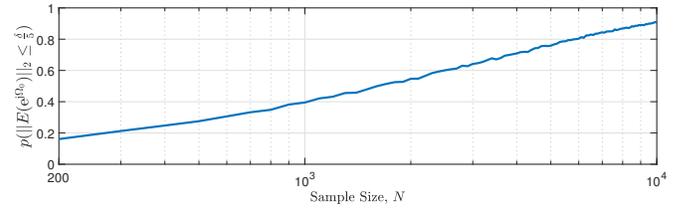


Fig. 1. Proportion of simulations that satisfy (11).

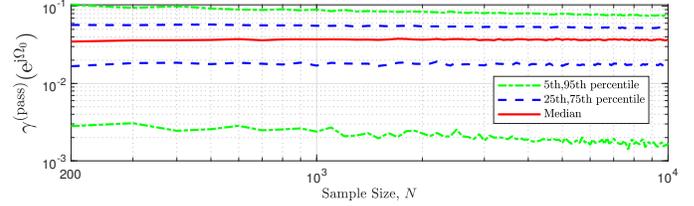


Fig. 2. Distribution of (18) for simulations that satisfy (11) with 5, 25, 75, and 95th percentiles, where the median is shown by the solid red line.

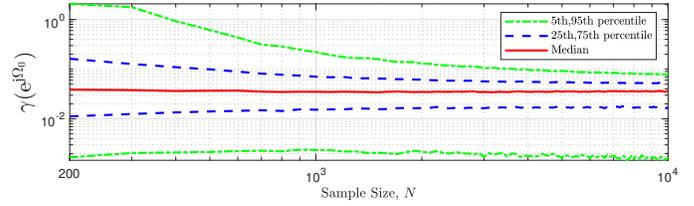


Fig. 3. Distribution of (18) for all ensemble probes with 5, 25, 75, and 95th percentiles, where the median is shown by the solid red line.

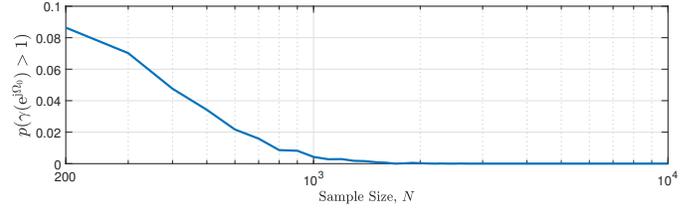


Fig. 4. Proportion of all ensemble probes that satisfy the ratio given in (18).

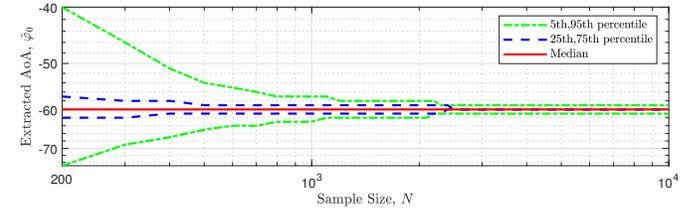


Fig. 5. Distribution of the extracted AoA, $\hat{\varphi}_0$, from estimated MUSIC spectra with 5, 25, 75, and 95th percentiles, where the median is shown by the solid red line.

$\hat{\varphi}_0 = \min\{\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)\}$ from each inverse MUSIC spectra calculated from each simulation and show the results as a statistical distribution.

It can be seen from Fig. 5 that there is a large deviation at low sample sizes, so that if the global minimum was used then we could extract an angle of arrival that is far from the correct value. As we increase the number of samples then we are more likely to extract values closer to the true value with minimal variation. The median of the distribution is constant

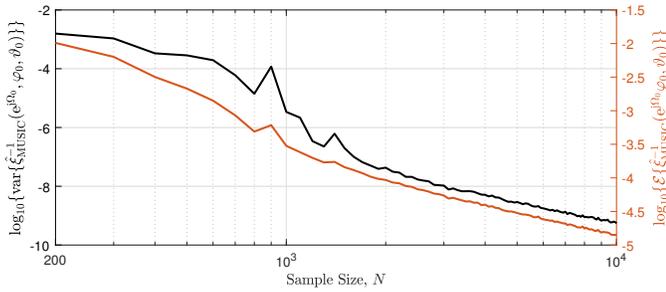


Fig. 6. Ensemble average (right) and variance (left) of $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)$ as a function of N on a logarithmic scale.

throughout the large number of simulations and indicates that we can extract the correct estimate but not consistently as the percentiles demonstrate. In defense applications, it may be useful to track targets within a certain angular range. The results in Fig. 5 shows that this could be applied for more targets/sources or for a lower sample size if the restriction of a single accurate AoA was lifted. For the scenario in this paper, a sample size $N < 200$ showed a heavy degradation in the extracted angle to the point that this angle can be ambiguous.

D. Mean and Variance of MUSIC

Given that we have discussed bounds and conditions related to the estimated MUSIC metric, $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)$, we now look at this metric directly. If $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)$ is evaluated at the ideal angle of arrival then we would expect that this value should be 0. Due to the estimated quantities we would now expect $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0) \rightarrow 0$ as $N \rightarrow \infty$. Therefore, if a distribution of this function was calculated for all simulations then the mean and variance of this function should decrease as N increases i.e. we should increase the reliability of $\hat{\xi}_{\text{MUSIC}}^{-1}(e^{j\Omega_0}, \varphi_0, \vartheta_0)$ as we increase N .

As demonstrated in Fig. 6 we notice that both the ensemble average and variance decrease on a logarithmic scale. This, combined with Fig. 5, allows us to see that the reliability of the inverse MUSIC algorithm estimated from quantities calculated from finite data increases as we use more data.

VI. CONCLUSION

When estimating a space-time covariance matrix, it is imperative to understand the effect of perturbations before using real-world applications such as the MUSIC algorithm. In this paper we have: (i) discussed how eigenspaces can be perturbed as a result of using space-time covariance matrices estimated from limited data, (ii) stated theoretical bounds for the perturbations of subspaces, and (iii) provided a condition which depends on the error between ideal and estimated space-time covariance matrices and the distance between signal-plus-noise and noise-only eigenvalues for the ideal case.

We have shown that subspace perturbations can be used to derive a theoretical bound to describe the degradation of the MUSIC algorithm, evaluated at a single frequency point, as a function of the sample size used. We tested the performance of this bound as well as the condition on which it depends.

For the simulations carried out it can be verified that if the condition is satisfied then this bound is always satisfied for all sample sizes tested. If this condition is not satisfied e.g. for low sample sizes, then it can be shown that the bound is satisfied 75% of the time.

If we now relate these findings to the extraction of an AoA then this can be shown to vary as a function of the sample size. Given a large number of samples we can successfully extract the AoA with small statistical variation. Additionally as we increase the number of samples, the extracted AoA tends towards the ideal case with decreasing variance.

REFERENCES

- [1] X. Mestre, "Improved estimation of eigenvalues and eigenvectors of covariance matrices using their sample estimates," *IEEE Trans. Information Theory*, vol. 54, no. 11, pp. 5113–5129, Nov 2008.
- [2] C. Delaosa, F. K. Coutts, J. Pestana, and S. Weiss, "Impact of space-time covariance estimation errors on a parahermitian matrix EVD," in *IEEE SAM*, Sheffield, UK, July, July 2018, pp. 1–5. [Online]. Available: <https://strathprints.strath.ac.uk/63998/>
- [3] X. Mestre, "On the asymptotic behavior of the sample estimates of eigenvalues and eigenvectors of covariance matrices," *IEEE Trans. Signal Processing*, vol. 56, no. 11, pp. 5353–5368, 2008.
- [4] C. Delaosa, J. Pestana, N. J. Goddard, S. D. Somasundaram, and S. Weiss, "Support estimation of a sample space-time covariance matrix," in *SSPD*, Brighton, UK, May, May 2019, pp. 1–5.
- [5] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," vol. 34, no. 3, pp. 276–280, March 1986.
- [6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed. Baltimore, Maryland: John Hopkins University Press, 2013.
- [7] J. K. Thomas, L. L. Scharf, and D. W. Tufts, "The probability of a subspace swap in the SVD," *IEEE Trans. Signal Processing*, vol. 43, no. 3, pp. 730–736, 1995.
- [8] M. Shaghghi and S. A. Vorobyov, "Subspace leakage analysis and improved DOA estimation with small sample size," *IEEE Trans. Signal Processing*, vol. 63, pp. 3251–3265, 2015.
- [9] E. Ollila and V. Koivunen, "Robust antenna array processing using m-estimators of pseudo-covariance," in *IEEE PIMRC*, vol. 3:2659-2663, Beijing, China, Sep. 2003.
- [10] X. Mestre and F. Rubio, "An improved subspace-based algorithm in the small sample size regime," in *IEEE ICASSP*, vol. 4:1073-1076, Toulouse, France, 2006.
- [11] M. Shaghghi and S. A. Vorobyov, "Iterative root-music algorithm for DOA estimation," in *IEEE CAMSAP*, St. Martin, France, Dec 2013, pp. 53–56.
- [12] A. Kangas, P. Stoica, and T. Soderstrom, "Finite sample and modelling error effects on esprit and music direction estimators," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 141, no. 5, pp. 249–255, 1994.
- [13] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine, "Splitting the Unit Delay," *IEEE Signal Processing Magazine*, vol. 13:30-60, January 1996.
- [14] J. Selva, "An efficient structure for the design of variable fractional delay filters based on the windowing method," *IEEE Trans. Signal Processing*, vol. 56:3770–3775, Aug. 2008.
- [15] S. Weiss, J. Pestana, and I. K. Proudler, "On the existence and uniqueness of the eigenvalue decomposition of a parahermitian matrix," *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2659–2672, May 2018.
- [16] S. Weiss, J. Pestana, I. K. Proudler, and F. K. Coutts, "Corrections to "on the existence and uniqueness of the eigenvalue decomposition of a parahermitian matrix"," *IEEE Trans. Signal Processing*, vol. 66, no. 23, pp. 6325–6327, May 2018.
- [17] S. Weiss, I. K. Proudler, F. K. Coutts, and J. Pestana, "Iterative approximation of analytic eigenvalues of a parahermitian matrix EVD," in *IEEE ICASSP*, Brighton, UK, May 2019, pp. 8038–8042.
- [18] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.
- [19] C. Delaosa, J. Pestana, N. J. Goddard, S. Somasundaram, and S. Weiss, "Sample space-time covariance matrix estimation," in *IEEE ICASSP*, Brighton, UK, 2019, pp. 8033–8037.

A Gaussian Process based Method for Multiple Model Tracking

Mengwei Sun, Mike E. Davies, Ian Proudler, James R. Hopgood

Abstract—Manoeuvring target tracking faces the challenge caused by the target motion model uncertainty, i.e., unknown model types or uncertain model parameters. Multiple-model (MM) methods have been generally considered to deal with this challenge, in which a bank of elemental filters is run simultaneously to provide a joint decision and estimation of motion model and localisation. However, if the uncertainty of the target trajectory increases, such as the target moves under mixed manoeuvring behaviours with time-varying parameters, more filters with different model assumptions have to be taken into account to match the motion of the target, which may lead to prohibitive computational complexity. To address this problem, we establish a training based algorithm which can learn the actual motion model as a Gaussian process (GP) regression. Then, by integrating the trained GP into the particle filter (PF), a GP-PF based tracking method is developed to track the manoeuvring targets in real-time. Monte Carlo experiments show that the proposed method had much lower tracking root mean square error (RMSE) and robustness compared with the most commonly used MM methods.

Index Terms—Gaussian process, manoeuvring target tracking, mixing manoeuvres, particle filtering

I. INTRODUCTION

TARGET tracking is a fundamental task in sensor-based applications, such as radar, sonar, and navigation [1]. Numerous mathematical models have been developed to approximate the motion trajectory of the target [2], such as the commonly used non-manoevring model, i.e., constant velocity (CV) model and highly manoeuvring model, i.e., coordinated turn (CT) model. The application of Bayesian filters for target tracking is based on the actual state-space model of objects, therefore being conditional on a motion model. However, in reality, there may be significant motion-model uncertainty when targets undergo unknown or mixed manoeuvring behaviours [3][4]: i.e., the evolution of the target state is too complex to be approximated as one specific mathematical model. Uncertainty can also be caused due to various parameters not being known a priori, or if they change with time. Examples of such parameters include: the turn rate when CT model is considered [5], or the process noise level. If incorrect models or parameters are applied,

the tracking performance of traditional Bayesian filters would degrade and even become unacceptable.

To deal with this problem, multiple-model (MM) methods are commonly used [3]. The basic idea of MM methods is to assume a set of motion models as possible candidates for the manoeuvring target, and then use a bank of elemental filters with these different models to capture the mixed motion behaviours of the target, and generate the overall estimation based on the results achieved by each elemental filter [3]. MM based methods can achieve satisfactory tracking performance when the uncertainty is low. However, in some cases, such as when the rotation is a time-varying or doubly-stochastic process, a sufficiently large set of models with different parameters is required to cover the range of possible motion models, which leads to a high computation complexity [4]. To address this aspect, [4] and [5] incorporated adaptive parameter estimation methods into the Bayesian filter framework, where the unknown parameter is estimated by approximating its distribution with particles and corresponding weights. However, as parametric models are not always able to capture all aspects of the motion behaviours, those methods might fail if the chosen motion model set is incapable of modelling the ground truth trajectory.

As an alternative to traditional Bayesian methods, machine learning based tracking algorithms have been proposed recently [6]-[8]. In [6], a quadruplet convolutional neural network (CNN) based algorithm was designed for multi-target tracking. A long short-term memory (LSTM) neural network was used in [7] to perform the prediction step of single-target localisation, which can improve the tracking accuracy and facilitate the use of computationally efficient low-dimensional state spaces. Wahlstrom [8] proposed an extended target tracking method using a Gaussian process (GP) to estimate the shape of the object whose moving trajectory followed a linear CV model. These methods achieved better performance and higher flexibility, but are limited to a fixed but unknown motion behaviour and did not take into consideration the case of mixed motion behaviours with uncertain parameters.

To overcome the limitations of existing methods regarding to mixed motion behaviours, in this paper, Gaussian process regression [9]-[11] is introduced to learn the motion model of the moving target. The key advantage of GP is the flexibility for modelling the uncertain systems, and the ability to learn noise and parameters from training data. The contributions of this paper can be summarised into two aspects. First, we develop a GP-PF based tracking method, showing how

M. W. Sun, M. E. Davies and J. R. Hopgood are with Institute of Digital Communications, University of Edinburgh, Edinburgh, EH9 3FG, U.K. E-mail: (msun; James.Hopgood; Mike.Davies)@ed.ac.uk.

I. Proudler is with the Centre for Signal Image Processing (CeSIP), Department of Electronic Electrical Engineering, University of Strathclyde, Glasgow, G1 1XW, U.K. E-mail: ian.proudler@strath.ac.uk.

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/S000631/1; and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

to integrate the trained GP prediction motion model into a particle filter (PF) tracking framework. Second, the GP-PF based method is compared to interacting MM (IMM) methods with different number of elemental filters. The simulation results show the improved tracking performance and higher robustness of the proposed method.

The remainder of this paper is organised as follows. The description of the formulated problem and GP regression are presented in Section II. The proposed algorithm is developed in Section III. Simulation results and performance comparisons are presented in Section IV. Finally, conclusions are provided in Section V.

II. BACKGROUND

A. Problem Formulation

For simplicity and clarity, a 2-D horizontal model is considered but it is easily extended to 3-D. The discrete time state-space model for tracking a single moving target is described by:

$$\mathbf{x}_{t+1} = f_t(\mathbf{x}_t) + \mathbf{v}_t, \quad (1)$$

$$\mathbf{z}_t = h(\mathbf{x}_t) + \mathbf{e}_t. \quad (2)$$

Here, $\mathbf{x}_t = [\xi_t, \dot{\xi}_t, \eta_t, \dot{\eta}_t]^T$ denotes the target state at time t in a 2-D plane, and (ξ_t, η_t) and $(\dot{\xi}_t, \dot{\eta}_t)$ represent the target position and the corresponding velocity in X-axis and Y-axis, respectively. The velocity is a vector with its magnitude as the speed. \mathbf{z}_t denotes the measurement detected by the sensor. The state motion functions defined as $f_t(\mathbf{x}_t) : \mathbb{R}^4 \rightarrow \mathbb{R}^4$ dynamically follow different motion models, and two typical motion models are considered in this paper, the non-manoeuving CV model, and the highly manoeuvring CT model with unknown turn rate. Note, the motion models assumed are not a given which means that more sophisticated state space models can also be used equally. The measurement function $h(\mathbf{x}_t) : \mathbb{R}^4 \rightarrow \mathbb{R}^2$ is defined as:

$$\mathbf{z}_t = H\mathbf{x}_t + \mathbf{e}_t, \quad (3)$$

where,

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (4)$$

$\mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, Q)$ and $\mathbf{e}_t \sim \mathcal{N}(\mathbf{0}, R)$ denote the process noise and the measurement noise, respectively. The matrix $Q = qI_{4 \times 4}$ and $R = rI_{2 \times 2}$, where q and r are the corresponding noise variance and are assumed to be static in this paper.

Based on Bayes' rule, the posterior probability density function (PDF) of the target state vector conditioned on the measurements is,

$$p(\mathbf{x}_{1:t} | \mathbf{z}_{1:t}) \propto p(\mathbf{z}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{x}_{1:t-1}) p(\mathbf{x}_{1:t-1} | \mathbf{z}_{1:t-1}), \quad (5)$$

where $p(\mathbf{z}_t | \mathbf{x}_t) = \mathcal{N}(\mathbf{z}_t | h(\mathbf{x}_t), R)$ describes the likelihood of the measurement given the state. The quantity $p(\mathbf{x}_t | \mathbf{x}_{1:t-1})$ is the prediction probability which can be calculated based on the real-time motion model $f_{t-1}(\mathbf{x}_{t-1})$, i.e., $p(\mathbf{x}_t | \mathbf{x}_{1:t-1}) = \mathcal{N}(\mathbf{x}_t | f_{t-1}(\mathbf{x}_{t-1}), Q)$. Typically, the motion model is a

parametric description for the underlying evolution process. While in practice, motions of the target may not follow parametric models. Therefore, GP is used to identify the motion models by placing GP priors on the unknown functions and will be introduced in the following.

B. Gaussian Process Regression

GP regression is a popular means of non-parametric nonlinear modelling that can directly capture model uncertainty. It attempts to model a function $g(\mathbf{x})$ by providing both the mean function $m(\mathbf{x})$ and covariance/kernel function $k(\mathbf{x}, \mathbf{x}')$ [11]:

$$g(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \quad (6)$$

$$m(\mathbf{x}) = \mathbb{E}[g(\mathbf{x})], \quad (7)$$

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(g(\mathbf{x}) - m(\mathbf{x}))(g(\mathbf{x}') - m(\mathbf{x}'))]. \quad (8)$$

Let us consider a general GP regression problem with noisy observations from an unknown function described as:

$$y = g(\mathbf{x}) + \mathbf{v}, \quad \mathbf{v} \sim \mathcal{N}(0, \sigma_v^2 I). \quad (9)$$

The training inputs are denoted as $X = [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{N-1}]$ and the outputs are $\mathbf{y} = [y_0, y_1, \dots, y_{N-1}]^T$. The purpose of GP is to derive the latent distribution of the function $\mathbf{g}^* = [g_0^*, \dots, g_{T-1}^*]^T$ at the test inputs $X^* = [\mathbf{x}_0^*, \dots, \mathbf{x}_{T-1}^*]^T$, conditioned on the training data set $\mathcal{D} = \{X, \mathbf{y}\}$. The joint distribution of the training measurement values \mathbf{y} and the test function value at one test point, i.e., $g_t^* = g(\mathbf{x}_t^*)$, is given as,

$$\begin{bmatrix} \mathbf{y} \\ g_t^* \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mathbf{m}(X) \\ m(\mathbf{x}_t^*) \end{bmatrix}, \begin{bmatrix} K(X, X) + \sigma_v^2 I & \mathbf{k}(X, \mathbf{x}_t^*) \\ \mathbf{k}(\mathbf{x}_t^*, X) & k(\mathbf{x}_t^*, \mathbf{x}_t^*) \end{bmatrix} \right). \quad (10)$$

where $\mathbf{k}(X, \mathbf{x}_t^*) = [k(\mathbf{x}_0, \mathbf{x}_t^*), \dots, k(\mathbf{x}_{N-1}, \mathbf{x}_t^*)]^T$, $k(\mathbf{x}_t^*, \mathbf{x}_t^*)$ is the covariance of $g(\mathbf{x}_t^*)$, and $K(X, X)$ denotes the covariance matrix for the training input data:

$$K(X, X) = \begin{bmatrix} k(\mathbf{x}_0, \mathbf{x}_0) & \dots & k(\mathbf{x}_0, \mathbf{x}_{N-1}) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_{N-1}, \mathbf{x}_0) & \dots & k(\mathbf{x}_{N-1}, \mathbf{x}_{N-1}) \end{bmatrix}. \quad (11)$$

From (10), the conditional distribution of the test function is derived as,

$$(g_t^* | \mathcal{D}) \sim \mathcal{N}(\bar{g}_t^*, \mathbb{V}[g_t^*]), \quad (12)$$

$$\bar{g}_t^* = m(\mathbf{x}_t^*) + \mathbf{k}_t^{*T} [K + \sigma_v^2 I]^{-1} (\mathbf{y} - \mathbf{m}(X)), \quad (13)$$

$$\mathbb{V}[g_t^*] = k(\mathbf{x}_t^*, \mathbf{x}_t^*) - \mathbf{k}_t^{*T} [K + \sigma_v^2 I]^{-1} \mathbf{k}_t^*. \quad (14)$$

III. GP-PF BASED TRACKING ALGORITHM

In this section a tracking algorithm is proposed where the uncertainty and mixing behaviour of the motion model is first learned by non-parametric stationary and GP regression off-line, i.e., the training process, and then the learned model is integrated into the PF framework to estimate the target states on-line, i.e., the test process. The notations used in this section are listed in Table I. The motion model $f(\mathbf{x})$ is

TABLE I: The meaning of notations used in Section III.

Notation	Meaning
$\mathbf{x}_{0:N}$	Target state of training data sequence
$\mathbf{z}_{0:N}$	Observation of training data sequence
$\mathbf{x}_{0:T}^*$	Target state of test data sequence
$\mathbf{z}_{0:T}^*$	Observation of test data sequence
(ξ_t, η_t)	Position of target at 2-D plane
$(\dot{\xi}_t, \dot{\eta}_t)$	Velocity of target at 2-D plane
Q	Covariance matrix of process noise
R	Covariance matrix of measurement noise
\mathcal{D}	Training data set for GP
\mathcal{X}	State space
\mathcal{Z}	Observation space
$\mathbf{f}_{0:N}(\mathbf{x}_{0:N})$	Motion function of training process
$\mathbf{f}_{0:T}^*(\mathbf{x}_{0:T}^*)$	Motion function of test process
$h(\mathbf{x}_t)$	Measurement function

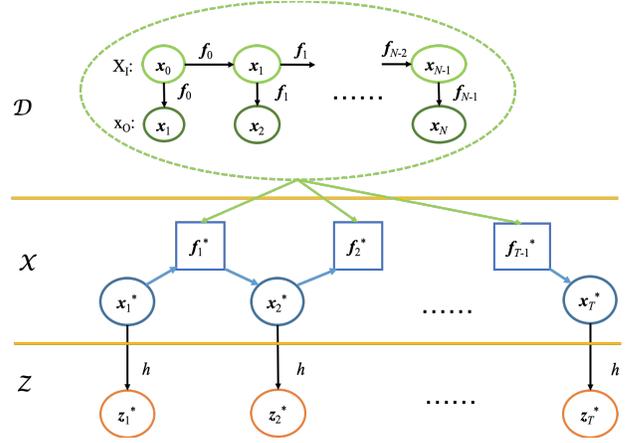


Fig. 1: State-space models: GP-based state transition and parametric measurement functions. Circle represents the variable node and square is the non-parametric function node. \mathcal{D} , \mathcal{X} and \mathcal{Z} represent the training data space, test state space and test observation space respectively.

learned using the non-parametric GP, while the measurement function $h(\mathbf{x})$ is deterministic by different sensor type with no correlation over time and is given in arbitrary parametric form in (3). Hence, the state-space model shown in (1)-(2) can be graphically illustrated in Fig. 1. Our goal is to integrate the learned GP regression into the PF framework to estimate the target states. Specifically, the motion function with mixed behaviours is approximated by a prediction Gaussian process, which is then applied into the PF framework for drawing particles. Next, the correction step is performed by updating the weights of particles and achieving the estimations of target states based on the maximum a posterior probability (MAP) criterion.

To train the state transition function, we take training data set $\mathcal{D} := \{X_I, \mathbf{x}_O\} \subset \mathcal{X} \times \mathcal{X}$ as the input and output respectively, where $X_I = [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{N-1}]$ and $\mathbf{x}_O = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]^T$, and $\mathbf{x}_n = [\xi_n, \dot{\xi}_n, \eta_n, \dot{\eta}_n]^T$. The test sequence are represented as $\{\mathbf{x}_{0:T}^*, \mathbf{z}_{0:T}^*\} \subset \mathcal{X} \times \mathcal{Z}$, where $\mathbf{z}_{0:T}^* := (z_0^*, \dots, z_T^*)$ is the sequence of test observations while the target states $\mathbf{x}_{0:T}^* := (\mathbf{x}_0^*, \dots, \mathbf{x}_T^*)$ are unknown and need to be estimated sequentially.

A. Learning the Motion Model based on GP Regression

The joint conditional distribution of the states for the test process can be partitioned as [11],

$$\begin{aligned} p(\mathbf{x}_{0:T}^* | \mathcal{D}) &= \int p(\mathbf{x}_{0:T}^*, \mathbf{f}_{0:T}^* | \mathcal{D}) d\mathbf{f}_{0:T}^*, \\ &= \int p(\mathbf{x}_{0:T}^* | \mathbf{f}_{0:T}^*, \mathcal{D}) p(\mathbf{f}_{0:T}^* | \mathcal{D}) d\mathbf{f}_{0:T}^*. \end{aligned} \quad (15)$$

Based on the GP regression (12), the distribution of motion function can be written as,

$$p(\mathbf{f}_{0:T}^* | \mathcal{D}) = \mathcal{N}(\mathbf{f}_{0:T}^* | \bar{\mathbf{f}}^*(\mathbf{x}_{0:T}^*), \mathbb{V}_{\mathbf{f}^*}(\mathbf{x}_{0:T}^*)), \quad (16)$$

where $\bar{\mathbf{f}}^*(\mathbf{x}_{0:T}^*) = [\bar{\mathbf{f}}_0^*, \dots, \bar{\mathbf{f}}_T^*]$ and $\mathbb{V}_{\mathbf{f}^*}(\mathbf{x}_{0:T}^*) = [\mathbb{V}_{\mathbf{f}_0^*}, \dots, \mathbb{V}_{\mathbf{f}_T^*}]$, where $\bar{\mathbf{f}}_t^*$ and $\mathbb{V}_{\mathbf{f}_t^*}$ can be calculated by (13) and (14) respectively. We have that the test input \mathbf{x}_t^* is independent from the training data set \mathcal{X} but relates to the

motion function \mathbf{f}_{t-1}^* and the state \mathbf{x}_{t-1}^* in the previous time slot, thus yielding,

$$\begin{aligned} p(\mathbf{x}_{0:T}^* | \mathbf{f}_{0:T}^*, \mathcal{D}) &= p(\mathbf{x}_{0:T}^* | \mathbf{f}_{0:T}^*), \\ &= p(\mathbf{x}_0^*) \prod_{t=1}^T p(\mathbf{x}_t^* | \mathbf{f}_{t-1}^*, \mathbf{x}_{t-1}^*), \\ &= p(\mathbf{x}_0^*) \prod_{t=1}^T \mathcal{N}(\mathbf{x}_t^* | \mathbf{f}_{t-1}^*, \mathbf{Q}). \end{aligned} \quad (17)$$

For the tracking problem, the Bayesian filter is implemented by realising (15) in an iterative way as shown in Fig. 1.

$$p(\mathbf{x}_t^* | \mathcal{D}) = \int p(\mathbf{x}_t^* | \mathbf{f}_{t-1}^*) p(\mathbf{f}_{t-1}^* | \mathcal{D}) d\mathbf{f}_{t-1}^*. \quad (18)$$

As we mainly focus on the four variables of the target states in this paper, i.e., $\mathbf{x}_t^* = [\xi_t^*, \dot{\xi}_t^*, \eta_t^*, \dot{\eta}_t^*]^T$, the motion function \mathbf{f}_{t-1}^* is decomposed into $\mathbf{f}_{t-1}^* = [f_{\xi,t-1}^*, f_{\dot{\xi},t-1}^*, f_{\eta,t-1}^*, f_{\dot{\eta},t-1}^*]$ and is learned and modelled as four Gaussian processes jointly, as shown in Fig. 2. Therefore, the prediction distribution in (18) can be decomposed as:

$$\begin{aligned} p(\mathbf{x}_t^* | \mathcal{D}) &= \prod_{s_t = \{\xi_t^*, \dot{\xi}_t^*, \eta_t^*, \dot{\eta}_t^*\}} \int p(s_t | f_{s,t-1}^*) p(f_{s,t-1}^* | \mathcal{D}) df_{s,t-1}^*. \end{aligned} \quad (19)$$

Here,

$$p(s_t | f_{s,t-1}^*) = \mathcal{N}(s_t | f_{s,t-1}^*, q), \quad (20)$$

$$\begin{aligned} p(f_{s,t-1}^* | \mathcal{D}) &= \mathcal{N}(f_{s,t-1}^* | \bar{f}_{s,t-1}^*(\mathbf{x}_{t-1}^*), \mathbb{V}_{f_{s,t-1}^*}(\mathbf{x}_{t-1}^*)), \\ &\sim \mathcal{N}(\bar{f}_{s,t-1}^*, \mathbb{V}[f_{s,t-1}^*]), \end{aligned} \quad (21)$$

where the mean and covariance function can be calculated as (13)-(14). By substituting (20) and (21) into (19), we obtain

$$\begin{aligned} p(\mathbf{x}_t^* | \mathcal{D}) &= \prod_{s_t = \{\xi_t^*, \dot{\xi}_t^*, \eta_t^*, \dot{\eta}_t^*\}} \mathcal{N}(s_t | \bar{f}_{s,t-1}^*, \mathbb{V}[f_{s,t-1}^*] + q). \end{aligned} \quad (22)$$

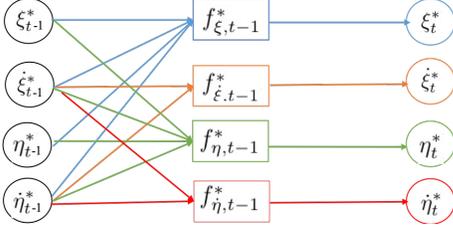


Fig. 2: Joint GPs for state prediction.

B. Data Test based on Particle Filter

In order to estimate the states for the test data iteratively, a PF is used to estimate the posterior distribution over the state $p(\mathbf{x}_t^* | \mathbf{z}_{1:T}^*, \mathcal{D})$ for each test time $t = 1, \dots, T$ [12] [13]. Based on the PF, the posterior probability can be approximated by the particles with corresponding weights as,

$$\begin{aligned} p(\mathbf{x}_t^* | \mathbf{z}_t^*, \mathcal{D}) &= \frac{p(\mathbf{z}_t^* | \mathbf{x}_t^*) p(\mathbf{x}_t^* | \mathcal{D})}{\int p(\mathbf{z}_t^* | \mathbf{x}_t^*) p(\mathbf{x}_t^* | \mathcal{D}) d\mathbf{x}_t^*} \\ &\approx \sum_{m=1}^M w_{m,t} \delta(\mathbf{x}_{m,t}^\dagger - \mathbf{x}_t^*), \\ &\approx \sum_{m=1}^M w_{m,t} \delta(\xi_{m,t}^\dagger - \xi_t^*) \delta(\dot{\xi}_{m,t}^\dagger - \dot{\xi}_t^*) \\ &\quad \times \delta(\eta_{m,t}^\dagger - \eta_t^*) \delta(\dot{\eta}_{m,t}^\dagger - \dot{\eta}_t^*). \end{aligned} \quad (23)$$

Here, m is the index of the particles, and $m = 1, \dots, M$. $\mathbf{x}_{m,t}^\dagger = [\xi_{m,t}^\dagger, \dot{\xi}_{m,t}^\dagger, \eta_{m,t}^\dagger, \dot{\eta}_{m,t}^\dagger]^T$ represents the particles and $w_{m,t}$ is the corresponding weight. The particles are drawn from the importance density which is chosen to be the prior probability [14] derived in (22), and weights are then calculated based on the likelihood function and observed measurement as (28). The pseudo-code of the proposed GP-PF based filter for tracking is illustrated in Algorithm 1.

IV. SIMULATION RESULTS

We first evaluate the derived distribution of the motion function $\mathbf{f}_{1:T}^*$ based on GP with a synthetic data-set. Then, the tracking performance is evaluated and compared with different existing tracking methods. The data for both training and test processes are synthetic, i.e., the generated target states evolve following motion models which switch between the CV or CT models as specified by the transition probability matrix:

$$\Gamma = \begin{bmatrix} p_{CV \rightarrow CV} & p_{CV \rightarrow CT} \\ p_{CT \rightarrow CV} & p_{CT \rightarrow CT} \end{bmatrix} = \begin{bmatrix} 0.6 & 0.4 \\ 0.4 & 0.6 \end{bmatrix}. \quad (30)$$

For all experiments, the following parameters are used: the model parameters are updated every 10 samples; the rotation rate of the CT model is distributed as $\omega_t \sim \mathcal{N}(0.1, 0.4)$; the process noise variance of both CT and CV models is $q = 1$. The length of the training data set is $N = 2000$.

A. Synthetic Data Set Evaluations

Before we evaluate the tracking performance of the proposed method, it is worth to verify the prediction

Algorithm 1

1) *Initialisation*: Draw M samples $\mathbf{x}_{m,0}^\dagger \sim p(\mathbf{x}_0)$, and the initial weights are set as $w_{m,0} = 1/M, m = 1, \dots, M$.

2) For $t = 1, \dots, T$, do:

For $m = 1, \dots, M$:

2.1) Draw particles $\mathbf{x}_{m,t}^\dagger = [\xi_{m,t}^\dagger, \dot{\xi}_{m,t}^\dagger, \eta_{m,t}^\dagger, \dot{\eta}_{m,t}^\dagger]^T$ using the distribution in (22),

$$\xi_{m,t}^\dagger \sim \mathcal{N}(\bar{f}_{\xi,t-1}^*, \mathbb{V}[f_{\xi,t-1}^*] + q), \quad (24)$$

$$\dot{\xi}_{m,t}^\dagger \sim \mathcal{N}(\bar{f}_{\dot{\xi},t-1}^*, \mathbb{V}[f_{\dot{\xi},t-1}^*] + q), \quad (25)$$

$$\eta_{m,t}^\dagger \sim \mathcal{N}(\bar{f}_{\eta,t-1}^*, \mathbb{V}[f_{\eta,t-1}^*] + q), \quad (26)$$

$$\dot{\eta}_{m,t}^\dagger \sim \mathcal{N}(\bar{f}_{\dot{\eta},t-1}^*, \mathbb{V}[f_{\dot{\eta},t-1}^*] + q), \quad (27)$$

2.2) Update the weight according to [14] as,

$$w_{m,t} \propto w_{m,t-1} \times \mathcal{N}(\mathbf{z}_t | h(\mathbf{x}_{m,t}^\dagger), R), \quad (28)$$

End For

2.3) The weight is then normalised and MAP estimation of the state is achieved according to,

$$[\hat{\mathbf{x}}_t^*]^{\text{MAP}} = \arg \max_{\mathbf{x}_t^*} p(\mathbf{x}_t^* | \mathbf{z}_{1:T}^*, \mathbf{x}_{1:t-1}^*). \quad (29)$$

2.4) Resampling: Replace particles with negligible weight by new particles in proximity of higher weighted particles.

End For

performance of the trained GP models for positions and velocities in 2D plane. The root-mean-square error (RMSE) of the prediction for each state within time are shown in Fig. 3. From this figure, we can conclude that the motion model with mixed behaviours and random rotation can be learned and approximated accurately using non-parametric GP regression.

B. Tracking Performance

This experiment compares and analyses the performance of different tracking methods. The performance of the target tracking is evaluated using the RMSE of the target positions. The proposed algorithm is compared with two other methods: The traditional PF with full information and the interacting multiple model-particle filter (IMM-PF) method with partial prior information. Specifically, for the traditional PF method, the dynamic motion mode in real-time and the turn rate for CT model ω_t are fully-known when operating the filter. This gives us measure of the optimum tracking performance. While for the IMM-PF method, the statistical distribution of the turn rate, i.e., $\omega_t \sim \mathcal{N}(\mu_\omega, \Sigma_\omega)$, are known as a *prior* information. The motion model is a mixture by CV and CT, therefore, two different multiple model (MM) set are used. The first denoted as IMM-PF1 is set to be $\{\text{CV}, \text{CT}_{\omega=\mu_\omega}\}$. The second denoted as IMM-PF2 is designed based on minimum distribution-mismatch design rule and

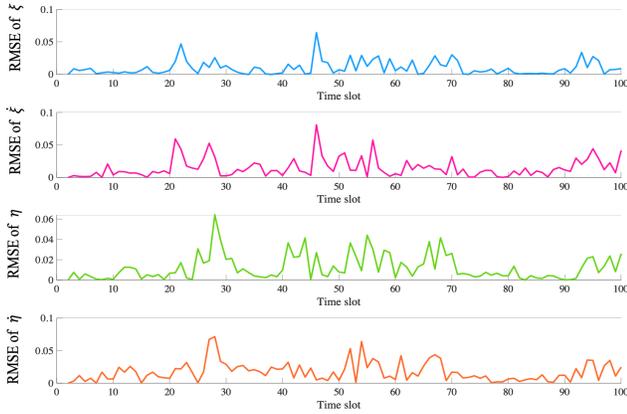


Fig. 3: Prediction performance of GP models: RMSE of the prediction for each state.

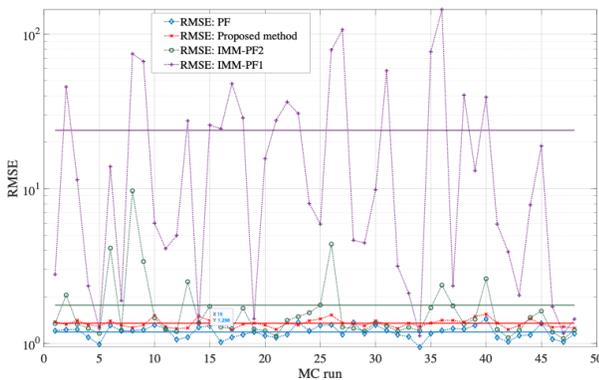


Fig. 4: Comparison of RMSE tracking performance achieved by different methods, and the straight lines show the mean value of the RMSE.

denoted as $\{CV, CT_{\omega=\omega(1)}, \dots, CT_{\omega=\omega(r)}\}$, where Υ is the number of the equal intervals that the cumulative distribution function (CDF) of ω is divided into. The comparisons of tracking performance achieved by the proposed GP-PF based method with others are shown in Fig. 4 and Fig. 5. In Fig. 4, $r = 1$, RMSE performance comparison is evaluated by different Monte Carlo realisations, i.e., different random target trajectories. In Fig. 5, RMSE performance with different measurement process noise variance is evaluated, where $r = \{0.1, 1, 2, 5, 10, 20\}$. From these two figures, we see that the proposed method outperforms the IMM-PF methods. Furthermore, the proposed method shows better robustness for different random trajectories of the target. Compared with the traditional PF with full information, the performance of GP-PF is very good considering it has no prior knowledge of the turn rate.

V. CONCLUSION

In this study, we propose a new method for single target tracking for mixed and uncertain motion. The proposed method is based on Gaussian process regression. The experiments show that this method has clear advantages to IMM methods when dealing with model uncertainty, and shows great robustness for random target trajectories and

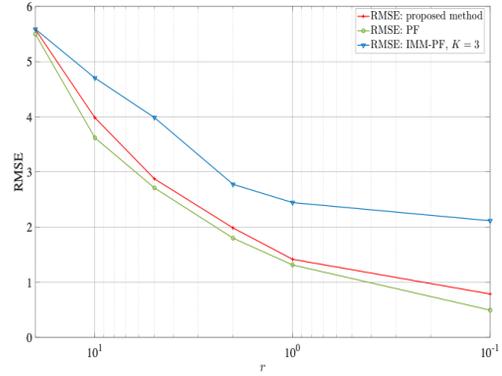


Fig. 5: RMSE tracking performance comparison within various measurement noise density, averaged over 100 Monte Carlo realisations.

system parameters. Further work will explore the proposed filter’s performance on real world data and the performance when dealing with the multi-target tracking problem.

REFERENCES

- [1] Z. X. Liu, Q. Q. Zhang, L. Q. Li, W. X. Xie, “Tracking multiple maneuvering targets using a sequential multiple target Bayes filter with jump Markov system models,” *Neurocomputing*, vol. 216, pp. 183–191, 2016.
- [2] X. Rong Li and V. P. Jilkov, “Survey of maneuvering target tracking. Part I. Dynamic models,” in *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1333-1364, Oct. 2003.
- [3] X. Rong Li and V. P. Jilkov, “Survey of maneuvering target tracking. Part V. Multiple-model methods,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 4, pp. 1255-1321, Oct. 2005.
- [4] J. L. Yang, L. Yang, Z. Y. Xiao, and J. J. Liu, “Adaptive parameter particle CBMeMber tracker for multiple maneuvering target tracking,” *IEURASIP J ADV SIG PR*, Vol. 2016, pp. 1-11, May 2016).
- [5] Z. X. Liu, D. H. Wu, W. X. Xie and L. Q. Li, “Tracking the Turn Maneuvering Target Using the Multi-Target Bayes Filter with an Adaptive Estimation of Turn Rate”, *Sensors*, vol. 17, no. 373, pp. 1-14, Feb. 2017).
- [6] J. Son, M. Baek, M. Cho, and B. Han, “Multi-object tracking with quadruplet convolutional neural networks,” in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2017, pp. 5620–5629.
- [7] S. Jung, I. Schlangen, and A. Charlish, “Sequential Monte Carlo Filtering with Long Short-Term Memory Prediction,” in *Proc. of 22nd Int. Conf. on Info. Fusion*, Jul. 2019.
- [8] N. Wahlström and E. Özkan, “Extended Target Tracking Using Gaussian Processes,” in *IEEE Trans. Signal Process.*, vol. 63, no. 16, pp. 4165-4178, Aug. 2015.
- [9] J. Wagberg, D. Zachariah, T. Schön, and P. Stoica, “Prediction performance after learning in Gaussian process regression,” in *Proc. 20th Int. Conf. Artif. Intell. Statist.*, vol. 54, pp. 1264–1272, Apr. 2017.
- [10] J. Ko and D. Fox, “GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models,” in *Proc. Int. Conf. Intell. Robots Syst.*, Nice, France, Sep. 2008, pp. 3471–3476.
- [11] Y. Zhao, C. Fritsche, G. Hendeby, F. Yin, T. Chen and F. Gunnarsson, “Cramér–Rao Bounds for Filtering Based on Gaussian Process State-Space Models,” in *IEEE Trans. Signal Process.*, vol. 67, no. 23, pp. 5936-5951, Dec. 2019.
- [12] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation,” *IEEE Proc. F—Radar Signal Process.*, vol. 140, no. 2, pp. 107–113, Apr. 1993.
- [13] A. Doucet and A. M. Johansen, “A tutorial on particle filtering and smoothing: Fifteen years later,” in *Nonlinear Filtering Handbook*. London, U.K.: Oxford Univ. Press, 2011, pp. 656–704.
- [14] M. S. Arulampalam, S. Maskell, N. Gordon and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” in *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174-188, Feb. 2002.

Narrowband Angle of Arrival Estimation Exploiting Graph Topology and Graph Signals

Ian K. Proudler, Vladimir Stankovic, and Stephan Weiss

Department of Electronic & Electrical Engineering, University of Strathclyde, Glasgow G1 1XW, Scotland

{ian.proudler,vladimir.stankovic,stephan.weiss}@strath.ac.uk

Abstract—Based on recent results of applying graph signal processing (GSP) to narrowband angle of arrival estimation for uniform linear arrays, we generalise the analysis to the case of arrays with elements placed arbitrarily in three dimensional space. We comment on the selection of the adjacency matrix, analyse how this new approach compares to the multiple signal classification (MUSIC) algorithm, and provide an efficient implementation. We demonstrate that the GSP approach can perform as well as the MUSIC algorithm in terms of accuracy and computational cost. Simulations indicate that the proposed GSP approach avoids the severe performance degradation with which MUSIC is associated at low signal to noise ratios.

I. INTRODUCTION

An important task in array processing is to estimate the angle of arrival (AoA) of any sources that illuminate the sensors. This is often based on the second order statistics of the array data. For example, the ‘classic’ multiple signal classification (MUSIC) algorithm [1] estimates the data covariance matrix; from its eigenvalue decomposition (EVD), MUSIC then derives a subspace decomposition that enables the AoA estimation. More than three decades after its publication, the MUSIC algorithm still remains subject to further investigations and extensions, see e.g., [2]–[4].

Although array processing algorithms such as MUSIC are not necessarily restricted to discrete data, many implementations operate on regularly spaced sampling in time and space. This however is not necessary: sampling in time can be non-uniform [5] particularly in the case of compressive sensing [6], and sampling in space can deviate from the linear uniformly spaced array (ULA), either deliberately or because of array deformation or calibration errors, to arbitrary array configurations [7], [8].

Graph signal processing (GSP) and graph spectral analysis allow the characterisation and efficient analysis of data that has been obtained on an irregularly sampled grid [10], and therefore provide an interesting fit to an array whose elements may be arbitrarily arranged in space. To date, two papers have attempted to harness GSP for array signal processing and AoA estimation in particular: [11] experimentally established a coarse correspondence of the graph Fourier transform (GFT) coefficients to the AoA for a single source in a ULA; also for a ULA, [12] have chosen the graph topology, and hence

the GFT, such that a MUSIC-like subspace projection can be exploited to estimate the AoA of a source. Specifically, the graph is constructed so that the steering vector for the source signal is an eigenvector (with unit eigenvalue) of the graph’s adjacency matrix. Hence, in the noise-free case, the array data will be orthogonal to the subspace defined by the remaining eigenvectors.

The ULA in [11], [12] defines a simple and straightforward adjacency between sensor elements, and derives a cyclic spatial graph structure by connecting each sensor node with its two nearest neighbours using unweighted [11] or weighted edges [12]. In [12], temporal samples acquired by each sensor are also modelled by a cyclic graph. Modelling the ULA by a cyclic graph leads to a sparse graph adjacency matrix [18] that contains only two non-zero elements in each row.

In this paper, we further develop the approach in [12] and particularly consider the case of non-uniform sampling, i.e. the case where arrays may be arbitrarily distributed in three dimensional space [9]. Since for an arbitrary array, spatial adjacency of sensors is not clearly defined, and sparsity of a graph’s adjacency matrix has no implications on the sparsity of the GFT, we replace the adjacency matrix used in [12] for a fully connected graph in both spatial and temporal domains. The resulting algorithm has a similar philosophy to that in [12] but differs in detail. We further develop a low complexity scheme, and highlight the performance difference between the GSP and MUSIC approaches. We show that the former, including its particularisation in [12], operates as a matched subspace detector [13].

First, we provide a brief review of the array signal model and MUSIC algorithm in Sec. II. Based on this, the graph topology and the GFT are motivated in Sec. III. Our GSP-based AoA estimation approach is then analysed in Sec. IV and compared to MUSIC via simulations in Sec. V.

II. SIGNAL MODEL AND ANGLE OF ARRIVAL ESTIMATION

A. Signal Model

We assume M sensors located arbitrarily in space, such that $\mathbf{r}_m \in \mathbb{R}^3$ is the Cartesian coordinate vector for the m th sensor, with $m = 1, \dots, M$. With respect to this array, a far-field source has a planar wavefront with normal vector $\mathbf{k}_{\varphi, \vartheta} = [\sin \vartheta \cos \varphi, \sin \vartheta \sin \varphi, \cos \vartheta]^T$ travelling across the array, whereby the source direction is given in spherical coordinates by azimuth φ and elevation ϑ . When normalised

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/S000631/1 and the MOD University Defence Research Collaboration in Signal Processing.

by the propagation speed c in the medium, $\mathbf{k}_{\varphi,\vartheta}/c$ is also known as the source's slowness vector.

The time delay which the wavefront experiences at the m th element relative to the origin is $t_m = \frac{1}{c}\mathbf{k}_{\varphi,\vartheta}^T \mathbf{r}_m$. If the unit length in the coordinate system is chosen as half the minimum wavelength, then $|\mathbf{k}_{\varphi,\vartheta}| = 1 = \lambda_{\min}/2 = c/(2f_{\max}) = cT_s$, where T_s is the sampling period assuming critical sampling in time. Therefore, $t_m = \tau_m T_s$ with

$$\tau_m = \mathbf{k}_{\varphi,\vartheta}^T \mathbf{r}_m, \quad (1)$$

which measures the wavefront's delay relative to the origin in samples.

If the array is illuminated by a narrowband source (labelled i) from direction $\{\varphi_i, \vartheta_i\}$, then its normalised angular frequency Ω_i turns the delay in (1) into a phase shift. The steering vector, which uniquely describes the phase shift for a source characterised by $\{\Omega_i, \varphi_i, \vartheta_i\}$, is

$$\mathbf{a}_{\Omega_i, \varphi_i, \vartheta_i} = \frac{1}{\sqrt{M}} [e^{j\Omega_i \tau_1}, \dots, e^{j\Omega_i \tau_M}]^H, \quad (2)$$

with $\tau_m, m = 1, \dots, M$, depending on $\{\varphi_i, \vartheta_i\}$ via (1). Also note that with the relative time lag between the m th and μ th sensor being

$$\tau_{m,\mu} = \mathbf{k}_{\varphi_i, \vartheta_i}^T (\mathbf{r}_\mu - \mathbf{r}_m), \quad (3)$$

the corresponding phase shift between the two sensors is reflected in the complex gain $e^{-j\Omega_i \tau_{m,\mu}}$.

Using the steering vector in (2), and assuming no loss in gain across the array, the M sensor signals $x_m[n] \in \mathbb{C}$, $m = 1, \dots, M$ with time index n , can be collected into a vector $\mathbf{x}[n] \in \mathbb{C}^M$,

$$\mathbf{x}[n] = \sum_{i=1}^I \mathbf{a}_{\Omega_i, \varphi_i, \vartheta_i} s_i[n] + \mathbf{v}[n], \quad (4)$$

where $s_i[n]$, $i = 1, \dots, I$, is the i th of I narrowband source signals illuminating the array, each with parameters $\{\Omega_i, \varphi_i, \vartheta_i\}$. The term $\mathbf{v}[n] \in \mathbb{C}^M$ in (4) represents spatially and temporally uncorrelated zero-mean Gaussian noise with $\mathcal{E}\{\mathbf{v}[n]\mathbf{v}^H[n-\tau]\} = \sigma_v^2 \mathbf{I} \delta[\tau]$, where $\mathcal{E}\{\cdot\}$ is the expectation operator, \mathbf{I} is the identity matrix, and $\delta[\tau]$ the impulse function.

B. Covariance Matrix and its Eigenvalue Decomposition

For independent narrowband sources with $\Omega_1 = \dots = \Omega_I$, the second order statistics of the data are captured by the spatial covariance matrix

$$\mathbf{R} = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^H[n]\} = \sum_{i=1}^I \sigma_i^2 \mathbf{a}_{\Omega_i, \varphi_i, \vartheta_i} \mathbf{a}_{\Omega_i, \varphi_i, \vartheta_i}^H + \sigma_v^2 \mathbf{I}, \quad (5)$$

where σ_i^2 is the power of the i th source. Thus, \mathbf{R} contains information on the angles of arrival $\{\varphi_i, \vartheta_i\}$ via the steering vectors $\mathbf{a}_{\Omega_i, \varphi_i, \vartheta_i}^H$, $i = 1, \dots, I$.

The covariance matrix can be factorised via the EVD $\mathbf{R} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^H = \sum_{m=1}^M \lambda_m \mathbf{q}_m \mathbf{q}_m^H$ with unitary $\mathbf{Q} = [\mathbf{q}_1, \dots, \mathbf{q}_M]$ containing the eigenvectors and $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_M\}$ the eigenvalues of \mathbf{R} . This factorisation is structurally similar to

(5), but the EVD generates orthogonal eigenvectors, a property that is not necessarily shared by the steering vectors in (5). Therefore, the eigenvectors do not provide direct access to the steering vectors and thus to the source parameters $\{\varphi_i, \vartheta_i\}$. However, if eigenvalues are arranged in descending order, the EVD provides a subspace partitioning

$$\mathbf{R} = [\mathbf{Q}_s \ \mathbf{Q}_n] \begin{bmatrix} \mathbf{\Lambda}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_n \end{bmatrix} \begin{bmatrix} \mathbf{Q}_s^H \\ \mathbf{Q}_n^H \end{bmatrix}, \quad (6)$$

where $\mathbf{\Lambda}_n = \sigma_v^2 \mathbf{I}$, and $\mathbf{\Lambda}_s \in \mathbb{R}^{I \times I}$ contains the dominant eigenvalues and all source steering vectors lie in the signal-plus-noise subspace spanned by their corresponding eigenvectors in $\mathbf{Q}_s \in \mathbb{C}^{M \times I}$. The remaining noise-only subspace is spanned by the columns of $\mathbf{Q}_n \in \mathbb{C}^{M \times (M-I)}$. Hence all the source steering vectors should be orthogonal to this noise-only subspace.

C. Multiple Signal Classification

In the multiple signal classification (MUSIC) algorithm [1], we estimate the space-time covariance matrix $\hat{\mathbf{R}}$ from the data. Due to finite sample size, such that $\mathbf{x}[n]$, $0 \leq n < N$, is only available for N snapshots, there will be a finite approximation error $(\mathbf{R} - \hat{\mathbf{R}})$ that depends on both \mathbf{R} and N [14]. As in (6), we perform a subspace decomposition

$$\hat{\mathbf{R}} = [\hat{\mathbf{Q}}_s \ \hat{\mathbf{Q}}_n] \begin{bmatrix} \hat{\mathbf{\Lambda}}_s & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{\Lambda}}_n \end{bmatrix} \begin{bmatrix} \hat{\mathbf{Q}}_s^H \\ \hat{\mathbf{Q}}_n^H \end{bmatrix}, \quad (7)$$

where the columns of $\hat{\mathbf{Q}}_s$ and $\hat{\mathbf{Q}}_n$ span the estimated signal-plus-noise and noise-only subspaces, respectively. The estimation error in $\hat{\mathbf{R}}$ will perturb these estimated subspaces w.r.t. the ground truth in (6) [15].

Given a steering vector $\mathbf{a}_{\Omega, \varphi, \vartheta}$ for a particular AoA $\{\varphi, \vartheta\}$, we then test how much it leaks into the estimated noise-only subspace $\hat{\mathbf{Q}}_n$, i.e.

$$\xi_{\text{MUSIC}}^{-1}(\varphi, \vartheta) = \|\hat{\mathbf{Q}}_n^H \mathbf{a}_{\Omega, \varphi, \vartheta}\|^2. \quad (8)$$

This is repeated for each AoA $\{\varphi, \vartheta\}$ of interest. The function $\xi_{\text{MUSIC}}^{-1}(\varphi, \vartheta)$ is zero if a steering vector lies entirely outside the estimated noise-only subspace. Typically, in MUSIC, instead of looking for zeros/dips of $\xi_{\text{MUSIC}}^{-1}(\varphi, \vartheta)$, we inspect $\xi_{\text{MUSIC}}(\varphi, \vartheta)$ for poles/spikes.

III. GRAPH TOPOLOGY AND GRAPH SIGNAL PROCESSING

In the following, akin to [11], [12], we operate with a single source, $I = 1$ (for brevity, let $\Omega_1 = \Omega$), and investigate how GSP could assist us with the AoA estimation problem.

A. Spatial Adjacency Matrix

In [12], the adjacency matrix takes into account the phase shift between neighbouring nodes in a ULA, which induces sparsity. For the arbitrary 3-d array considered here, defining neighbourhood or ordering of the sensors is less straightforward, and hence we work with the cross-correlation between nodes. This leads to a fully connected graph. Since all elements have the same gain towards the source signal, they have an

instantaneous cross-correlation of identical magnitude. Hence for a single narrowband source with frequency Ω , we have

$$\mathbf{A}_s = \frac{1}{M} \begin{bmatrix} 0 & e^{-j\Omega\tau_{1,2}} & \dots & e^{-j\Omega\tau_{1,M}} \\ e^{-j\Omega\tau_{2,1}} & 0 & \dots & e^{-j\Omega\tau_{2,M}} \\ \vdots & & \ddots & \vdots \\ e^{-j\Omega\tau_{M,1}} & \dots & e^{-j\Omega\tau_{M,(M-1)}} & 0 \end{bmatrix} \\ = \mathbf{a}_{\Omega,\varphi,\vartheta} \mathbf{a}_{\Omega,\varphi,\vartheta}^H - \frac{1}{M} \mathbf{I}, \quad (9)$$

where $\mathbf{a}_{\Omega,\varphi,\vartheta}$ is the steering vector defined in (2). The lag $\tau_{i,j}$ is the relative delay experienced by the source signal between the i th and j th sensor elements as defined in (3). The only sparsity of \mathbf{A}_s in (9) are the zero values on the diagonal. The much sparser arrangement in [12] has little consequence, since the EVD of \mathbf{A}_s generally will not reflect this sparsity.

B. Temporal Adjacency Matrix

If N snapshots of $\mathbf{x}[n]$, $n = 0, \dots, (N-1)$, are available, then this temporal window of the data can be embedded into the graph structure. Similar to the cyclic graph structure for the spatial ULA component, in [12] the temporal dimension is embedded as a cyclic graph for each sensor element, such that the overall adjacency matrix emerges as a Kronecker product between the spatial and the temporal adjacency matrices.

For a narrowband signal at frequency Ω , the snapshots will induce a phase progression which can be gathered in a vector $\mathbf{a}_\Omega \in \mathbb{C}^N$,

$$\mathbf{a}_\Omega = \frac{1}{\sqrt{N}} [1, e^{j\Omega}, \dots, e^{j(N-1)\Omega}]^H, \quad (10)$$

that is similar to a steering vector. Thus, we formulate

$$\mathbf{A}_t = \mathbf{a}_\Omega \mathbf{a}_\Omega^H - \frac{1}{N} \mathbf{I} \quad (11)$$

as an $N \times N$ adjacency matrix of the temporal graph associated with each sensor. With this and (9), the overall adjacency matrix of the graph connecting all sensors in both spatial and temporal domains becomes $\mathbf{A} = \mathbf{A}_s \otimes \mathbf{A}_t$, with \otimes denoting the Kronecker product (see Fig. 2 in [12] for the ULA example).

C. Graph Fourier Transform

Following [12], we use the graph Fourier transform (GFT) [18] to analyse signals defined on graphs. It is based on the EVD of the adjacency matrix $\mathbf{A} = \mathbf{A}_s \otimes \mathbf{A}_t$,

$$\mathbf{A} = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^H. \quad (12)$$

Note that with the EVDs $\mathbf{A}_s = \mathbf{Q}_s \mathbf{\Lambda}_s \mathbf{Q}_s^H$ and $\mathbf{A}_t = \mathbf{Q}_t \mathbf{\Lambda}_t \mathbf{Q}_t^H$, (12) simplifies [12] to $\mathbf{\Lambda} = \mathbf{\Lambda}_s \otimes \mathbf{\Lambda}_t$ and $\mathbf{Q} = \mathbf{Q}_s \otimes \mathbf{Q}_t$, whereby the latter represents the GFT matrix, containing the GFT basis in its columns [19].

For the EVD of \mathbf{A}_s , given (9) it is easy to show that

$$\mathbf{A}_s = [\mathbf{a}_{\Omega,\varphi,\vartheta} \ \mathbf{V}_s] \mathbf{\Lambda}_s \begin{bmatrix} \mathbf{a}_{\Omega,\varphi,\vartheta}^H \\ \mathbf{V}_s^H \end{bmatrix} \quad (13)$$

where $\mathbf{\Lambda}_s = \text{diag}\{\frac{M-1}{M}, -\frac{1}{M}, \dots, -\frac{1}{M}\}$. Due to the eigenvalue $\lambda_s = -\frac{1}{M}$ possessing an algebraic multiplicity of $(M-1)$, $\mathbf{V}_s \in \mathbb{C}^{M \times (M-1)}$ can be selected arbitrarily with orthogonal columns such that $\mathbf{V}_s^H \mathbf{V}_s = \mathbf{I}$ and $\mathbf{V}_s^H \mathbf{a}_{\Omega,\varphi,\vartheta} = \mathbf{0}$.

For the temporal component \mathbf{A}_t of the adjacency matrix, based on (11) and analogously to (13), we have that

$$\mathbf{A}_t = [\mathbf{a}_\Omega \ \mathbf{V}_t] \mathbf{\Lambda}_t \begin{bmatrix} \mathbf{a}_\Omega^H \\ \mathbf{V}_t^H \end{bmatrix}, \quad (14)$$

with $\mathbf{\Lambda}_t = \text{diag}\{\frac{N-1}{N}, -\frac{1}{N}, \dots, -\frac{1}{N}\}$ and $\mathbf{V}_t \in \mathbb{C}^{N \times (N-1)}$ arbitrary such that $\mathbf{V}_t^H \mathbf{V}_t = \mathbf{I}$ and $\mathbf{V}_t^H \mathbf{a}_\Omega = \mathbf{0}$.

This defines the GFT matrix \mathbf{Q} in terms of the spatial and temporal steering vectors $\mathbf{a}_{\Omega,\varphi,\vartheta}$ and \mathbf{a}_Ω , and their orthogonal complements \mathbf{V}_s and \mathbf{V}_t . It is not difficult to show, when the array configuration is reduced to a ULA, that this \mathbf{Q} is identical (up to ambiguities w.r.t. \mathbf{V}_s and \mathbf{V}_t) to the GFT associated with the sparse adjacency matrices selected in [12].

IV. MUSIC-LIKE SPECTRUM BASED ON GRAPHS AND GRAPH SIGNALS

A. Scanning Subspaces

Given a fully-connected graph defined by the adjacency matrix $\mathbf{A} = \mathbf{A}_s \otimes \mathbf{A}_t$, we define the graph signal as the concatenation of the snapshots $\mathbf{x}[n]$, i.e.,

$$\mathbf{x}^H = [\mathbf{x}^H[0], \mathbf{x}^H[1], \dots, \mathbf{x}^H[N-1]]. \quad (15)$$

In the noiseless case, this graph signal \mathbf{x} for a source defined by $\{\Omega, \varphi_1, \vartheta_1\}$ is aligned with the principal eigenvector of the GFT matrix \mathbf{Q} , i.e. $\mathbf{a}_{\Omega,\varphi_1,\vartheta_1} \otimes \mathbf{a}_\Omega$. Consequently, \mathbf{x} is orthogonal to the space spanned by the other eigenvectors. Similar to the MUSIC algorithm, for robustness when noise is present (or for multiple sources), instead of probing for the alignment of \mathbf{x} with this principal eigenvector, we check for the leakage of \mathbf{x} into the complement \mathbf{V} of the principal eigenvector, which is given by

$$\mathbf{V}_{\Omega,\varphi,\vartheta} = [\mathbf{a}_{\Omega,\varphi,\vartheta} \otimes \mathbf{V}_t, \ \mathbf{V}_s \otimes \mathbf{Q}_t]. \quad (16)$$

The columns of $\mathbf{V}_{\Omega,\varphi,\vartheta}$ therefore span the noise-only subspace of the graph signal. With this, we scan the graph signal for leakage into the noise-only subspace, i.e. we look for zeros/dips in

$$\xi_{\text{GSP}}^{-1}(\varphi, \vartheta) = \|\mathbf{V}_{\Omega,\varphi,\vartheta}^H \mathbf{x}\|_2^2, \quad (17)$$

with $\Omega = \Omega_1$ fixed, or akin to MUSIC, check $\xi_{\text{GSP}}(\varphi, \vartheta)$ for poles/spikes.

B. Comparison to MUSIC

The approach in (17) is similar to MUSIC in (8) in the sense that a noise-only subspace is scanned. The difference lies in which quantities are estimated, and which quantities are used to scan a range of parameters. In the GSP approach for $\xi_{\text{GSP}}(\varphi, \vartheta)$ in (17), the noisy data contributes the vector \mathbf{x} while we obtain $\mathbf{V}_{\Omega,\varphi,\vartheta}$ deterministically for a range of values $\{\Omega, \varphi, \vartheta\}$. In contrast, MUSIC estimates the noise-only subspace from the (noisy) sample covariance matrix, and scans this with the deterministic steering vector $\mathbf{a}_{\Omega,\varphi,\vartheta}$.

As a further difference, the GSP approach for $\xi_{\text{GSP}}(\varphi, \vartheta)$ utilises a vector space of dimension MN , and attempts to average out noise through the matrix multiplication in (17).

MUSIC averages across the temporal window of N snapshots to calculate an $M \times M$ sample space time covariance matrix, such that the dimension of the vector used for scanning is only M .

C. Numerical Efficiency

For both MUSIC and GSP approaches (8) and (17), we need to evaluate a norm of the form $\|\mathbf{V}^H \mathbf{y}\|_2^2$. Therefore, we have that

$$\begin{aligned} \|\mathbf{V}^H \mathbf{y}\|_2^2 &= \|\mathbf{V} \mathbf{V}^H \mathbf{y}\|_2^2 \\ &= \|(\mathbf{I} - \mathbf{V}^\perp \mathbf{V}^{\perp,H}) \mathbf{y}\|_2^2, \end{aligned} \quad (18)$$

where $[\mathbf{V}^\perp, \mathbf{V}]$ is unitary. The first step is easy to confirm via an SVD of \mathbf{V} , and the second step is based on subspace projections [16].

Identifying $\mathbf{V}^\perp = \hat{\mathbf{Q}}_s \in \mathbb{C}^{M \times 1}$ and $\mathbf{y} = \mathbf{a}_{\Omega, \varphi, \vartheta}$, MUSIC can thus be evaluated equivalently as

$$\xi_{\text{MUSIC}}^{-1}(\varphi, \vartheta) = \|\mathbf{a}_{\Omega, \varphi, \vartheta} - \hat{\mathbf{Q}}_s (\hat{\mathbf{Q}}_s^H \mathbf{a}_{\Omega, \varphi, \vartheta})\|_2^2. \quad (19)$$

Calculating (19) only requires $2M$ complex multiply accumulates (MACs) per tested angle pair $\{\varphi, \vartheta\}$. However to determine $\hat{\mathbf{Q}}_s$, MUSIC additionally requires the evaluation of $\hat{\mathbf{R}}$, which over N snapshots absorbs $\frac{1}{2}M(M+1)N$ MACs, and the calculation of its EVD at a cost of $\mathcal{O}(M^3)$.

For the GSP approach, in (18) we identify $\mathbf{V}^\perp = \mathbf{a}_{\Omega, \varphi, \vartheta} \otimes \mathbf{a}_\Omega$ and $\mathbf{y} = \mathbf{x}$, such that instead of (17), we can evaluate

$$\xi_{\text{GSP}}^{-1}(\varphi, \vartheta) = \|\mathbf{x} - (\mathbf{a}_{\Omega, \varphi, \vartheta} \otimes \mathbf{a}_\Omega)(\mathbf{a}_{\Omega, \varphi, \vartheta} \otimes \mathbf{a}_\Omega)^H \mathbf{x}\|_2^2. \quad (20)$$

This requires $2MN$ MACs for the evaluation of an angle pair $\{\varphi, \vartheta\}$, but no cost needs to be expended on covariance estimation or an EVD.

V. SIMULATIONS AND RESULTS

To compare the GSP approach and MUSIC, we assume an array of $M = 5$ sensors that lies within a cube of unit side length, where unity refers to critical spatial sampling at $\Omega = \pi$. The three spatial coordinates for each sensor are drawn—once for all simulations—from a uniform distribution $\mathcal{U}(0, 1)$, while the source signal at a narrowband frequency $\Omega_1 = \frac{\pi}{2}$ illuminates the array from an azimuth $\varphi_1 = 20^\circ$ and an elevation $\vartheta_1 = 70^\circ$. As in [12], we collect $N = 41$ snapshots of data at an adjustable signal-to-noise ratio (SNR).

Over a grid of azimuth and elevations angles $\{\varphi, \vartheta\}$, Figs. 1 to 4 show the evaluations of (19) and (20) for $\Omega = \Omega_1$ and two different SNR regimes. For the high SNR scenario of 30 dB in Figs. 1 and 2, both the GSP approach and MUSIC provide an accurate extremum at the source parameters $\{20^\circ, 70^\circ\}$, with a sharper lobe for MUSIC. For the lower SNR scenario of 0 dB in Figs. 3 and 4, the peaks are of lower intensity compared to the 30 dB scenario, with $\xi_{\text{MUSIC}}(\varphi, \vartheta)$ still providing a sharper peak compared to $\xi_{\text{GSP}}(\varphi, \vartheta)$.

To assess the accuracy of the proposed GSP-based method, we evaluate the estimated AoA using the cost functions in (19) and (20) as

$$\{\hat{\varphi}_1, \hat{\vartheta}_1\} = \arg \min_{\varphi, \vartheta} \xi_i^{-1}(\varphi, \vartheta) \quad (21)$$

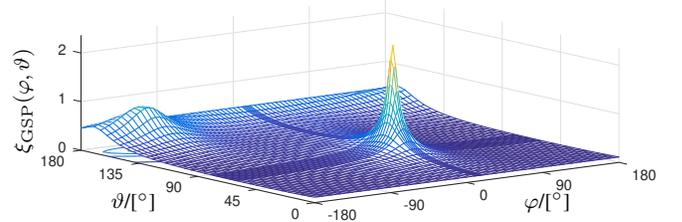


Fig. 1. GSP-based metric $\xi_{\text{GSP}}(\varphi, \vartheta)$ for $M = 5$ and $N = 41$ at 30 dB SNR.

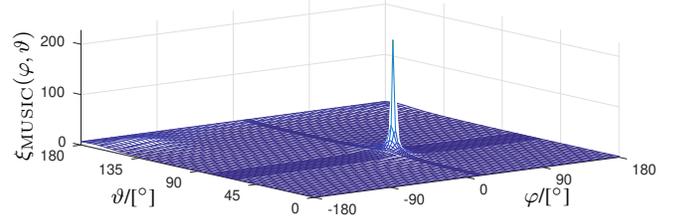


Fig. 2. MUSIC metric $\xi_{\text{MUSIC}}(\varphi, \vartheta)$ for $M = 5$ and $N = 41$ at 30 dB SNR.

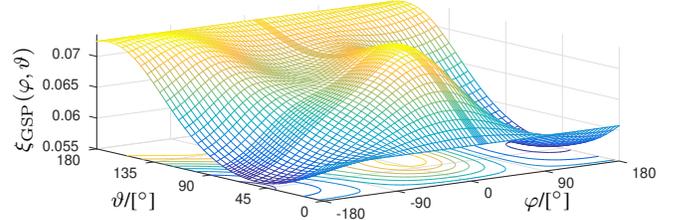


Fig. 3. GSP-based metric $\xi_{\text{GSP}}(\varphi, \vartheta)$ for $M = 5$ and $N = 41$ at 0 dB SNR.

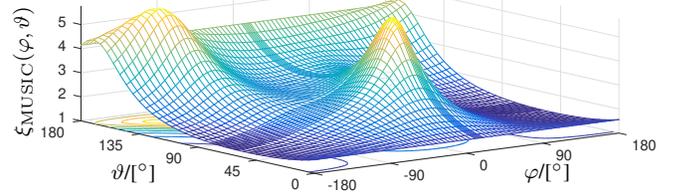


Fig. 4. MUSIC metric $\xi_{\text{MUSIC}}(\varphi, \vartheta)$ for $M = 5$ and $N = 41$ at 0 dB SNR.

with $i = \{\text{MUSIC}, \text{GSP}\}$. This non-linear optimisation problem is solved by the simplex algorithm [17], and performed over an ensemble of 10^4 realisation with different noise vectors $\mathbf{v}[n]$ in (4) for every SNR value under test. As a metric, we measure the angle between the direction $\mathbf{k}_{\varphi_1, \vartheta_1}$ of the source, and the direction associated with the estimated source parameters, $\mathbf{k}_{\hat{\varphi}_1, \hat{\vartheta}_1}$ via

$$\psi = \angle\{\mathbf{k}_{\varphi_1, \vartheta_1}, \mathbf{k}_{\hat{\varphi}_1, \hat{\vartheta}_1}\} = \arccos(\mathbf{k}_{\varphi_1, \vartheta_1}^H \mathbf{k}_{\hat{\varphi}_1, \hat{\vartheta}_1}) \quad (22)$$

i.e. the Hermitian angle between the two direction vectors.

Varying the SNR over the range from -20 dB to 20 dB, mean values of ψ for the GSP-based approach and for MUSIC are shown in Fig. 5. For high SNR, as expected from the surface of the metrics in Figs. 1 and 2, the mean performance is identical. To give an insight into the spread of the distribution of the ensemble of 10^4 experiments per SNR, Fig. 5 also contains the quartiles, which highlight that the MUSIC and GSP provide asymptotically identical accuracy as the SNR increases.

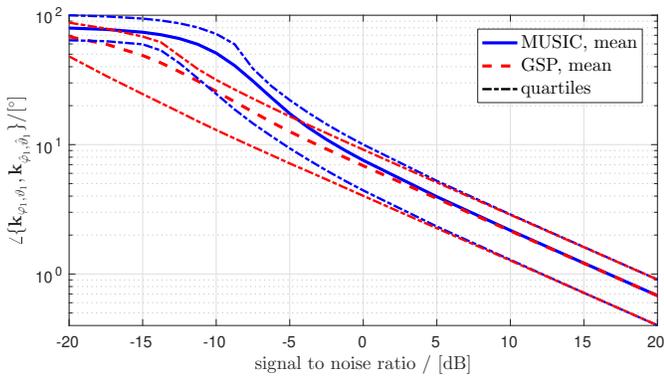


Fig. 5. Ensemble results for measuring the deviation of the estimated direction from the true one, ψ in (22), in dependency of the SNR of the data.

Towards low SNRs, the covariance matrix estimate $\hat{\mathbf{R}}$ becomes increasingly poor, leading to larger perturbation of the subspaces in (7) as compared to (6). For MUSIC, this subspace leakage is known to reach a cliff-edge type performance degradation as the estimation error increases [3]. This degradation can be observed in Fig. 5 for SNR region below 0 dB; similar results have been observed in e.g. [4]. Albeit that neither algorithms produce very accurate results, the GSP-based approach degrades gracefully and hence at lower SNR values outperforms MUSIC, since (20) essentially represents a matched filter using a beamformer [13], which is known to be optimal for the single user case.

For the above simulations, solving (21) under Matlab on an Intel CPU N3350 1.1GHz with 4GB RAM takes on average 11.9 ms CPU time for the GSP approach, while MUSIC takes 11.0 ms. Since for the selected parameters $MN \approx \mathcal{O}(M^3)$, the two computational costs from Sec. IV-C are in balance.

VI. CONCLUSIONS

In this paper we have investigated a GSP approach for angle of arrival estimation. Starting from [12], we have extended the approach from a ULA to an arbitrary sensor array, investigated the similarities and differences to the MUSIC algorithm, and provided a numerically efficient approach over [12] to the evaluation of both the GSP approach and MUSIC: while MUSIC estimates the noise-only subspace from data, and probes it with accurate steering vectors, in the GSP approach, the subspace is obtained from the adjacency matrix and therefore deterministically from the graph topology, which is probed by the potentially noisy graph signal.

For the selected simulation scenario and its parameters, both algorithms were of similar computational complexity. Simulations suggest that as SNR increases, the GSP approach asymptotically performs like MUSIC. At low SNR however, the GSP approach, implementing an optimum single user matched filter, avoids the fast degradation that is associated with MUSIC due to subspace leakage. This implies that the GSP approach, acting as a matched subspace detector, cannot provide any new gains for AoA estimation. This is perhaps unsurprising since GSP draws its advantages from additional

information [18], while the matched subspace detector already operates optimally on all available data.

Because of the arbitrary three-dimensional array configuration, we have not embedded any measure of physical adjacency of sensors in the graph topology, and both the graph and its adjacency matrix were selected non-sparse. While this does not impact on the GFT and hence the results in this paper, sparsity in the adjacency matrix can be important when seeking distributed implementations [20] and will therefore be worth investigating as a future step.

ACKNOWLEDGEMENT

We would like to thank Geert Leus of TU Delft, The Netherlands, for very helpful discussions and input.

REFERENCES

- [1] R.O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas & Propagation*, **34**(3):276–280, Mar. 1986.
- [2] S. Weiss, M. Alrmah, S. Lambotaran, J. McWhirter, and M. Kaveh, "Broadband angle of arrival estimation methods in a polynomial matrix decomposition framework," in *IEEE CAMSAP*, pp. 109–112, Dec. 2013.
- [3] M. Shaghghi and S.A. Vorobyov, "Subspace leakage analysis and improved DoA estimation with small sample size," *IEEE Trans. Signal Processing*, **63**(12):3251–3265, June 2015.
- [4] M.W. Morency, S.A. Vorobyov, and G. Leus, "Joint detection and localization of an unknown number of sources using the algebraic structure of the noise subspace," *IEEE Trans. Signal Processing*, **66**(17):4685–4700, Sep. 2018.
- [5] F. Marvasti, "Nonuniform sampling theorems for bandpass signals at or below the Nyquist density," *IEEE Trans. Signal Processing*, **44**(3):572–576, Mar. 1996.
- [6] D.L. Donoho, "Compressed sensing," *IEEE Trans. Information Theory*, **52**(4):1289–1306, Apr. 2006.
- [7] H.L. Van Trees, *Detection, Estimation and Modulation Theory: Optimum Array Processing*. New York: Wiley, 2002.
- [8] W. Liu and S. Weiss, *Wideband Beamforming — Concepts and Techniques*. Wiley, 2010.
- [9] A. Alzin, F. Coutts, J. Corr, S. Weiss, I.K. Proudler, and J.A. Chambers, "Adaptive broadband beamforming with arbitrary array geometry," in *IET/EURASIP ISP*, London, UK, Dec. 2015.
- [10] A. Ortega, P. Frossard, J. Kovacevic, J. M. F. Moura, and P. Vanderghenst, "Graph Signal Processing: Overview, Challenges, and Applications," *Proc. of the IEEE*, **106**(5):808–828, May 2018.
- [11] E. Alcantara, L. Atlas, and S. Abadi, "Graph signal smoothness for direction-of-arrival estimation of source targets using non-uniform line arrays," *J. Acoustical Society of America*, **146**(4):2886–2886, 2019.
- [12] L.A.S. Moreira, A.L.L. Ramos, M.L.R. de Campos, J.A. Apolinário, and F. G. Serrenho, "A graph signal processing approach to direction of arrival estimation," in *European Signal Processing Conference*, La Coruña, Spain, Sep. 2019.
- [13] L. L. Scharf and B. Friedlander, "Matched subspace detectors," *IEEE Trans. Signal Processing*, **42**(8):2146–2157, Aug. 1994.
- [14] C. Delaosa, J. Pestana, N.J. Goddard, S. Somasundaram, and S. Weiss, "Sample space-time covariance matrix estimation," in *ICASSP*, Brighton, UK, pp. 8033–8037, May 2019.
- [15] C. Delaosa, F.K. Coutts, J. Pestana, and S. Weiss, "Impact of space-time covariance estimation errors on a parahermitian matrix EVD," in *IEEE SAM*, Sheffield, UK, July 2018.
- [16] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: John Hopkins University Press, 1996.
- [17] J.A. Nelder and R. Mead, "A Simplex Method for Function Minimization," *The Computer Journal*, **7**(4):308–313, Jan. 1965.
- [18] D. Shuman, S. Narang, P. Frossard, A. Ortega, and P. Vanderghenst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, **30**(3):83–98, May 2013.
- [19] A. Sandryhaila and J.M.F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Processing*, **61**(7):1644–1656, Apr. 2013.
- [20] M. Coutino, E. Isufi, and G. Leus, "Advances in distributed graph filtering," *IEEE Trans. Signal Processing*, **67**(9):2320–2333, May 2019.

Information-Theoretic Compressive Measurement Design for Micro-Doppler Signatures

Fraser K. Coutts, John Thompson, and Bernard Mulgrew

Institute for Digital Communications, University of Edinburgh, Edinburgh, EH9 3FG, UK

fraser.coutts@ed.ac.uk

Abstract—In this work, we utilise gradient-ascent multi-objective optimisation within an information-theoretic compressive sensing framework to classify micro-Doppler (m-D) signatures in the presence of structured input noise. The proposed framework has the potential to simultaneously detect the class of a primary source exhibiting m-D features in its radar return and the class of a secondary, coincident source with its own m-D signature. We demonstrate through simulations with real radar return data that there is a configurable trade-off between the classification accuracies for the two sources and that, given a sufficient number of compressive measurements, the performance for the secondary source can be improved without substantially impacting the classification accuracy for the primary source.

I. INTRODUCTION

The main motion of an object with respect to a radar determines its predominant Doppler frequency shift; any secondary motions, such as the rotation of an aircraft’s rotor blades, contribute with features known as micro-Doppler (m-D) signatures [1], [2]. Such signatures appear superimposed on the object’s main Doppler contribution, and can be used to identify targets such as humans [3], animals [4], and helicopters [5]. Since m-D signatures are generally best perceived in the time-frequency domain [2], dimensionality reduction and feature extraction efforts typically revolve around spectrogram [2], [6], cepstrum [7], or bispectrum and bicepstrum [8] representations. In general, when utilising features extracted from such time-frequency representations, the most prominent m-D classification approaches involve support vector machines [6], Bayesian classifiers [9], Gaussian mixture models (GMMs) [7], [8], or neural networks [3].

Dimensionality reduction methods based on linear random projections — i.e., compressive sensing [10] (CS) — have gained significant attention recently; however, random projections may not be the best choice if we know the statistical properties of the underlying signal [11]. By employing an information-theoretic approach, one can design a linear projection such that the mutual information (MI) between the projected signal and the source signal or its class label is maximised [12]. Intuitively, as the MI increases, the recovery of the source signal or label information improves; indeed, the Bayes classification error is bounded by the MI [12]. Work in [13] utilises MI optimisation to design a linear projection that can trade-off signal recovery and classification accuracy for sources with arbitrary distributions.

The majority of CS approaches tackle scenarios in which a source is compressively sampled in the presence of measurement noise — i.e., noise added *after* the act of measurement.

The main signal model considered in this paper instead corresponds to an instance of CS with input — or “folded” [14] — noise, which some argue is a more realistic setup for CS [15]. In the context of m-D signatures in radar returns from moving objects, input noise might correspond to non-static clutter or a secondary source of m-D information. Existing CS-based m-D classification approaches [5] typically frame the problem as one of sparse signal recovery and do not account for input noise.

In general, a single Gaussian model does not provide a sufficiently accurate description of source signals [16]; instead, the distribution of a collection of signals can be approximated by a mixture of several Gaussians. In addition to being capable of modelling m-D features extracted from a time-frequency representation [7], GMMs have been shown to be effective [17] and in some cases superior to sparse signal models [16] in CS scenarios. Recent work [18] utilised MI maximisation and GMMs within a CS framework to optimise information throughput for a source signal in the presence of structured input noise.

Here, we generalise a key information-theoretic result of [18] to complex signal models and demonstrate its use in a subspace learning algorithm that can be applied to real radar data containing m-D signatures. In this context, we propose a novel system capable of jointly classifying the m-D signatures of a primary, always-present source and a secondary, fleeting source. By modelling both sources via GMMs and treating each as a source of structured input noise for the other, this system employs an iterative gradient-ascent approach to design a linear projection matrix capable of discriminating between the information throughput for each source. We demonstrate the various performance trade-offs that exist within the chosen signal model through simulations.

Below, Sec. II and Sec. III establish the signal model considered in this paper and a key theoretical result, respectively. Sec. IV provides an overview and practical demonstration of the proposed approach, and conclusions are drawn in Sec. V.

Notation: Straight bold lowercase and uppercase symbols denote vectors and matrices, respectively, and \mathbf{I}_n is an $n \times n$ identity matrix. Italicised uppercase letters such as \mathbf{Y} and \mathbf{C} denote random vectors and variables; their realisations are lowercase equivalents, such as \mathbf{y} or c . The superscripts $\{\cdot\}^H$ and $\{\cdot\}^*$ denote Hermitian transpose and complex conjugate operations. Operators $\mathbb{E}[\cdot]$, $|\cdot|$, and $\text{Arg}\{\cdot\}$ evaluate the expectation, magnitude, and phase. The complex gradient matrix is defined as in [19], such that $\nabla_{\Phi} F \triangleq \partial F / \partial \Phi^*$, where

$[\nabla_{\Phi} F]_{ij} = \partial F / \partial [\Phi^*]_{ij}$ and $[\cdot]_{ij}$ represents the (i, j) th element of a matrix. Similarly, $[\cdot]_i$ returns the i th element of a vector.

II. SIGNAL MODEL

We consider the following complex-valued signal model, which can be considered an input (“folding” [14]) noise model:

$$\mathbf{Y} = \Phi(\mathbf{X} + \mathbf{N}) + \mathbf{W}. \quad (1)$$

Following the compressive sampling protocol, we have measurements $\mathbf{Y} \in \mathbb{C}^m$ obtained from some desired signal $\mathbf{X} \in \mathbb{C}^n$ via a compressive measurement matrix $\Phi \in \mathbb{C}^{m \times n}$, with $m \ll n$. The signal $\mathbf{N} \in \mathbb{C}^n$, which is independent of \mathbf{X} and — when attempting to recover the features of \mathbf{X} — can be considered as input noise, is distributed according to a complex Gaussian mixture (GM); i.e.,

$$\mathbf{N} \sim \sum_{k=1}^K s_k \mathcal{CN}(\mathbf{n}; \boldsymbol{\mu}^{(k)}, \Gamma^{(k)}), \quad (2)$$

with $\boldsymbol{\mu}^{(k)} \in \mathbb{C}^n$, $\Gamma^{(k)} \in \mathbb{C}^{n \times n}$, and $\sum_{k=1}^K s_k = 1$. The vector $\mathbf{W} \sim \mathcal{CN}(\mathbf{w}; \mathbf{v}, \Lambda)$ represents additive complex Gaussian noise with mean $\mathbf{v} \in \mathbb{C}^m$ and covariance $\Lambda \in \mathbb{C}^{m \times m}$. The distribution of \mathbf{Y} given the observation of \mathbf{x} and the knowledge that \mathbf{n} has been generated according to the k th Gaussian component in (2) is therefore

$$p_{\mathbf{y}|\mathbf{x},k}(\mathbf{y}|\mathbf{x},k) = \mathcal{CN}(\mathbf{y}; \bar{\mathbf{y}}^{(k)}, \Sigma^{(k)}), \quad (3)$$

with mean vector $\bar{\mathbf{y}}^{(k)} = \Phi(\mathbf{x} + \boldsymbol{\mu}^{(k)}) + \mathbf{v}$ and covariance $\Sigma^{(k)} = \Phi \Gamma^{(k)} \Phi^H + \Lambda$. It is assumed that $\Sigma^{(k)}$ is invertible.

Suppose that an instance of \mathbf{X} is generated by one of J_x underlying classes, with each class, $c = 1 \dots J_x$, occurring with probability z_c . If the data distribution for class c is $p_{\mathbf{x}|c}(\mathbf{x}|c)$, the joint density is $p_{\mathbf{x},c}(\mathbf{x},c) = z_c p_{\mathbf{x}|c}(\mathbf{x}|c)$, and the global signal density is $p_{\mathbf{x}}(\mathbf{x}) = \sum_{c=1}^{J_x} z_c p_{\mathbf{x}|c}(\mathbf{x}|c)$. For now, we make no assumption on the form of $p_{\mathbf{x}|c}(\mathbf{x}|c)$.

III. KEY THEORETICAL RESULT

In [18], several terms relating to the gradient of MI for a real-valued version of the signal model in (1) are introduced. In this section, we generalise one of these results to our chosen complex signal model.

Theorem 1. With $I(C; \mathbf{Y})$ defined as the Shannon MI between the underlying class of \mathbf{X} — denoted by a discrete random variable C — and \mathbf{Y} , the gradient of $I(C; \mathbf{Y})$ with respect to Φ for the signal model of (1) is

$$\nabla_{\Phi} I(C; \mathbf{Y}) = \Lambda^{-1} \Phi \mathbf{E}_{z,c}. \quad (4)$$

Here, $\mathbf{Z} = \mathbf{X} + \mathbf{N}$ such that $C \rightarrow \mathbf{X} \rightarrow \mathbf{Z} \rightarrow \mathbf{Y}$ forms a Markov chain, $\mathbf{E}_{z,c} = \mathbb{E}_{c,y} [(\mathbb{E}[\mathbf{z}|c,\mathbf{y}] - \mathbb{E}[\mathbf{z}|\mathbf{y}])(\mathbb{E}[\mathbf{z}|c,\mathbf{y}] - \mathbb{E}[\mathbf{z}|\mathbf{y}])^H]$,

$$\begin{aligned} \mathbb{E}[\mathbf{z}|c,\mathbf{y}] &= \sum_{k=1}^K p_{k|c,y}(k|c,\mathbf{y}) \left(\Gamma^{(k),-1} + \Phi^H \Lambda^{-1} \Phi \right)^{-1} \\ &\times \left(\Gamma^{(k),-1} (\mathbb{E}[\mathbf{x}|c,\mathbf{y},k] + \boldsymbol{\mu}^{(k)}) + \Phi^H \Lambda^{-1} (\mathbf{y} - \mathbf{v}) \right), \end{aligned} \quad (5)$$

and $\mathbb{E}[\mathbf{z}|\mathbf{y}] = \sum_{c=1}^{J_x} p_{c|\mathbf{y}}(c|\mathbf{y}) \mathbb{E}[\mathbf{z}|c,\mathbf{y}]$.

Proof. The gradient of $I(C; \mathbf{Y})$ can be defined as $\nabla_{\Phi} I(C; \mathbf{Y}) = \nabla_{\Phi} h(\mathbf{Y}) - \nabla_{\Phi} h(\mathbf{Y}|C)$. In a proof that will be described in a subsequent publication, we formulate

expressions for the gradient of the entropy of \mathbf{Y} , $\nabla_{\Phi} h(\mathbf{Y})$, and the gradient of the conditional entropy, $\nabla_{\Phi} h(\mathbf{Y}|C)$. ■

IV. APPLYING THE SIGNAL MODEL TO RADAR DATA

A. Scenario

For a practical demonstration of the utility of the expression in (4), we consider a scenario in which the signal model of (1) is applied to radar data containing m-D signatures. GMMs have been shown to be effective in CS [16], [17] and for the modelling of m-D time-frequency features [7]. We therefore opt for both \mathbf{X} and \mathbf{N} to be represented by GMMs. Random vector \mathbf{X} represents some signal of interest that is guaranteed to be present in the signal model and is distributed as

$$\mathbf{X} \sim \sum_{c=1}^{J_x} z_c \sum_{o=1}^O \pi_{c,o} \mathcal{CN}(\mathbf{x}; \chi_{c,o}, \Omega_{c,o}). \quad (6)$$

That is, the probability distributions of classes $c = 1 \dots J_x$ of \mathbf{X} are each characterised by a GM with O components. Random vector \mathbf{N} represents a secondary source that may or may not be present in the system at the time of measurement. The secondary source — which has a probability of occurrence of $\gamma_{\tilde{\mathbf{n}}}$ — is distributed as

$$\tilde{\mathbf{N}} \sim \sum_{\ell=1}^{J_n} \tilde{r}_{\ell} \sum_{g=1}^G \tilde{v}_{\ell,g} \mathcal{CN}(\tilde{\mathbf{n}}; \tilde{\boldsymbol{\eta}}_{\ell,g}, \tilde{\boldsymbol{\Theta}}_{\ell,g}). \quad (7)$$

When the secondary source is not present, \mathbf{N} is distributed as $\mathcal{CN}(\mathbf{0}, \sigma \mathbf{I}_n)$ for some arbitrarily small σ ; i.e., the value of \mathbf{N} at this moment is close to zero. Thus, the distribution of \mathbf{N} is

$$\mathbf{N} \sim \sum_{\ell=1}^{J_n} r_{\ell} \sum_{g=1}^G v_{\ell,g} \mathcal{CN}(\mathbf{n}; \boldsymbol{\eta}_{\ell,g}, \boldsymbol{\Theta}_{\ell,g}), \quad (8)$$

with $J_n = J_{\tilde{\mathbf{n}}} + 1$ and

$$r_{\ell} = \begin{cases} \gamma_{\tilde{\mathbf{n}}} \tilde{r}_{\ell}, & \ell < J_n \\ 1 - \gamma_{\tilde{\mathbf{n}}}, & \ell = J_n \end{cases}, \quad v_{\ell,g} = \begin{cases} \tilde{v}_{\ell,g}, & \ell < J_n \\ 1/G, & \ell = J_n \end{cases},$$

$$\boldsymbol{\eta}_{\ell,g} = \begin{cases} \tilde{\boldsymbol{\eta}}_{\ell,g}, & \ell < J_n \\ \mathbf{0}, & \ell = J_n \end{cases}, \quad \boldsymbol{\Theta}_{\ell,g} = \begin{cases} \tilde{\boldsymbol{\Theta}}_{\ell,g}, & \ell < J_n \\ \sigma \mathbf{I}_n, & \ell = J_n \end{cases}.$$

Within the context of the signal model of (1), we primarily wish to design Φ such that we are able to accurately classify \mathbf{X} given \mathbf{Y} . Provided that there is no significant detrimental impact on the classification accuracy of \mathbf{X} , it might also be beneficial to use knowledge of the characteristics of \mathbf{N} to complement the design of Φ such that both \mathbf{X} and \mathbf{N} can be accurately classified given \mathbf{Y} . Designing Φ in this way would be useful if the operational circumstances of the system were to change, with \mathbf{N} (i.e., $\tilde{\mathbf{N}}$) becoming a signal of interest.

If we want to identify the Φ that maximises $I(C; \mathbf{Y})$, we can reparameterise \mathbf{N} according to (2) and use (4) in a gradient ascent scenario akin to that of [13], [18]. Analogously, if ℓ denotes an instance of a discrete random variable L , we could maximise $I(L; \mathbf{Y})$ by treating \mathbf{X} as a noise signal. Given the above scenario, the objective function to be maximised is

$$F(\Phi, \beta) = I(C; \mathbf{Y}) + \beta I(L; \mathbf{Y}), \quad (9)$$

where $\beta \in \mathbb{R}$ controls the relative importance of the information from the secondary source, $I(L; \mathbf{Y})$. This problem is reminiscent of the one in [13]; though in our case, we distinguish between two independent sources instead of balancing

the reconstruction and classification of a single source. For our purposes, we maintain $\beta \geq 0$. If β is negative as in [18], information relating to the class of the secondary source is actively penalised; this might be advantageous if the class of the secondary source is to be kept hidden from the receiver of the compressed measurement.

An iterative gradient ascent algorithm can attempt to identify the Φ that maximises $F(\Phi, \beta)$ by setting $\Phi \leftarrow \Phi + \delta \nabla_{\Phi} F(\Phi, \beta)$ at each iteration. The step size $\delta > 0$ controls the rate of change of Φ . To obtain $\nabla_{\Phi} I(C; \mathbf{Y})$, we reparameterise the probability distribution of \mathbf{N} according to (2) with

$$K = J_{\mathbf{n}} G, \quad s_k = r_{\ell'} \nu_{\ell', g'}, \quad \boldsymbol{\mu}^{(k)} = \boldsymbol{\eta}_{\ell', g'},$$

$$\boldsymbol{\Gamma}^{(k)} = \boldsymbol{\Theta}_{\ell', g'}, \quad \ell' = \lceil \frac{k}{G} \rceil, \quad g' = ((k-1) \bmod G) + 1,$$

where $\lceil \cdot \rceil$ is the ceiling function. We can then evaluate (4) via Monte Carlo (MC) integration and utilise the inference model detailed in Sec IV-B.

To obtain $\nabla_{\Phi} I(L; \mathbf{Y})$, we redefine the probability distribution of \mathbf{X} as a GM with $J_{\mathbf{x}} O$ components and use an appropriately modified version of the inference model below.

B. Inference Model

Under the chosen signal model in (1), the general Bayesian inference model for \mathbf{X} is a novel complex-valued extension of the model in [18]. It is constructed as follows:

$$p_{c|\mathbf{y}}(c|\mathbf{y}) = \frac{z_c p_{\mathbf{y}|c}(\mathbf{y}|c)}{p_{\mathbf{y}}(\mathbf{y})} = \frac{z_c p_{\mathbf{y}|c}(\mathbf{y}|c)}{\sum_{c'=1}^{J_{\mathbf{x}}} z_{c'} p_{\mathbf{y}|c}(\mathbf{y}|c')}, \quad (10)$$

$$p_{\mathbf{y}|c}(\mathbf{y}|c) = \sum_{o=1}^O \sum_{k=1}^K \pi_{c,o} s_k p_{\mathbf{y}|c,k,o}(\mathbf{y}|c, k, o), \quad (11)$$

$$p_{\mathbf{y}|c,k,o}(\mathbf{y}|c, k, o) = \mathcal{CN}(\mathbf{y}; \Phi(\boldsymbol{\chi}_{c,o} + \boldsymbol{\mu}^{(k)}) + \mathbf{v}, \boldsymbol{\Sigma}^{(k)} + \Phi \boldsymbol{\Omega}_{c,o} \Phi^H), \quad (12)$$

$$p_{k|c,\mathbf{y}}(k|c, \mathbf{y}) = \frac{s_k \sum_{o=1}^O \pi_{c,o} p_{\mathbf{y}|c,k,o}(\mathbf{y}|c, k, o)}{p_{\mathbf{y}|c}(\mathbf{y}|c)}, \quad (13)$$

$$p_{\mathbf{x}|c,\mathbf{y},k,o}(\mathbf{x}|c, \mathbf{y}, k, o) = \mathcal{CN}(\mathbf{x}; \tilde{\boldsymbol{\chi}}_{c,o}^{(k)}, \tilde{\boldsymbol{\Omega}}_{c,o}^{(k)}), \quad (14)$$

$$\tilde{\boldsymbol{\Omega}}_{c,o}^{(k)} = \left(\Phi^H \boldsymbol{\Sigma}^{(k), -1} \Phi + \boldsymbol{\Omega}_{c,o}^{-1} \right)^{-1}, \quad (15)$$

$$\tilde{\boldsymbol{\chi}}_{c,o}^{(k)} = \tilde{\boldsymbol{\Omega}}_{c,o}^{(k)} \left(\Phi^H \boldsymbol{\Sigma}^{(k), -1} (\mathbf{y} - \Phi \boldsymbol{\mu}^{(k)} - \mathbf{v}) + \boldsymbol{\Omega}_{c,o}^{-1} \boldsymbol{\chi}_{c,o} \right). \quad (16)$$

A closed-form expression can therefore be found for

$$\mathbb{E}[\mathbf{x}|c, \mathbf{y}, k] = \frac{\sum_{o=1}^O \pi_{c,o} p_{\mathbf{y}|c,k,o}(\mathbf{y}|c, k, o) \tilde{\boldsymbol{\chi}}_{c,o}^{(k)}}{\sum_{o'=1}^O \pi_{c,o'} p_{\mathbf{y}|c,k,o'}(\mathbf{y}|c, k, o')}. \quad (17)$$

Furthermore, the most likely class given knowledge of \mathbf{y} is

$$\hat{c} = \max_c p_{c|\mathbf{y}}(c|\mathbf{y}) = \max_c \{z_c p_{\mathbf{y}|c}(\mathbf{y}|c)\}. \quad (18)$$

A similar inference model can also be derived for \mathbf{N} .

C. Gradient-Based Numerical Solution

To attempt to identify the Φ that maximises $F(\Phi, \beta)$, we adapt the iterative gradient ascent approach of [13]. Note that $F(\Phi, \beta)$ is not, in general, a convex or concave function of Φ ; thus, finding a global-optimal solution is not guaranteed. During iterations, we constrain the energy of Φ such that

$\text{tr}\{\Phi \Phi^H\} = m$. The gradient ascent approach operates over a total of ρ iterations and can be summarised as follows:

- Draw $[\Phi]_{ij}$ from $\mathcal{CN}(0, 1)$; normalise s.t. $\text{tr}\{\Phi \Phi^H\} = m$.
- While the number of iterations is below ρ :
 - 1) Draw S samples of \mathbf{X} , \mathbf{N} , and \mathbf{W} ; evaluate (1);
 - 2) Compute (15) and (16); evaluate (13) and (17);
 - 3) Compute $\nabla_{\Phi} I(C; \mathbf{Y})$ via (4) using MC integration;
 - 4) Update the inference model for \mathbf{N} ; compute $\nabla_{\Phi} I(L; \mathbf{Y})$;
 - 5) $\Phi \leftarrow \Phi + \delta \nabla_{\Phi} F(\Phi, \beta)$; normalise s.t. $\text{tr}\{\Phi \Phi^H\} = m$.

D. Formatting the Radar Data

To establish the parameters of the GMMs prior to the design of Φ , instances of the primary and secondary sources, \mathbf{X} and $\tilde{\mathbf{N}}$, are obtained by applying the short-time Fourier transform (STFT) to continuous-wave (CW) radar return time series data captured in the presence of each source in isolation. For testing purposes, when either the primary source or both sources are present, instances of \mathbf{X} or $(\mathbf{X} + \tilde{\mathbf{N}})$ are obtained by applying the STFT to radar return time series data captured in the presence of the primary source or both sources, respectively.

A time series \mathbf{r} is the output of a radar receiver system and can contain the radar returns from the primary source, the secondary source, or both sources together. The time series has been reduced to baseband and downsampled such that the final sampling rate is adequate to capture the maximum m-D frequency shift without aliasing. Using the terminology of [4], for feature extraction purposes, the time series is split into R non-overlapping ‘frames’ of a predetermined length. Each frame \mathbf{r}_f , $f = 1 \dots R$, is then further segmented into B overlapping ‘bursts’. Each burst from frame f , $\mathbf{b}_{f,b}$, $b = 1 \dots B$ is then windowed to reduce spectral leakage and transformed to the frequency domain via the discrete Fourier transform to obtain $\mathbf{a}_{f,b}$.

When training the GMMs that describe the probability distributions of $\mathbf{X}, \tilde{\mathbf{N}} \in \mathbb{C}^n$ via the expectation-maximisation (EM) algorithm [20], instances of \mathbf{X} and $\tilde{\mathbf{N}}$ are obtained from $\mathbf{a}_{f,b}$ when \mathbf{r} contains radar returns from either source in isolation. Each vector $\mathbf{a}_{f,b} \in \mathbb{C}^n$ has elements defined as $[\mathbf{a}_{f,b}]_i = |[\mathbf{a}_{f,b}]_i| e^{j \text{Arg}[\mathbf{a}_{f,b}]_i}$, $i = 1 \dots n$. A disadvantage of this approach is that the probability distribution of the phase term, $e^{j \text{Arg}[\mathbf{a}_{f,b}]_i}$, is a circular uniform distribution, since it is unknown *a priori* at which point in its cycle an m-D signature is sampled. To successfully approximate the distributions of \mathbf{X} or $\tilde{\mathbf{N}}$ in this form, a significant number of Gaussian mixture components might be required.

In m-D signature analysis, the magnitude of the time-frequency representation is often used [2]; an approach that discards the phase could therefore generate instances of \mathbf{X} or $\tilde{\mathbf{N}}$ from examples of $|\mathbf{a}_{f,b}|$. However, it should be noted that if the phase is discarded during testing, when $\mathbf{a}_{f,b}$ contains features from both sources, the signal model of (1) does not necessarily apply, since $|\mathbf{x}_i + [\tilde{\mathbf{n}}]_i| \leq |\mathbf{x}_i| + |[\tilde{\mathbf{n}}]_i|$, with equality only when $\text{Arg}[\mathbf{x}_i] = \text{Arg}[\tilde{\mathbf{n}}]_i + 2\pi d$ for $i = 1 \dots n$ and $d \in \mathbb{Z}$.

In this paper, we choose to uphold the complex signal model in (1) and accept that a significant number of Gaussian mixture components might be required to encapsulate

TABLE I
APPROXIMATE FAN SPEEDS IN ROTATIONS PER SECOND.

Fan	Speed 1	Speed 2	Speed 3
1	5.68 rps	6.21 rps	6.78 rps
2	2.63 rps	4.10 rps	5.06 rps

the aforementioned phase ambiguity. Fortunately, results in Sec. IV-E demonstrate that a moderate number of components can still achieve a good level of classification accuracy.

E. Simulation Results for Real Data

Real radar returns from two fixed-location, three-bladed fans were acquired with a monostatic K-MC3 24 GHz radar system in CW mode at a sample rate of 22 kHz. The fans were positioned at a similar distance from the radar in an anechoic chamber and possessed three rotation speeds; these were estimated from the data and can be seen in Tab. I. Note that the fans had different blade designs and therefore produced m-D signatures with different characteristics. Three acquisitions of 30 seconds were made for each speed; these were downsampled to a rate of 5.5 kHz, which was found to adequately capture the maximum Doppler shift from each fan without aliasing. Both fans were positioned such that the plane in which the fan blades were rotating was approximately parallel to the radar line of sight. In our simulations, Fan 1 and Fan 2 are considered to be the primary and secondary sources, respectively.

For feature extraction purposes, we use a frame length of 700, as this captures a period of the m-D signature of the slowest fan speed. As has been noted elsewhere [7], as the GMM dimensionality increases, the parameter estimation performance in the GMM training stage degrades due to the finite training database size. We therefore limit the number of frequency coefficients generated in the STFT (and therefore the burst length and dimensionalities of $\mathbf{X}, \mathbf{N} \in \mathbb{C}^n$) to $n = 32$. To aid with the extraction of m-D signatures in various stages of their cycles, we use a burst overlap of 0.75; i.e., each burst shares 75% of the same time series data with its neighbours. Each burst is windowed using a 32-point Hamming window. Given these parameters, each frame produces $B = 84$ bursts.

Training data is obtained from 200 frames of radar return time series data recorded for each fan individually at each speed. For testing purposes, Fan 1 is either alone or in the presence of Fan 2 in the anechoic chamber; since each fan has three speeds, a total of 12 configurations are available. Data used for testing purposes only is obtained from 32 frames of time series data for each testing configuration.

We set the probability of occurrence of the second source signal, \tilde{N} , to be $\gamma_{\tilde{n}} = 0.75$ to match the rate of occurrence in the dataset used for testing. For the MC integration step at each iteration, $S = 500$ draws are used to evaluate $\nabla_{\Phi} F(\Phi, \beta)$. Note that if $\gamma_{\tilde{n}}$ were to be decreased, additional MC draws might be required at each iteration to adequately account for the characteristics of the now less-probable second source when designing Φ .

Using the EM algorithm [20] and the training examples for each fan speed, the distributions of \mathbf{X} and \tilde{N} are fitted such

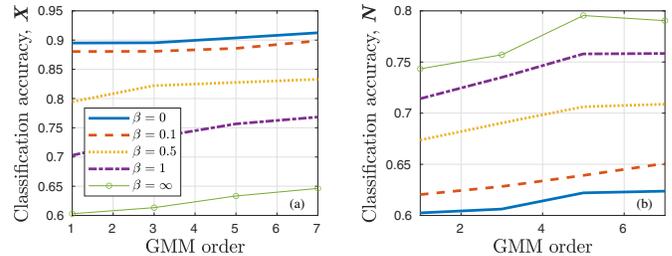


Fig. 1. Classification accuracy for sources (a) \mathbf{X} and (b) \mathbf{N} for various β and $G = O \in \{1, 3, 5, 7\}$ mixture components for each class with $m = 5$.

that each of the $J_x = J_{\tilde{n}} = 3$ classes had $O = G \in \{1, 3, 5, 7\}$ mixture components. For demonstration purposes, samples of measurement noise $\mathbf{W} \in \mathbb{C}^m$ are drawn according to $\mathbf{W} \sim \mathcal{CN}(\mathbf{w}; \mathbf{0}, 10^{-6} \mathbf{I}_m)$. Values of $m \in \{1, 3, 5, 7, 9\}$ and $\delta = 0.01$ are used. To illustrate the flexibility of the approach, β is varied in the range $\beta \in [0, \infty)$. Here, $\beta = 0$ results in the optimisation of $I(C; \mathbf{Y})$ only, while $\beta = 1$ attributes equal weight to both gradient terms. We use the notation $\beta = \infty$ to represent the optimisation of $I(L; \mathbf{Y})$ only; this corresponds to the case where $\nabla_{\Phi} I(C; \mathbf{Y})$ is ignored and $\beta = 1$. In the absence of the second source, the covariance of \mathbf{N} is determined by $\sigma = 10^{-6}$. At each iteration, the aforementioned test samples are used to compute the classification accuracy. While each burst can be classified with respect to the classes of \mathbf{X} via (18), if each burst $b = 1 \dots B$ generates an output $\mathbf{y}_1 \dots \mathbf{y}_B$, we can classify each frame of test data via

$$\hat{c}_f = \max_c \{z_c p_{y|c}(\mathbf{y}_1|c) \dots p_{y|c}(\mathbf{y}_B|c)\}, \quad (19)$$

under the assumption that \mathbf{y}_b , $b = 1 \dots B$, are mutually independent when conditioned on class $c = 1 \dots J_x$. We similarly classify each frame with respect to the classes of \mathbf{N} . Results are obtained after $\rho = 500$ iterations and averaged over 100 instances of the simulation scenario.

Fig. 1 demonstrates the impact of changing the number of mixture components for various choices of β with $m = 5$. These results confirm that the method is working as expected; i.e., increasing β reduces the throughput of information relating to \mathbf{X} while increasing knowledge of \mathbf{N} at the output. For this choice of m , the method cannot significantly increase the classification accuracy for \mathbf{N} without notably impacting the performance for \mathbf{X} . Fig. 1 also confirms that increasing the GMM order generally increases performance. However, given that performance for \mathbf{N} has in general only increased slightly — and, indeed, has decreased slightly in the case of $\beta = \infty$ — with the transition from order 5 to 7, it can be established that increasing the GMM order beyond this point is unlikely to be worth the additional computational cost.

The choice of number of measurements, m , enables a trade-off between performance and computational and memory requirements for the end-user. This trade-off is evidenced by Fig. 2, which confirms that increasing m generally increases classification accuracy for both \mathbf{X} and \mathbf{N} . It should be noted that for $m = 7$ and $m = 9$ and $\beta \in [0, 1]$, performance for \mathbf{X} is roughly the same, indicating that the maximum achievable classification accuracy for \mathbf{X} has been reached. The increase

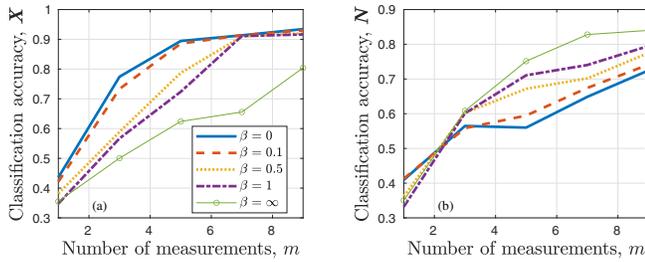


Fig. 2. Classification accuracy for sources (a) X and (b) N for various β and $m \in \{1, 3, 5, 7, 9\}$ measurements with $G = O = 1$.

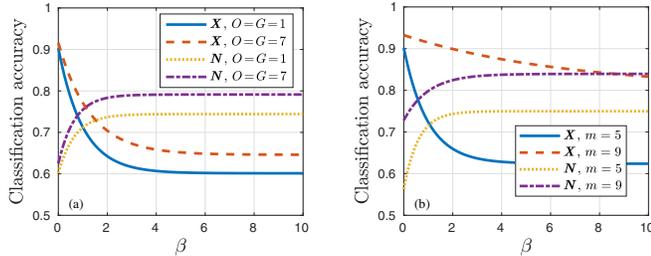


Fig. 3. Classification accuracy versus β for sources X and N with: (a) $G = O \in \{1, 7\}$ and $m = 5$; (b) $G = O = 1$ and $m \in \{5, 9\}$.

in m for these cases therefore only significantly increases performance for N . Clearly, the advantages offered by the proposed approach are most significant for these larger values of m , as there are sufficient degrees of freedom within the design process for Φ such that information from X and N can be jointly maximised.

To evaluate the impact of increasing β on classification accuracy for X and N , we have extrapolated the results of Fig. 1 and 2 by fitting a set of exponential functions to the data points. Fig. 3(a) illustrates the functions fitted to Fig. 1 for $G = O \in \{1, 7\}$ and Fig. 3(b) similarly shows the functions for Fig. 2 for $m \in \{5, 9\}$. Fig. 3(b) highlights that for $\beta \in [0, 1]$ and $m = 9$, a small sacrifice in performance for X translates to relatively large gains in performance for N . If the proposed framework were to be deployed in another application, a similar figure could aid the system designer when selecting system parameters.

V. CONCLUSIONS

In this work, we have utilised an information-theoretic expression for the gradient of MI to develop a novel CS framework capable of jointly classifying the m-D signatures of a primary, always-present source, and a secondary, fleeting source. Through simulations, we have demonstrated that when using the proposed approach, there is a configurable trade-off between the classification accuracies for the sources. Notably, we have shown that if the number of compressive measurements is sufficiently high, the classification accuracy for the secondary source can be increased without significantly impacting the performance for the primary source.

The proposed framework could be extended to applications in which the operational parameters are liable to change such that a secondary source of information — which might be ignored in normal circumstances — becomes more important.

ACKNOWLEDGEMENTS

This work was supported by the Engineering and Physical Sciences Research Council of the UK (EPSRC) Grant number EP/S000631/1 and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing. The authors would like to thank Prof. Mike Davies, Prof. Mathini Sellathurai, and Dr João Mota for their valuable input. Gratitude is also extended to Dr Domenico Gaglione, Dr Christos Ilioudis, and Dr Carmine Clemente for capturing the radar data.

REFERENCES

- [1] J. Martin and B. Mulgrew, "Analysis of the effects of blade pitch on the radar return signal from rotating aircraft blades," in *Int. Conf. on Radar*, 1992, pp. 446–449.
- [2] V. C. Chen, *The micro-Doppler effect in radar*. Artech House, 2019.
- [3] B. Vandersmissen et al., "Indoor person identification using a low-power FMCW radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 7, pp. 3941–3952, July 2018.
- [4] W. van Eeden, J. de Villiers, R. Berndt, W. Nel, and E. Blasch, "Micro-Doppler radar classification of humans and animals in an operational environment," *Expert Syst. with Applications*, vol. 102, pp. 1–11, 2018.
- [5] D. Gaglione, C. Clemente, F. K. Coutts, Gang Li, and J. J. Soraghan, "Model-based sparse recovery method for automatic classification of helicopters," in *IEEE Radar Conf.*, May 2015, pp. 1161–1165.
- [6] C. Clemente, L. Pallotta, A. De Maio, J. J. Soraghan, and A. Farina, "A novel algorithm for radar classification based on Doppler characteristics exploiting orthogonal pseudo-Zernike polynomials," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 1, pp. 417–430, Jan. 2015.
- [7] I. Bilik, J. Tabrikian, and A. Cohen, "GMM-based target classification for ground surveillance Doppler radar," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 1, pp. 267–278, Jan. 2006.
- [8] P. O. Molchanov, J. T. Astola, K. O. Egiazarian, and A. V. Totsky, "Classification of ground moving targets using bicepstrum-based features extracted from micro-Doppler radar signatures," *EURASIP J. Adv. Signal Process.*, vol. 2013, no. 1, p. 61, 2013.
- [9] S. Challa and G. W. Pulford, "Joint target tracking and classification using radar and ESM sensors," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, no. 3, pp. 1039–1055, July 2001.
- [10] E. Candes and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, March 2008.
- [11] J. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1395–1408, July 2009.
- [12] Z. Nenadic, "Information discriminant analysis: Feature extraction with an information-theoretic objective," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 8, pp. 1394–1407, Aug. 2007.
- [13] L. Wang, M. Chen, M. Rodrigues, D. Wilcox, R. Calderbank, L. Carin, "Information-theoretic compressive measurement design," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1150–1164, June 2017.
- [14] E. Arias-Castro and Y. C. Eldar, "Noise folding in compressed sensing," *IEEE Signal Process. Lett.*, vol. 18, no. 8, pp. 478–481, Aug. 2011.
- [15] K. Krishnamurthy, R. Willett, and M. Raginsky, "Target detection performance bounds in compressive imaging," *EURASIP J. Adv. Signal Process.*, vol. 2012, no. 205, Sep. 2012.
- [16] G. Yu and G. Sapiro, "Statistical compressed sensing of Gaussian mixture models," *IEEE Trans. on Signal Process.*, vol. 59, no. 12, pp. 5842–5858, Dec. 2011.
- [17] F. Renna, R. Calderbank, L. Carin, and M. Rodrigues, "Reconstruction of signals drawn from a Gaussian mixture via noisy compressive measurements," *IEEE Trans. Signal Process.*, vol. 62, no. 9, pp. 2265–2277, May 2014.
- [18] F. K. Coutts, J. Thompson, and B. Mulgrew, "Gradient of mutual information in linear vector Gaussian channels in the presence of input noise," in *European Signal Processing Conference*, Aug. 2020.
- [19] D. P. Palomar and S. Verdú, "Gradient of mutual information in linear vector Gaussian channels," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 141–154, Jan. 2006.
- [20] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

Identification of Radar Emitter Type with Recurrent Neural Networks

Sabine Apfeld*, Alexander Charlish*, Gerd Ascheid†

*Dept. Sensor Data and Information Fusion, Fraunhofer FKIE, Wachtberg, Germany

†Institute for Communication Technologies and Embedded Systems, RWTH Aachen University, Germany

Email: sabine.apfeld@fkie.fraunhofer.de

Abstract—In this paper, we present a method for the identification of different multifunction radar emitter types. It is based on Long Short-Term Memory recurrent neural networks and a previously published hierarchical modelling approach. This approach maps radar pulses to different levels of symbols which can be regarded as parts of a radar language. We evaluate our method with an example emitter that can make use of three different resource management techniques. The results show that it is possible to distinguish between radar types that mainly use the same emission parameters but differ in the resource management method.

I. INTRODUCTION

Identification of the radar emitter type from a stream of pulses is an important aspect in the field of electronic warfare (EW) as it provides information about the threat an emitter poses to a platform. Traditionally, emitter identification is performed by comparing the parameters stored in a database to the ones measured from the received signal. Incoming pulses are described by pulse descriptor words (PDWs), which include parameters like pulse repetition interval (PRI) or pulse repetition frequency (PRF), radio frequency (RF), and pulse width (PW). However, modern multifunction radars are agile such that simple pattern matching methods do not provide satisfactory identification accuracy any more. To recognise an agile radar, the sequential information provided in the PDW stream needs to be considered. In this paper, we therefore present an identification approach based on a special variant of a recurrent neural network, the Long Short-Term Memory (LSTM) [1], which is especially well suited to process sequential data that contains long-term dependencies.

Data from different radars is collected by listening to a specific frequency for a certain amount of time, which is called the (receiver) dwell time. Usually, several emitters are active simultaneously and hence, pulses from different radars are received at once. The first processing step therefore consists of sorting the pulses into sequences that should contain PDWs from one emitter only. This step is called deinterleaving, an introduction to the basic methods can be found in [2]. No information about the emitter identity is available after the deinterleaving step. Therefore, it is not possible to combine sequences received in different dwells into a longer sequence of PDWs of the same emitter to improve the identification accuracy. Effectively, data from different emitters is alternating in the input to the identification method. This is illustrated in

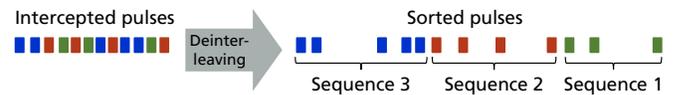


Fig. 1. The input to the identification method is a stream of sequences that consist of deinterleaved pulses from potentially different emitters.

Fig. 1. At this point, start and end of each sequence are known, but not to which emitter it belongs.

A. Related Work

First attempts to identify emitters with neural networks date back to 1990 [3]. Since then, radars have become more complex and neural networks more powerful. More recent approaches include the use of multilayer perceptrons [4], convolutional neural networks [5] and deep belief networks [6]. However, these methods need a fixed-length representation of their input and are therefore not efficiently applicable for processing of streaming data. The methods most similar to ours are those presented in [7]–[9] since they also employ recurrent neural networks (RNNs) for emitter identification. The RNN employed in the papers is either an LSTM or a Gated Recurrent Unit (GRU) [10]. Studies that evaluate the differences between gated RNN architectures like LSTM and GRU showed that none of them is fundamentally superior to the other [11], [12].

In contrast to our approach, [7]–[9] use a different encoding or modelling of the input data and different architectures of the networks. In [7], a sequence of n PDWs with several features (PRI, RF, PW, bandwidth (BW), amplitude (Amp), RF shift) is projected to a higher dimension and then averaged by a trainable layer before being processed by a GRU. In [8], the features of the pulses considered are PRI and PW. These are discretised with reference to a global maximum, embedded into a vector and input to a GRU network. In [9], the features PRI, RF and PW are used. They perform two types of input normalisation and concatenate the results, which is then input to parallel layers of LSTM cells that process different features separately.

B. Our Contributions

In this paper, we develop an LSTM based identification approach that is applicable to the hierarchical modelling presented in [13], [14] and briefly repeated in Section II.

Moreover, we investigate which level of the modelling is best suited for identifying the emitter type.

As described above, the input to the identification network consists of a stream of alternating sequences from different emitters (see Fig. 1). None of the papers [7]–[9] thoroughly investigates how the length of a consecutive sequence of data from the same emitter influences the identification accuracy. In this paper, we show how the identification accuracy depends on this length, which we call the sequence length. The impact of the sequence length on training and testing of the LSTMs is evaluated. We also demonstrate that LSTMs are able to distinguish between emitter types with mainly identical emission parameters but different resource management methods. This means that the main difference between the radars is in the frequency of certain parameters and agility of their emissions and not in the emissions themselves. Furthermore, we show that our identification approach is very robust with regards to missing and additional data in the input stream.

In Section II, we briefly repeat the hierarchical modelling approach presented in [14]. The example emitter used for the evaluation is described in Section III. In Section IV, the details on the implementation of the identification network are given and in Section V, the evaluation is presented. The conclusions are given in Section VI.

II. HIERARCHICAL MODELLING

The identification method presented in this paper is based on the hierarchical modelling approach of a radar language described in [13], [14]. Radar emissions are structured into different levels of modelling, which are in analogy to natural language called letters, syllables, words, commands, and functions (called tasks in [14]). These are characterised as follows:

- **Letters:** Letters correspond to the pulses the radar emits and can be characterised e.g. by the triple (PRF, RF, PW).
- **Syllables:** Syllables are common combinations of letters. They correspond to a radar burst.
- **Words:** Syllables are combined to form words. These correspond to a radar dwell.
- **Commands:** Commands describe words on a higher level of abstraction and can be interpreted as word types or classes.
- **Functions:** Functions are related to the purpose of the emission, e.g. searching or tracking targets.

Fig. 2 shows a schematic visualisation of the processing chain based on the hierarchical modelling. After the pulses have been intercepted and deinterleaved, the sequences of PDWs need to be mapped to the corresponding symbols (symbol extraction). The signal to noise ratio (SNR) of the received signal influences whether a symbol can be detected or not, but does not affect the processing after a successful detection. The symbol sequences are then input to the LSTM, which outputs an emitter ID per sequence.

III. EXAMPLE EMITTER

For evaluation of the proposed identification method, we use simulation data from an airborne multifunction radar.

TABLE I
NUMBER OF SYMBOLS USED BY EACH EMITTER TYPE [14].

Method	Letter	Syllable	Word	Command	Function
QoS	18	25 380	26 653	10	3
Rules-v1	13	103	21	2	3
Rules-v2	18	27 786	34 440	10	3

The same data was also used in [14]. The radar can make use of three different resource management methods, which determine how the radar resources, e.g. time, are allocated to different tasks it needs to fulfil. Furthermore, the waveform parameters are selected adaptively. The simplest resource management method uses a set of simple and fixed rules to chose the waveform parameters and allocate time to different tasks. A more sophisticated version of the fixed rules constitutes the second method. The most complex method makes use of a Quality of Service (QoS) approach. More details on the resource management methods and the parameters of the radar can be found in [14]–[16]. Since the resource management has a big impact on the structure and complexity of the emissions, the radar can be regarded as three different radars with the same language but a different grammar. In the following, the radars are abbreviated with rules-v1, rules-v2, and QoS. Table I lists the number of symbols (i.e. letters, syllables, words, commands, and functions) used by each radar type.

IV. IMPLEMENTATION

Traditionally, emitters are identified by comparing the waveform parameters of the intercepted signal with the entries in a database. However, the three example emitters presented above partly make use of the same parameters and therefore cannot be identified solely by the parameters used. To distinguish between the example emitters, the identification method needs to take the frequency and agility of the emissions into account, i.e. it needs to identify the resource management method. Hence, we train an LSTM network for identification.

A. Input Encoding

The input to the LSTM consists of radar symbols (see Fig. 2), which have to be encoded into a numerical representation. In [14], word embeddings are used to encode the symbols of each example emitter, whereby each emitter has its own dictionary. Since the identification network needs to know the symbols of all emitters, a global dictionary of word embeddings must be learnt. This is done in the same way as in [14] by using the reference implementation of `word2vec`¹ as a basis and the parameters shown in Table II. The global dictionary contains different vector representations of the symbols than the individual dictionaries of the emitters since the learnt representation is context-dependent and the radars differ in their behaviour.

¹https://github.com/tensorflow/tensorflow/blob/master/tensorflow/examples/tutorials/word2vec/word2vec_basic.py



Fig. 2. Schematic representation of the processing chain.

TABLE II
WORD2VEC PARAMETERS FOR THE VECTOR REPRESENTATIONS OF THE RADAR SYMBOLS.

Parameter	Letter	Syllable	Word	Command	Function
Embedding size	8	64	64	4	2
Batch size	128	128	128	128	128
Skip window	30	30	30	30	30
Num sampled	4	64	64	2	1

B. Data

For training and testing the same data as in [14] is used. It contains emissions from eight scenarios with 300 Monte Carlo runs per scenario and emitter type. Each scenario represents a different kind of situation, which include raids of hostile aircraft, fighters, jamming and missiles. Training is performed on six of the eight scenarios with 1440 runs per emitter type. The validation set consists of two scenarios with 240 runs per emitter type. For testing, all eight scenarios were used with 480 Monte Carlo runs per emitter type. For letters, a huge amount of data is available and hence, only two runs per scenario and emitter type were considered for training and testing.

C. Architecture & Training

To show how the identification accuracy depends on the sequence length during training, we develop different network types that are trained with different sequence lengths. To simulate the alternating sequences from different emitters (see Fig. 1), the input is changed to symbols from another emitter after the specified sequence length is reached. Per network, we independently optimise the number of LSTM layers and cells. One network type is trained with a sequence length of ten symbols, which we call LSTM₁₀. Another one is trained with random sequence lengths in the interval [1, 1400], we call it LSTM_{rand}. The third network type is trained with the complete scenarios and is called LSTM_{scen}. The complete scenarios consist of about 5 to 15 million letters each, those are mapped to about 7000 to 30 000 syllables and about 1400 to 6000 words, commands, and functions.

Per symbol level, i.e. letter, syllable, command, and function, an LSTM is trained. In total, we have 15 independent networks, one per symbol level and sequence length used during training. The parameters of these networks are shown in Table III. After the LSTM layer, a dropout of 50% is applied. It is followed by a dense layer with linear activation and a softmax layer to normalise the output.

The networks are trained with the Adam optimiser, a learning rate of 0.0002, and the cross entropy loss $H(P, Q)$ between

TABLE III
LSTM PARAMETERS FOR THE DIFFERENT MODELLING LEVELS AND NETWORK TYPES.

Type	Param.	Letter	Syllable	Word	Command	Func.
LSTM ₁₀	# layers	1	1	1	1	1
	# cells	16	4	4	4	4
LSTM _{rand}	# layers	1	1	1	1	1
	# cells	16	4	4	4	4
LSTM _{scen}	# layers	1	1	1	1	1
	# cells	4	4	8	16	16

the true probability distribution P and the learnt distribution Q

$$H(P, Q) = -\mathbb{E}_{x \sim P} \log Q(x). \quad (1)$$

During training, batches of 120 simulation runs (12 for letters) are fed in parallel to the networks and the LSTM state is kept between batches.

V. EXPERIMENTAL RESULTS

A. Evaluation Under Ideal Conditions

In a first step, the LSTMs are evaluated under ideal conditions without missing or additional symbols. In this setting, the LSTMs are compared to two simple identification strategies. One is random guessing with a probability of 33.33% for each emitter. The second one checks for each emitter if all symbols of the current input sequence are in its individual dictionary, i.e. it performs a dictionary lookup. If they are not, the emitter is assigned a probability of zero. If a symbol can be found in the dictionary of more than one or none of the emitters, the probability is equally distributed. Which emitter is the top 1 prediction is randomly chosen. This method corresponds to matching the waveform to a database.

Since the identification accuracy depends on the number of symbols intercepted in a row from one emitter, the LSTMs are tested with different sequence lengths in the set $S = \{1, 10, 50, 100, 200, 400, 600, 800, 1000, 1200, 1400\}$. With a sequence length of e.g. 10 symbols, a new emitter is randomly selected after ten symbols of the current emitter have been processed. The new emitter must not be equal to the current one. In our modelling, 1400 words correspond to about 1.5 min to 3 min, 200 words last about 35 s to 45 s. On average, a word consists of five to six syllables, therefore 1400 syllables correspond to about 25 s to 50 s, 200 to about 6 s to 9 s. These values are only intended for a rough reference since they strongly depend on the scenarios and the emitters.

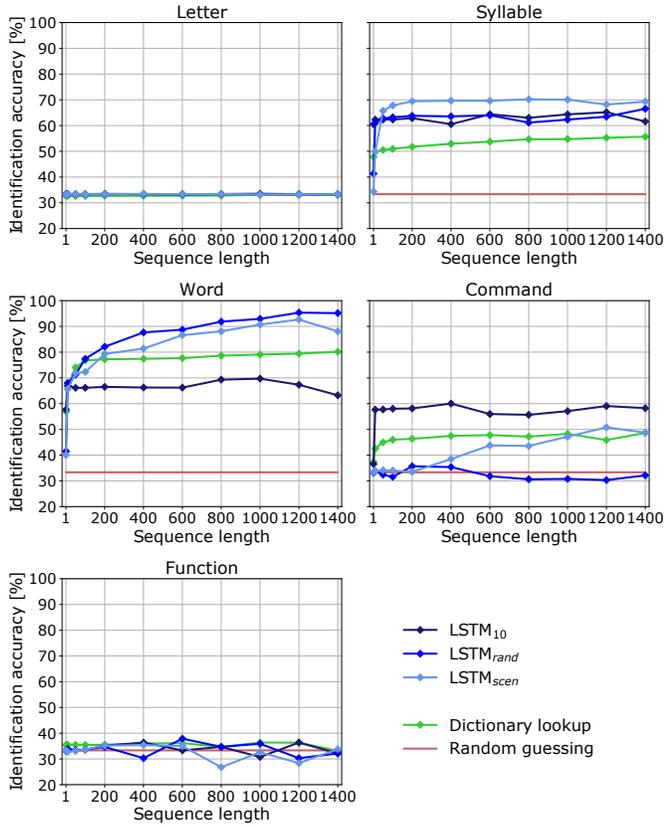


Fig. 3. Identification accuracies of the different networks on the test data of all emitters in comparison to simple strategies.

Fig. 3 shows the identification accuracies $acc(s)$ on the test data of all emitters, per symbol, identification method and sequence length $s \in S$, with

$$acc(s) = \frac{\# \text{ correctly classified seq. of length } s}{\# \text{ sequences of length } s} \cdot 100\%. \quad (2)$$

As can be observed, the emitters cannot be distinguished based on their letters, the results are equal to random guessing. Since the simulated radar is airborne, it integrates many pulse repetitions and therefore sequences of the same letter are very long (e.g. word $w_{26101507}$ consists of 16 384 repetitions of the same letter). When words are repeated, the sequences become even longer and are too long for an LSTM to learn.

For syllables and longer sequences, the dictionary lookup achieves much better results than guessing, but for a single syllable the accuracy is only 47.8%. The LSTM₁₀ provides a much higher identification accuracy than the other LSTMs when tested with only one syllable (60.47% vs. 34.37% for the LSTM_{scen} and 41.34% for the LSTM_{rand}). However, with increasing sequence length the other two LSTMs come close or are even better. The confusion matrices of the networks show that the rules-v1 radar cannot be recognised, regardless of the sequence length. All syllables of the rules-v1 radar are also contained in the dictionaries of the other radars.

TABLE IV
MEAN DIFFERENCE IN IDENTIFICATION ACCURACY $\overline{acc_{rel}}$ [%] FOR LSTM₁₀, LSTM_{rand}, AND LSTM_{scen} WITH MISSING AND ADDITIONAL SYLLABLES AND WORDS, AVERAGED OVER ALL SEQUENCE LENGTHS, WITH RESPECT TO THE RESULTS FOR IDEAL DATA.

	Rate	Syllables			Words		
		10	rand	scen	10	rand	scen
Missing	1%	-0.75	0.08	1.65	0.36	-0.07	0.04
	5%	-0.44	-0.98	1.88	2.02	-0.29	-0.22
	10%	-0.88	1.17	1.33	1.06	-0.26	-0.01
	20%	0.78	0.79	0.98	2.68	0.80	0.06
Missing in blocks	1%	-0.26	-0.48	1.47	0.91	-0.77	-0.75
	5%	-0.39	0.41	1.38	-1.03	-0.46	-0.19
	10%	-0.71	-0.22	1.21	-0.67	-0.01	0.11
	20%	-0.20	-0.15	0.73	1.03	0.63	-0.95
Additional	1%	-0.16	-0.90	1.30	-0.32	-7.44	-1.38
	5%	-2.19	-2.07	0.32	0.84	-15.95	-1.47
	10%	-2.96	-3.03	-0.33	-0.19	-18.07	-0.05
	20%	-5.90	-5.67	-3.12	-2.41	-19.17	-2.43
Additional in blocks	1%	-1.53	0.16	0.55	0.08	-4.35	-0.48
	5%	-1.82	-1.99	1.36	-0.50	-11.62	-1.37
	10%	-4.61	-3.96	-0.02	-0.57	-16.19	-0.76
	20%	-7.97	-6.27	-0.17	-2.79	-19.99	-2.99

For only one word, the LSTM₁₀ provides the best accuracy. However, the accuracy does not improve with increasing sequence length, since the network has not learnt to make use of longer sequences. Fig. 4 shows the confusion matrices of the LSTM_{rand} for words and different sequence lengths. The QoS radar is identified with a very high accuracy also with short sequences. As the two rule based approaches are more similar in their behaviour regarding words, longer sequences are needed to discriminate between them.

Using commands, the radars cannot be distinguished with satisfying accuracy. The LSTM₁₀ is able to identify the QoS radar with an accuracy of 77%, but the two rule based radars are always confused. The LSTM_{scen} identifies the rules-v1 radar with an accuracy of 50% with longer sequence lengths, but is almost never able to recognise the QoS radar.

For functions, the identification accuracy is basically identical to random guessing. Since the radars only use three different functions (search, confirm, track) and their behaviour is similar, they cannot be distinguished based on them.

B. Evaluation With Missing and Additional Symbols

In an actual application, the data is probably not ideal. Therefore, we also test the LSTMs with missing and additional symbols, which might occur due to errors made in the deinterleaving or pulses that could not be detected because of a low SNR or because the receiver is listening to a different RF. The additional symbols are randomly selected from the global dictionary, i.e. also symbols from different emitters are inserted. Two cases are considered for the evaluation. In the first one, single symbols are randomly removed or inserted. In the second case, symbols are removed or inserted in blocks of five. Since the results for letters, commands, and functions

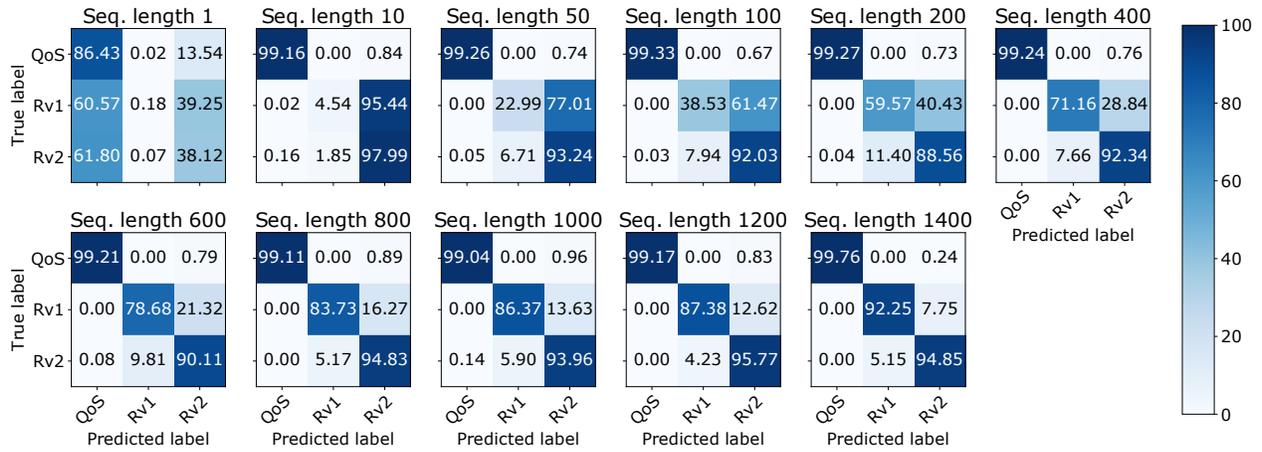


Fig. 4. Confusion matrices at different sequence lengths (testing) of the LSTM_{rand} for words.

are already unsatisfactory for ideal data, we only present the results for syllables and words. Table IV shows the accuracy \overline{acc}_{rel} averaged over all sequence lengths in the set S and relative to the results obtained with ideal data

$$\overline{acc}_{rel} = \frac{\overline{acc}_{corrupt} - \overline{acc}_{ideal}}{\overline{acc}_{ideal}} \cdot 100\%, \quad (3)$$

with

$$\overline{acc} = \frac{1}{S} \sum_{s \in S} acc(s). \quad (4)$$

The results show that all networks are in general very robust with respect to missing and additional symbols, the LSTM_{rand} with additional words being an exception. Furthermore, it is observed that additional symbols cause a bigger decrease in accuracy than missing ones, which have little impact. The difference between single additional symbols and additional symbols in blocks of five is small. This robustness is an advantage over the dictionary lookup. An additional symbol from a different emitter would cause the assigned probability of the correct emitter to be zero for the complete sequence and therefore result in a substantial accuracy decrease.

VI. CONCLUSION

In this paper, we proposed a method to distinguish between different radar emitter types based on Long Short-Term Memory networks and a hierarchical modelling approach of a radar language. We demonstrated that LSTMs are able to identify emitter types based on the frequency and agility of their emissions and are therefore capable of recognising the resource management method. Our evaluations show that the identification accuracy depends on the length of consecutive emissions received from the same radar and that longer sequences are needed to discriminate between similar emitter types. Furthermore, it could be seen that radar words, which correspond to radar dwells, are the modelling level best suited for identification. Moreover, we demonstrated that the LSTMs are in general very robust with respect to corrupted data.

REFERENCES

- [1] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [2] R. Wiley, *ELINT: The Interception and Analysis of Radar Signals*. Artech House, 2006.
- [3] G. B. Willson, "Radar classification using a neural network," *SPIE Applications of Artificial Neural Networks*, vol. 1294, pp. 200–210, 1990.
- [4] I. Jordanov and N. Petrov, "Intelligent Radar Signal Recognition and Classification," in *Recent Advances in Computational Intelligence in Defense and Security*, ser. Studies in Computational Intelligence, R. Abielmona, R. Falcon, N. Zincir-Heywood, and H. A. Abbass, Eds. Springer International Publishing, 2016, vol. 621.
- [5] L. Cain, J. Clark, E. Pauls, B. Ausdenmoore, R. Clouse, and T. Josue, "Convolutional Neural Networks for Radar Emitter Classification," *IEEE Computing and Communication Workshop and Conference*, 2018.
- [6] H. Li, W. Jing, and Y. Bai, "Radar Emitter Recognition Based on Deep Learning Architecture," *CIE International Conference on Radar*, 2016.
- [7] S. A. Shapero, A. B. Dill, and B. O. Odelowo, "Identifying Agile Waveforms with Neural Networks," in *International Conference on Information Fusion (FUSION)*, 2018.
- [8] Z.-M. Liu and P. S. Yu, "Classification, Denoising, and Deinterleaving of Pulse Streams With Recurrent Neural Networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, no. 4, 2019.
- [9] P. Notaro, M. Paschali, C. Hopke, D. Wittmann, and N. Navab, "Radar Emitter Classification with Attribute-specific Recurrent Neural Networks," *pre-print arXiv: 1911.07683*, 2019.
- [10] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *Conference on Empirical Methods in Natural Language Processing*, 2014.
- [11] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An Empirical Exploration of Recurrent Network Architectures," *International Conference on Machine Learning*, 2015.
- [12] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017.
- [13] N. A. Visnevski, V. Krishnamurthy, A. Wang, and S. Haykin, "Syntactic modeling and signal processing of multifunction radars: A stochastic context-free grammar approach," *Proceedings of the IEEE*, vol. 95, 2007.
- [14] S. Apfeld, A. Charlish, and G. Ascheid, "Modelling, Learning and Prediction of Complex Radar Emitter Behaviour," *IEEE International Conference on Machine Learning and Applications*, 2019.
- [15] A. Charlish and F. Katsilieris, "Array radar resource management," in *Novel Radar Techniques and Applications Volume 1: Real Aperture Array Radar, Imaging Radar, and Passive and Multistatic Radar*. Institution of Engineering and Technology, 2017, pp. 135–171.
- [16] A. Charlish and F. Hoffmann, "Cognitive radar management," in *Novel Radar Techniques and Applications Volume 2: Waveform Diversity and Cognitive Radar, and Target Tracking and Data Fusion*. Institution of Engineering and Technology, 2017, pp. 157–193.

FAST SURFACE DETECTION USING SINGLE-PHOTON DETECTION EVENTS

Abderrahim Halimi, Andrew Wallace, Gerald S. Buller, Stephen McLaughlin,

School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh U.K.

ABSTRACT

This paper presents a fast object detection algorithm for 3D single-photon Lidar data. Lidar imaging acquires time-of-flight (ToF) events in different spatial locations to build a 3D image of the observed objects. However, high ambient light or obscurants, might affect the reconstruction quality of the 3D scene. This paper proposes a solution by first detecting the pixels containing photons reflected from a object/surface, allowing a higher level processing of the data while only accounting for informative pixels. In contrast to histogram based approaches, the proposed algorithm operates on the detected photon events allowing a reduction in memory requirements and computational times. A Bayesian approach is considered leading to analytical estimates that can be computed efficiently. Results on simulated and real data highlight the benefit of the proposed approach when compared to a state-of-the-art algorithm based on histogram of counts.

Index Terms— 3D Lidar imaging, Bayesian approach, target detection, sparse photon regime, single-photon events.

1. INTRODUCTION

Single-photon 3D laser detection and ranging (Lidar) imaging has emerged as a candidate technology for a number of application areas including defence, automotive [1], and environmental sciences [2]. This imaging system builds a high-resolution 3D image of the observed objects by sending laser pulses and collecting the reflected photons from a surface while measuring their time-of-flight (ToF). The ToFs contain information about the system-target distance while the number of collected photons inform on the reflectivity of the observed scene. It is also common to pre-process the detected ToFs events into a histogram of counts and to apply different processing strategies on the resulting waveforms. However, this data representation is memory inefficient especially in the sparse photon regime, and requires an additional computational cost to convert photon events to histograms. This paper operates on the raw ToFs photon events to ensure an optimized exploitation of the available computational resources.

This work was supported by the UK Royal Academy of Engineering under the Research Fellowship Scheme (RF/201718/17128), EPSRC Grants EP/T00097X/1, EP/S000631/1, EP/S026428/1, EP/N003446/1, the Dasa project DSTLX1000147844 and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

Thanks to their good resolution and low sensitivity to noise, time-correlated single-photon counting (TCSPC) Lidar systems are currently used to perform long-range imaging [3] in addition to imaging through obscurants [4–7]. Several pixels are scanned in both cases, however, some pixels might only contain background counts due to ambient light, reflection from the observation environment (air, water, etc.) or dark events due to the detector noise. Therefore, several algorithms have been designed to detect pixel with useful photons, i.e., photons reflected from an object or a surface. Such approaches include the Markov chain Monte-Carlo (MCMC) method proposed in [8], which is time consuming due to the use of a sampling MCMC strategy. Two fast algorithms were recently proposed in [9, 10], which use a Bayesian formulation to output a per-pixel probability of target presence. These algorithms showed state-of-the-art performance, however, they operated on a histogram of counts which is not an optimal data representation given limited computing resources.

This paper proposes a new fast algorithm for per-pixel object detection. We adopt a Bayesian approach operating on the raw ToFs data and defining as parameters the target depth (if present), a signal-to-background related parameter and a binary parameter indicating the presence or absence of a target. A probability mixture model is considered for the likelihood, while appropriate prior distributions are chosen for each model parameters to express their known properties. The resulting model selection problem is then solved by marginalizing the depth and SBR parameters, leading to analytical expressions for the probability of detecting a target. The resulting analytical expressions are however combinatorial, and an approximation is introduced to ensure fast computations. The proposed approach is validated on simulated and real Lidar data showing good performance when compared to the algorithm [9] in terms of computational cost and detection performance.

The paper is structured as follows. Section 2 introduces the observation model of the detected photon events. Section 3 presents the proposed Bayesian model for target detection. The computation of the marginal probabilities are described in Section 4. Results and conclusions are finally reported in Sections 5 and 6.

2. OBSERVATION MODEL

Single-photon Lidar systems generally emit laser pulses and detect the reflected photons, together with their ToFs, from the target for each spatial location/pixel. The detected photons and measured ToFs provide useful information regarding the distance of the observed target and its reflectivity, allowing the construction of 3D images of the observed scene. It is common to gather the measured ToFs into a histogram of counts $y_{n,t}$, for the n th pixel and t th bin where $n \in \{1, \dots, N\}$ and $t \in \{1, \dots, T\}$, which is modelled using a Poisson distribution given by

$$y_{n,t} \sim \mathcal{P}[r_n g(t - d_n) + b_n] \quad (1)$$

where $\mathcal{P}(\cdot)$ denotes a Poisson distribution, $d_n \in \{1, \dots, T\}$ represent target range's position, $r_n \geq 0$ the reflected counts from the target, $b_n \geq 0$ denotes the background and dark counts of the detector, g is the system impulse response (SIR) assumed to be known from a calibration step and normalized ($\sum_{t=1}^T g(t) = 1$) and T is the length of the ToFs histogram. In the absence of a target, i.e. $r_n = 0$, the measured histogram reduces to background counts $y_{n,t} \sim \mathcal{P}(b_n)$. In this paper, we approximate the SIR with a Gaussian shape as it helps obtain analytical probability results in Section 4. Model (1) has been used in many studies, however, it assumes the availability of histograms which in practice should be built from the raw detected photons and ToFs and thus involve additional computational cost. In addition, modelling the data using histograms is memory consuming especially in the sparse photon regime where only few photons are detected per-pixel. In this paper, we aim to design a low memory and fast detection algorithm, thus we directly model the detected list of photons $s_{n,m}$ for the n th pixel and for $m \in \{1, \dots, \bar{y}\}$, using a mixture of densities as in [11, 12]

$$P(s_{n,m}|w_n, d_n) = \frac{(1 - w_n)}{T} + w_n g(s_{n,m} - d_n) \quad (2)$$

where $w_n = \frac{r_n}{r_n + b_n T}$ represents the probability of the detected photon to belong to a target or a uniform background and \bar{y} the total number of photons detected in the n th pixel. Model (10) shows that in absence of a target in the n th pixel (i.e., $r_n = w_n = 0$), the ToFs will be uniformly distributed as follows

$$P(s_{n,m}|w_n = 0, d_n) = 1/T. \quad (3)$$

Assuming the independence of the observed ToFs leads to the joint likelihood distribution

$$P(\mathbf{s}_n|w_n, d_n) = \prod_{m=1}^{\bar{y}} P(s_{n,m}|w_n, d_n). \quad (4)$$

where $\mathbf{s}_n = (s_{n,1}, \dots, s_{n,\bar{y}})$ gathers all detections for the n th pixel. Given a ToFs list denoted by \mathbf{s}_n , our goal is to design a fast target detection algorithm to decide if $0 < w_n \leq 1$ or $w_n = 0$, i.e. if there is a target or not. Note that this is an ill-posed inverse problem, since the parameters (w_n, d_n) are unknown in practice, and we propose to solve it using a Bayesian strategy as detailed in the next section.

3. BAYESIAN MODEL FOR TARGET DETECTION

This section introduces a Bayesian model for target detection. The Bayesian framework assigns prior distributions to the unknown parameters to include additional information and regularize the ill-posed inverse problem. The next section introduces the proposed prior distributions for the unknown parameters.

3.1. Prior distribution for w

The parameter $0 \leq w_n \leq 1$ represents the probability of the detected ToFs to belong to a background ($w_n = 0$) or a target ($0 < w_n \leq 1$). To satisfy these constraints, we assign this parameter a common spike and slab prior distribution [13] as follows

$$p(w_n|u_n) = \delta(w_n)(1 - u_n) + u_n \text{Beta}(\alpha, \beta) \quad (5)$$

where $\delta(\cdot)$ denotes the Dirac delta distribution centred in 0, $\text{Beta}(\alpha, \beta)$ is the beta distribution with known shape parameters $\alpha, \beta > 0$ and $u_n \in \{0, 1\}$ is a binary variable that indicates the presence ($u_n = 1$) or absence ($u_n = 0$) of a target. In this work, the parameters $\alpha, \beta > 0$ are assumed known and fixed to reflect our prior knowledge on the parameter w_n . The latter parameter is directly related to the signal to background (SBR) level (as follows $w_n = \frac{\text{SBR}}{1 + \text{SBR}}$, where $\text{SBR} = r_n / (b_n T)$) which allow fixing the hyper-parameters from calibration measurements. In what follows, we assume non-informative prior and fix the parameters to $\alpha = \beta = 1$.

3.2. Prior distribution for u_n

The parameter u_n is assigned a Bernoulli distribution with a probability of target presence π , i.e., $p(u_n = 1) = \pi$ and $p(u_n = 0) = 1 - \pi$. The parameter π is fixed to 0.5 in what follows, reflecting the absence of additional information regarding this parameter.

3.3. Prior distribution for d_n

A non-informative uniform prior distribution is assigned for the discrete variable d_n , as follows $p(d_n) = 1/T, \forall n$. However, this choice can be changed in presence of additional information regarding the target position.

3.4. Posterior distribution and decision rule

Using Bayes rule, the posterior distribution can be expressed as follows

$$f(w_n, d_n, u_n|\mathbf{s}_n) \propto f(\mathbf{s}_n|d_n, w_n)f(d_n)f(w_n|u_n)f(u_n) \quad (6)$$

where \propto means ‘‘proportional to’’. To perform target detection, we are interested on the marginals of the variable u_n and

build our test rule as in [9]

$$f(u_n = 0 | \mathbf{s}_n) \underset{H_1}{\overset{H_0}{\geq}} f(u_n = 1 | \mathbf{s}_n) \quad (7)$$

where H_0 , (resp. H_1) represents the absence (resp. presence) of a target and

$$f(u_n | \mathbf{s}_n) = \sum_{d_n=1}^T \int_0^1 f(w_n, d_n, u_n | \mathbf{s}_n) dw_n \quad (8)$$

The next section introduces the details to compute (8)

4. COMPUTATION OF DETECTION PROBABILITIES

Our goal is to compute the marginals in (8). It is straightforward to show that

$$p(u_n = 0 | \mathbf{s}_n) = \frac{1 - \pi}{T\bar{y}}. \quad (9)$$

To compute $p(u_n = 1 | \mathbf{s}_n)$ we first note that the joint likelihood distribution can be expressed as a polynomial, as follows

$$\begin{aligned} p(\mathbf{s}_n | w_n, d_n) &= w_n^{\bar{y}} \prod_{m=1}^{\bar{y}} [x_n + g(s_{n,m} - d_n)] \\ &= w_n^{\bar{y}} \sum_{m=0}^{\bar{y}} a_{nm}(d_n) x_n^m \end{aligned} \quad (10)$$

where $x_n = \frac{(1-w_n)}{T w_n}$, and $a_{nm}(d_n) > 0$ are expressed with respect to the sum and product of the coefficients $r_m^{d_n} = g(s_{n,m} - d_n)$ using the Vieta's formulas given by

$$a_{nm}(d_n) = \sum_{1 \leq i_1 \leq i_2 \leq \dots \leq i_k \leq \bar{y}} r_{i_1}^{d_n} r_{i_2}^{d_n} \dots r_{i_k}^{d_n} \quad (11)$$

At this stage, we approximate the SIR g (which is playing the role of signal counts distribution) by a Gaussian distribution with standard deviation σ as follows $g(s_{n,m} - d_n) = \mathcal{N}_{d_n}(s_{n,m}, \sigma^2)$. This is a common approximation that has been used in several previous studies [14] [15]. It is worth mentioning that using a continuous Gaussian distribution to represent the discrete ToF values has a limited effect on the performance of the proposed approach, this is due to the time resolution of single-photon detectors being generally very small compared to σ . Under these assumptions, Eq. (11) reduces to a sum and products of Gaussian distributions which is analytically available. The marginalization in (8) can be analytically done leading to

$$\begin{aligned} p(u_n = 1 | \mathbf{s}_n) &= \frac{\pi}{TBeta(\alpha, \beta)} \\ &\times \sum_{m=0}^{\bar{y}} \left[\frac{Beta(\bar{y} + \alpha - m, \beta + m)}{T^i} \bar{a}_{nm} \right] \end{aligned} \quad (12)$$

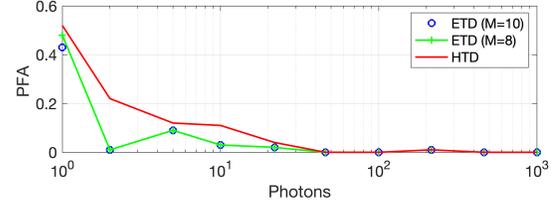


Fig. 1. Comparison of false alarm probability for the proposed method with different approximation levels, and the histogram-based method in [9].

where \bar{a}_{nm} is the result of marginalizing the Gaussians in $a_{nm}(d_n)$ with respect to d_n , where we have assumed the target location d_n is far from the observation window edges leading to $\sum_{d_n} \mathcal{N}_{d_n}(\mu, \sigma^2) \approx 1$. Although Eq. (12) shows an analytical formula for the probability of detection, it should be noted that it is a sum of combinatorial products (see (11)) that can not be computed efficiently for large $\bar{y} > 10$. Several strategies can be adopted to solve this problem and we distinguish two promising directions, (i) an iterative estimation approach where (12) is evaluated for a small number of photons M , the resulting probability is then used to update our prior distribution by setting $\pi_n^{t+1} = p(u_n^t = 1 | \mathbf{s}_n^t)$. The procedure can be repeated iteratively to account for all detected photons \bar{y} . The second strategy, which is adopted in this paper, is to approximate $p(u_n = 1 | \mathbf{s}_n)$ by limiting the number of terms summed in (11) to $K = \binom{M}{M/2} = \frac{M!}{(M/2)!(M/2)!}$, where M is a user fixed parameter ensuring better approximation for large values and ! denotes the factorial operator. Note that the complexity of the proposed algorithm is proportional to the small number of detected photons instead of the size of observation window "T" as in [9].

Finally we mention that the obtained probability maps results from an independent processing of pixels. Assuming a similar number of surfaces for adjacent pixels [9, 16], the probability maps can be post-processed to enforce spatial correlation between pixels using a total-variation regularization as in [9]. The latter procedure leads to better visual results as shown in the next section.

5. RESULTS

We first evaluate the performance of the proposed algorithm, denoted ETD for event based target detection, on simulated data. We generate the data according to model (10) with $T = 2500$ bins, $\sigma = 20$, while varying SBR in the range [0.01, 100] and the total photons \bar{y} in the range [1, 1000]. The proposed strategy is evaluated for two approximation levels $M \in \{8, 10\}$ and is compared to the histogram based TD algorithm (HTD) introduced in [9] as it showed state-of-the-art results with reduced computational time. All results are

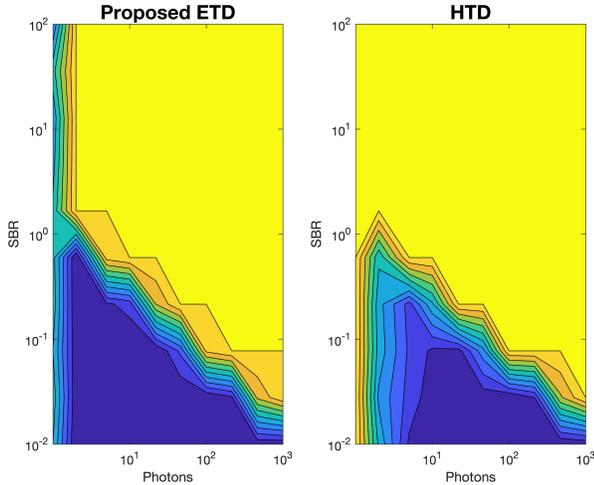


Fig. 2. Comparison of true positive (TP) probability for (left) the proposed method with $M=10$ and (right) the histogram-based method in [9].

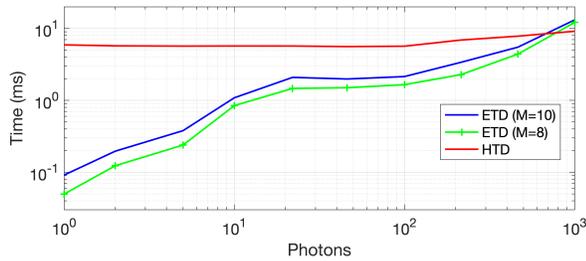


Fig. 3. Comparison of the computational time of the proposed method with different approximation levels, and the histogram-based method in [9].

obtained on Matlab 2018a on a Mac Quad-Core Intel Core i7@3.1GHz, 16 GB RAM. Fig. 1 shows the probability of false alarm (PFA) of the two algorithms highlighting the good results of the proposed strategy. The true positive (TP) probabilities are presented in Fig. 4 where the algorithm [9] shows more detection for low photons leading to better TP at the expense of a higher PFA. The main benefit of the proposed algorithm is the reduced computational time of the order of 1ms per-pixel for $M=10$, as illustrated in Fig. 3, which shows an improvement factor of 10 compared to the algorithm [9]. Fig. 3 shows however that the proposed algorithm complexity is proportional to the approximation coefficient M , and to the number of detected photons, showing best performance for $\bar{y} < 100$ per pixel.

The proposed strategy is also validated on real data. Akin to [9], we consider the mannequin face scene measured at a stand-off distance of 325 metres at midday in Heriot-Watt

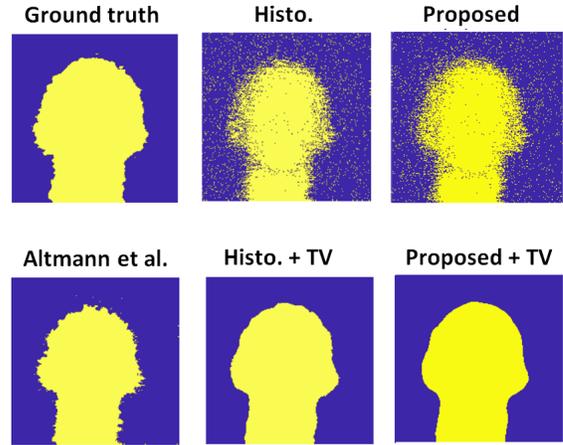


Fig. 4. Detected maps for the mannequin face with 3ms acquisition time per pixel (yellow: a detected target, blue: no target).

University, in bright conditions. The data has 200×200 pixels, $T=1700$ bins, an SBR of 0.29 with a 5th-95th percentile interval of (0.05,0.67). We focus on the data with 3ms acquisition time per pixel which has 61 average photon-per-pixel, and we refer the reader to [8,9] for more details regarding this dataset. Results in Table 1 shows the PD, PFA and computational cost of the studied methods (when enforced, spatial regularization is denoted by TV) showing good performance for the proposed strategy (for $M=10$) especially in term of computational cost. Fig. 4 shows the obtained detection maps with ETD, HTD [9] and Altmann et al[8] indicating similar performance, before and after applying spatial regularization.

Table 1. Probability of detection (PD), false alarm (PFA) and computational times (in ms) of the two methods on real data with different acquisition times. The processing time is indicated in ms for each pixel while assuming a parallel processing. The TV regularization requires 31ms for the full image.

		HTD	HTD-TV	ETD	ETD-TV
3ms data	PD(%)	80	92	85	93
	PFA(%)	4	0.07	6	0.11
	Time (ms)	6	6+ 31 (TV)	1	1+ 31 (TV)

6. CONCLUSIONS

This paper has introduced a new algorithm for fast target detection in single-photon Lidar data. In contrast to histogram based methods, the proposed strategy operates on single-photon ToF events to reduce memory requirements and ensure fast processing. The proposed algorithm showed good

performance especially in presence of few photons per pixel, which is a common scenario for rapid or long-range imaging. The algorithm can serve as a building block for higher-level applications such as adaptive sampling to improve data acquisition [17], and can be used as a pre-processing step to several reconstruction algorithms [11, 14, 18–21]. Future work includes the consideration of a different approach to enforce spatial regularization between pixels. Considering an iterative approach to approximate the marginal posterior is also interesting for the fast online processing of the detected photons. A generalization to imaging through obscurants will also be investigated.

7. ACKNOWLEDGMENT

The authors thanks Dr. Tachella for sharing the HTD code used for comparison in this work.

8. REFERENCES

- [1] A. M. Wallace, A. Halimi, and G. S. Buller, “Full waveform lidar for adverse weather conditions,” *IEEE Trans. Vehicular Tech.*, 2020, In press.
- [2] A. M. Wallace, A. McCarthy, C. J. Nichol, X. Ren, S. Morak, D. Martinez-Ramirez, I. H. Woodhouse, and G. S. Buller, “Design and evaluation of multispectral lidar for the recovery of arboreal parameters,” *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4942–4954, 2014.
- [3] A. M. Pawlikowska, A. Halimi, R. A. Lamb, and G. S. Buller, “Single-photon three-dimensional imaging at up to 10 kilometers range,” *Opt. Express*, vol. 25, no. 10, pp. 11 919–11 931, May 2017.
- [4] G. Satat, M. Tancik, and R. Raskar, “Towards photography through realistic fog,” in *Computational Photography (ICCP), 2018 IEEE International Conference on*. IEEE, 2018, pp. 1–10.
- [5] A. Halimi, A. Maccarone, A. McCarthy, S. McLaughlin, and G. S. Buller, “Object depth profile and reflectivity restoration from sparse single-photon data acquired in underwater environments,” *IEEE Trans. Comput. Imaging*, vol. 3, no. 3, pp. 472–484, 2017.
- [6] A. Maccarone, A. McCarthy, X. Ren, R. E. Warburton, A. M. Wallace, J. Moffat, Y. Petillot, and G. S. Buller, “Underwater depth imaging using time-correlated single-photon counting,” *Opt. Express*, vol. 23, no. 26, pp. 33 911–33 926, Dec 2015.
- [7] R. Tobin, A. Halimi, A. McCarthy, M. Laurenzis, F. Christnacher, and G. S. Buller, “Three-dimensional single-photon imaging through obscurants,” *Opt. Express*, vol. 27, no. 4, pp. 4590–4611, Feb 2019.
- [8] Y. Altmann, X. Ren, A. McCarthy, G. S. Buller, and S. McLaughlin, “Robust bayesian target detection algorithm for depth imaging from sparse single-photon data,” *IEEE Trans. Comput. Imaging*, vol. 2, no. 4, pp. 456–467, 2016.
- [9] J. Tachella, Y. Altmann, S. McLaughlin, and J. . Tourneret, “Fast surface detection in single-photon lidar waveforms,” in *EUSIPCO-19*, A Coruna, Spain, 2019, pp. 1–5.
- [10] J. Tachella, Y. Altmann, S. McLaughlin, and J.-Y. Tourneret, “On fast object detection using single-photon lidar data,” in *Wavelets and Sparsity XVIII*, vol. 11138. SPIE, 2019, pp. 252 – 261.
- [11] J. Rapp and V. K. Goyal, “A few photons among many: Un-mixing signal and noise for photon-efficient active imaging,” *IEEE Trans. Comput. Imaging*, vol. 3, no. 3, pp. 445–459, Sept. 2017.
- [12] Y. Altmann and S. McLaughlin, “Range estimation from single-photon lidar data using a stochastic em approach,” in *EUSIPCO-18*, Rome, Italy, Sept. 2018, pp. 1112–1116.
- [13] E. I. George and R. E. McCulloch, “Variable selection via gibbs sampling,” *Journal of the American Statistical Association*, vol. 88, no. 423, pp. 881–889, 1993.
- [14] A. Halimi, Y. Altmann, A. McCarthy, X. Ren, R. Tobin, G. S. Buller, and S. McLaughlin, “Restoration of intensity and depth images constructed using sparse single-photon data,” in *Proc. EUSIPCO*, 2016, pp. 86–90.
- [15] Y. Altmann, S. McLaughlin, and M. E. Davies, “Fast online 3d reconstruction of dynamic scenes from individual single-photon detection events,” *IEEE Transactions on Image Processing*, vol. 29, pp. 2666–2675, 2020.
- [16] S. Hernandez-Marin, A. M. Wallace, and G. J. Gibson, “Multilayered 3D lidar image construction using spatial models in a bayesian framework,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 6, pp. 1028–1040, June 2008.
- [17] A. Halimi, P. Ciuciu, S. McLaughlin, and G. S. Buller, “Fast adaptive scene sampling for single-photon 3d lidar images,” in *IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Guadeloupe, France, Dec 2019.
- [18] J. Tachella, Y. Altmann, X. Ren, A. McCarthy, G. S. Buller, S. McLaughlin, and J.-Y. Tourneret, “Bayesian 3d reconstruction of complex scenes from single-photon lidar data,” *SIAM Journal on Imaging Sciences*, vol. 12, no. 1, pp. 521–550, 2019.
- [19] A. Halimi, R. Tobin, A. McCarthy, J. Bioucas-Dias, S. McLaughlin, and G. S. Buller, “Robust restoration of sparse multidimensional single-photon lidar images,” *IEEE Trans. Comput. Imaging*, vol. 6, pp. 138–152, 2020.
- [20] D. Shin, F. Xu, F. N. C. Wong, J. H. Shapiro, and V. K. Goyal, “Computational multi-depth single-photon imaging,” *Opt. Express*, vol. 24, no. 3, pp. 1873–1888, Feb 2016.
- [21] R. Tobin, A. Halimi, A. McCarthy, X. Ren, K. J. McEwan, S. McLaughlin, and G. S. Buller, “Long-range depth profiling of camouflaged targets using single-photon detection,” *Optical Engineering*, vol. 57, pp. 031 303 (1–10), 2017.

Robust depth imaging in adverse scenarios using single-photon Lidar and beta-divergences

Q. Legros*, S. McLaughlin, Y. Altmann
School of Engineering and Physical Sciences
Heriot-Watt University
 Edinburgh, United Kingdom
 qll1@hw.ac.uk

S. Meignen
Laboratoire Jean Kuntzmann
University Grenoble Alpes
 Grenoble, France

Mike E. Davies
School of Engineering
University of Edinburgh
 Edinburgh, United Kingdom

Abstract—This paper addresses the problem of robust estimation of range profiles from single-photon Lidar waveforms associated with single surfaces using a simple model. In contrast to existing methods explicitly modeling nuisance photon detection events, the observation model considered neglects such events and the depth parameters are instead estimated using a cost function which is robust to model mismatch. More precisely, the family of β -divergences is considered instead of the classical likelihood function. This reformulation allows the weights of the observations to be balanced depending on the amount of robustness required. The performance of our approach is assessed through a series of experiments using synthetic data under different observation scenarios. The obtained results demonstrate a significant improvement of the robustness of the estimation compared to state-of-the-art pixelwise methods, for different background illumination and imaging scenarios.

Index Terms—3D reconstruction, Single-photon lidar, Robust estimation.

I. INTRODUCTION

Light detection and ranging (Lidar) systems have received an increasing interest in the past few years for their ability to efficiently reconstruct 3D scenes at high resolution [1], [2]. One particular case is single photon Lidar (SPL) that uses a high repetition rate pulsed laser source in conjunction with single-photon detectors. One of the main advantages of this technology is its temporal resolution allowing sub-millimeter depth estimation, which makes SPL particularly attractive for a variety of problems such as long range imaging [3], [4], [5], underwater imaging [6], [7], or even through obscurants [8]. Lidar technologies allow the acquisition of the depth structure of scenes, by analysing at the time-of-flight (ToF) of photons originally emitted by a laser source and reflected by surfaces of interest. More precisely, time correlated single-photon counting (TCSPC), which is used in SPL, correlates the time-of-arrivals (ToAs) of detected photons with the time of emission of the last pulse, and often produces ToA histograms. Repeating this acquisition process for different pulse emission

directions allows a region of the 3D space to be sensed and reconstructed. However, additional detection events occur in the presence of strongly scattering media and additional light sources, such as solar illumination. Although these events are generally modelled as uniformly distributed, they can present more complex distributions. This is typically the case for instance when imaging through obscurants, where light scattering can produce a significant number of detection events shortly after the pulse emission (photons are reflected with high probability before they can reach the scene of interest).

Bayesian approaches have demonstrated their efficiency to perform a depth profile estimation of the illuminated scene from the Lidar measurements in many different applications [9], [10], [11], [12]. However, the quality of the estimates depends on that of the likelihood (or observation model) used. In this paper, we specifically concentrate on the choice/design of this data fidelity term. Traditionally in SPL analysis, studies have introduced complex parametric models to approximate as accurately as possible the actual data acquisition process, including background illumination, broadening of the system impulse response [13], attenuation due to scattering [14] and detector dead-time [15], [16], [17]. However, such models often comes with an increased computational complexity of the depth estimation process, e.g. can require iterative algorithms.

In this work a simple observation model is considered, whereby we assume that only signal photons, those originally omitted by the laser source, can be detected. The main focus of this work is to estimate the distance between the imaging system and the scene. Although intensity information can be important, it is not addressed here. To overcome the limitations of the simple model used, we do not adopt a maximum likelihood approach but instead use β -divergences [18], [19] to define a more robust depth estimators. Using this family of divergences, we can reinterpret the classical depth estimator via matched filtering (MF), seen as a specific minimum divergence depth estimator for SPL. Using β -divergences for SPL has been recently investigated, in a pseudo Bayesian framework in [20], where the main focus was to propose an online 3D imaging method. Here, we do not adopt a Bayesian perspective and concentrate on pixelwise, regularization-free depth estimation, to better understand the benefits of the β -divergence in various illumination conditions and several types

This work was supported by the Royal Academy of Engineering under the Research Fellowship scheme RF201617/16/31, the UK Defence Science and Technology Laboratory (DSTL X1000114765) and by the Engineering and Physical Sciences Research Council (EPSRC) (grants EP/N003446/1, EP/T00097X/1 and EP/S000631/1) and the MOD University Defence Research Collaboration (UDRC) in Signal Processing. M. D. acknowledges support from the Royal Society Wolfson Research Merit Award and the ERC advanced grant C-SENSE, project 694888.

of model mismatch.

The remainder of this paper is organized as follows. Section II introduces and motivates the simple observation model considered in this study. The divergence used for robust depth estimation in the presence of an imperfectly known system impulse response function (IRF) is introduced in Section III. A comparison to state-of-the-art approaches using synthetic data is conducted in Section IV to identify benefits of the proposed approach. Conclusions are finally reported in Section V.

II. OBSERVATION MODEL

In this paper, we address the pixelwise estimation of object range from SPL data. Thus, to simplify notation, we omit indices representing pixel dependency. Two main pixelwise representations of photon ToAs are currently used in the context of SPL. The earliest methods have considered each pixel as a ToA histogram $\mathbf{y} = [y_1, \dots, y_T]^\top$, which consists of T non-overlapping temporal bins (the bin width being usually given by the temporal resolution of the detector). Note that in this work, we implicitly assume that the repetition period of the laser source is T , where the arbitrary time unit is width of a temporal bin. However, with the development of high-resolution photodetectors, T can be extremely large (although the actual temporal span of the histogram remains constant, the width of the time bins decreases) and the measured ToAs can now also be seen as continuous variables [5]. In a context of photon-starved measurements, the alternative ToA representation is simply based on sets of individual photon ToAs. If we assume that P photons are detected, the observations are denoted by $\mathbf{s} = \{s^p\}_{p=1}^P$, where s^p is the ToA of the p th detected photon.

Neglecting detector dark counts and additional light sources apart from the classical emission laser and assuming the a single surface is visible in the field of view, the observation model for any ToA s can be expressed as

$$f(s|d) = h\left(s - \frac{2d}{c}\right), \quad (1)$$

where h is the normalized IRF associated with the imaging system. This IRF is generally measured during calibration of the Lidar-based imaging system. In (1), c is the speed of light and d is the distance to the target, such that $\frac{2d}{c}$ is the characteristic ToF associated with the illuminated target. Eq. (1) implicitly assumes that the scene is approximately static and that the shape of $h(\cdot)$ remains the same for all the admissible values of d . When P photons are detected, if the dead-time of the detector can be neglected, the photon ToAs are mutually independent (given d) and the joint likelihood can be expressed as

$$f(\mathbf{s}|d) = \prod_{p=1}^P f(s^p|d). \quad (2)$$

III. ROBUST ESTIMATION USING β DIVERGENCES

The model in (2) is simple (it only depends on a single parameter d per pixel) but is often not accurate enough,

especially when ambient illumination cannot be neglected. In particular, using maximum likelihood (ML) estimation strategies using Eq. (1) to infer d yields poor estimation performance.

In general, the observation model used for depth estimation is chosen to be "similar" to the actual (to usually unknown) data distribution, to enable reliable parameter estimation. The similarity measure used also impacts the estimation performance. For instance, the estimator constructed from the ML criterion can also be seen as the estimator minimizing the Kullback-Leibler (KL) divergence $\text{KL}(\hat{g}(s)||f(s|d))$, between the empirical data distribution $\hat{g}(s) = \frac{1}{P} \sum_{p=1}^P \delta(s - s^p)$ and the postulated observation model in (1) ($\delta(\cdot)$ stands for the Dirac delta function). Instead of using the traditional KL as similarity measure, here we consider a more general family of divergences, to reflect the potential mismatch between the actual data distribution and the postulated model. The main objective here is to use a similarity measure that leads to a robust and computationally attractive depth estimator, where *robust* refers to the presence of spurious detection events.

In this work, we consider the family of β -divergences, defined by

$$\begin{aligned} D_\beta(g||f) = & \int_0^T f^{1+\beta}(x|d) - \frac{1+\beta}{\beta} [g(x)f^\beta(x|d)] \\ & + \frac{1}{\beta} [(g)^{1+\beta}(x)] dx, \quad \beta > 0. \end{aligned} \quad (3)$$

to measure the similarity between two distributions g and f . In a similar fashion to the ML estimation which reduces to minimizing $\text{KL}(\hat{g}(s)||f(s|d))$, here we estimate the depth in (1) by minimizing $D_\beta(\hat{g}(s)||f(s|d))$. Under mild assumptions, the resulting estimator is given

$$\hat{d} = \underset{d}{\operatorname{argmax}} \left\{ \frac{1+\beta}{\beta P} \sum_{p=1}^P f^\beta(s^p|d) - \text{Const.} \right\}, \quad (4)$$

where the constant corresponds to the first and third terms on the right-hand side of Eq. (3). While the third term does not depend on d , the first term does not either in practice as we assume that the shape and the integral of $h(\cdot)$ does not depend on d over its domain of definition ($2d/c$ is expected to be far from 0 and T). Note that $f^{1+\beta}(x|d)$ depends on divergence parameter β though.

An interesting link with histogram-based depth estimation methods and (4) has been discussed in [20] and is briefly recalled here. If the data in \mathbf{s} are represented using \mathbf{y} , i.e., a set photons counts being detected in each of the T time bins, Eq. (4) can be rewritten

$$\hat{d} = \underset{d}{\operatorname{argmax}} \sum_{t=1}^T y_t h^\beta\left(t - \frac{2d}{c}\right), \quad (5)$$

which corresponds to maximizing the cross-correlation between $[h^\beta(1 - \frac{2d}{c}), \dots, h^\beta(T - \frac{2d}{c})]^\top$ and \mathbf{y} .

The depth estimation based on (5) (or (4)) depends on the divergence parameter β . Two special cases to be mentioned are

1) when $\beta = 1$, where solving Eq. (5) reduces to matched-filtering the data with $h^\beta(t - \frac{2d}{c})$, and 2) when $\beta \rightarrow 0$, where solving Eq. (5) reduces to log-matched filtering the data with the logarithm of $h(t - \frac{2d}{c})$. In the latter case, the resulting estimator is the classical ML estimator (the β -divergence converges to the KL divergence when $\beta \rightarrow 0$).

Solving Eq. (4) instead of maximizing the classical log-likelihood leads to a pseudo-ML estimation, also referred to as minimum divergence (MD) estimation.

IV. RESULTS

In this section, we assess the depth reconstruction performance of the proposed approach by comparing it to that of existing pixelwise estimation procedures. We first consider two IRFs, (depicted in Fig.1 top), i.e., a real asymmetric IRF measured in [21] and a Gaussian IRF presenting the same full width at half maximum (FWHM) (28 bins, each bin representing a 2ps interval) and the same mode. Based on these IRFs, we generated synthetic data with $T = 1500$ and added different types of model mismatch, including constant and non-uniform background levels and measured peaks broader than the original IRF, referred to as IRF broadening, as typically occurs when the surface imaged is not orthogonal to the beam direction [13]. For all the methods considered, we set the admissible temporal range to $[t_{min}, t_{max}] = [101, 1400]$, which ensures that the integrals of the IRFs remain constant over the admissible object range. All the results presented have been averaged over $N_{iter} = 2000$ Monte Carlo realizations. Different scenarios have been reproduced to assess the performance of the proposed method through both the mean signal photon counts (MSC) and the average signal to background ratio (SBR). The MSC is the number of detected photons originally emitted by the laser and whose distribution is given by (1), and the SBR is defined as the ratio of the MSC over the total number of nuisance detection events. For each scenario, data have been simulated using different MSCs, ranging from 1 to 10^3 and SBRs between 10^2 and 10^{-4} , with the ground truth depth fixed to correspond to the 620th bin.

Our method is compared to ML and robust estimators for pixelwise depth estimation. More precisely, we considered the log-matched filtering (LMF) approach, which is the ML estimator based on Eq. (2), the robust Huber’s estimator [22], [23], the Half-sample-mode (HSM) estimator [24]. For all the simulations, we also considered the Oracle estimator, i.e., the ML estimator of the depth based on the true model used to generate the data, with other model parameters (e.g. SBR and MSC) being set to their actual values. Note that Huber’s estimator requires a user-defined hyperparameter to be tuned, and it has been fixed to 0.4 for the experiments conducted in this work, aiming to discard 80% of the detected photons prior to estimating the truncated mean with the remaining data. This value has been set such that the performance of the estimator remains satisfactory across a range of SBR values.

To quantify the depth estimation quality, we computed the probability of *accurately* estimating the target depth, whereby a detection is deemed accurate if the absolute error between

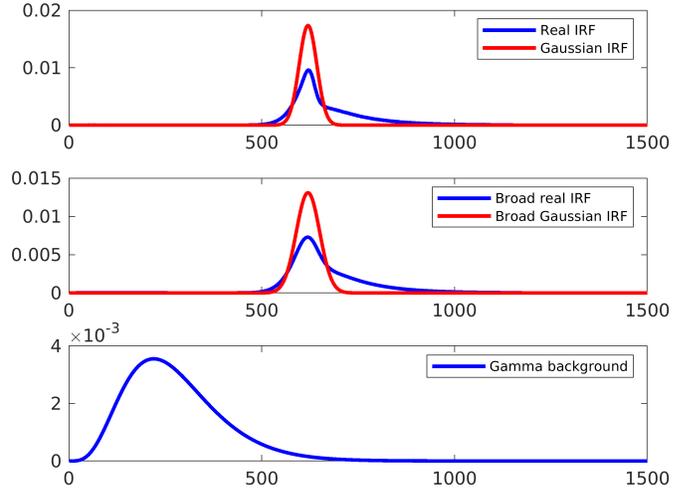


Fig. 1. Top: Real asymmetric (red) and Gaussian IRFs used to simulate the data used for the experiments in Sections IV-A and IV-C. Middle: broadened IRFs used in Section IV-B, obtained by convolving the IRFs from the top plot with a Gaussian of standard deviation 20. Bottom: gamma distribution used to generate non-uniform background detection events in Section IV-C.

the estimated depth and the ground truth is below a threshold that has been fixed to the IRF FWHM. The curves displayed in Fig. 2 represent for each SBR the MSC necessary to reach a probability of accurate detection higher than 85% (the working region of each method is on the right-hand side of each curve).

A. Constant background level

In this section, we first investigate the robustness of the selected methods, in the case of constant background levels corrupting the observations. The main results are presented in Fig. 2.

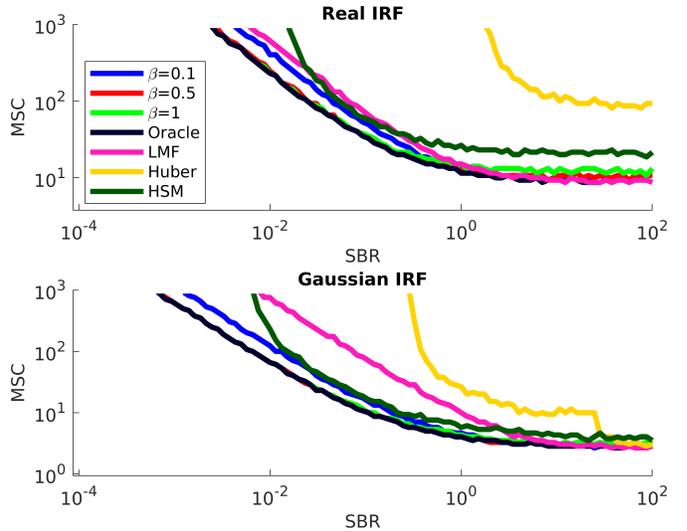


Fig. 2. Threshold of accurate detection higher than 85% for different robust methods, as function of the MSC and SBR. Top: results related to data generated with the real IRF displayed in red in Fig. 1top. Bottom : histograms generated with the Gaussian IRF approximation displayed in blue in Fig. 1 top.

Note first in Fig. 2 that using the real IRF (top) or its Gaussian approximation (bottom) does not modify the order of performances of the compared methods. Although the IRF shape does not seem to affect the relative performance of the methods considered, it impacts the overall performance as show in Fig. 2 (bottom) where the symmetry of the IRF seems to improve the estimation performance. The best estimation performance is obtained by the proposed MD for β close to 1, whereas the worst estimation is obtained with Huber. The performance of the MD is significantly improved when β is increased, but this tails off when β gets closer to 1. The detection threshold associated with the MD estimators converge to that of the Oracle when β tends to 1. Even though high values of β enhance the depth estimation performance for low SBR cases, we observe the opposite phenomena when the SBR is higher than 1. HSM gives on average worse performance than MD, but performs better than Huber.

B. Broadening of the IRF

In this section, we assess the performance of our method in situations where the empirical IRF is broader than the postulated IRF (see Fig. 1 middle). IRF broadening can occur when surfaces observed that are not orthogonal to direction of the laser beam and when the size of the laser footprint on target can no longer be neglected. It can also occur in the presence of partially transparent materials, whereby part of the light penetrates deeper into the objects before being reflected (e.g. forest canopy). For simplicity, the broadened IRFs (see Fig. 1 middle) are modeled here by convolving the true IRFs from Fig. 1 top) by a Gaussian kernel whose standard is equal 20 bins.

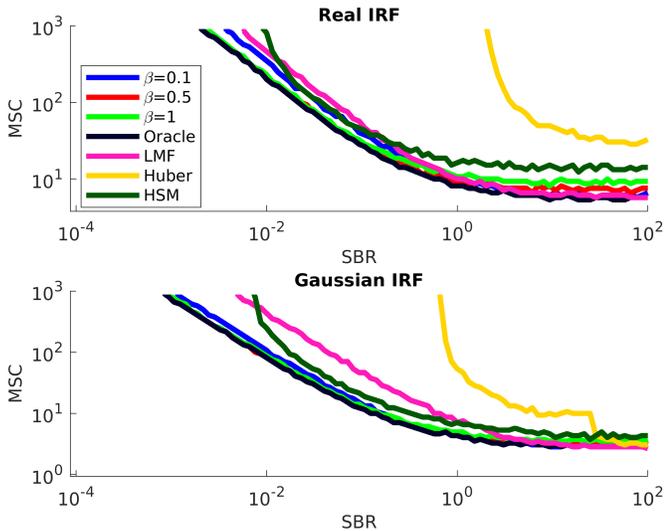


Fig. 3. Threshold of accurate detection higher than 85% for different robust methods, as function of the MSC and SBR. Top : results obtained with a broad version of the real IRF displayed in red in Fig. 1top. Bottom : results obtained with a broad version of the Gaussian IRF displayed in blue in Fig. 1top.

The main results obtained with broadened IRFs are depicted in Fig. 3. This figure shows that the estimation performance

of the MD estimators and the LMF remain similarly to that depicted in Fig. 2. The best performance is still obtained by the Oracle that gives the SBR bound (for a given MSC) under which all the other methods provide less than 85% of accurate detection. The MD estimator for high values of β provides the most robust reconstructions whereas Huber provides the least satisfying ones (still relative to the fixed detection threshold). HSM achieves similar performance than in Fig. 2, and is less robust than the MD estimators. As in Fig. 2, the symmetry of the IRF used in Fig. 3 (bottom) enhances slightly the estimation performance of all the methods. However, the Huber estimator seems more affected by the asymmetry of the IRF (Fig. 3 (top)).

C. Non-uniform background

Imaging scenarios in presence of scattering media are receiving growing interest in underwater and automotive (e.g. fog and rain) applications. In such cases, the background temporal profile is expected to follow a non-uniform distribution. To investigate this issue, we applied our method to data generated with a gamma background distribution, depicted in Fig. 1 (bottom). The parameters of the gamma distribution have been set to 5 bins (shape) and 55 bins (scale) and this profile is similar to that observed in [7].

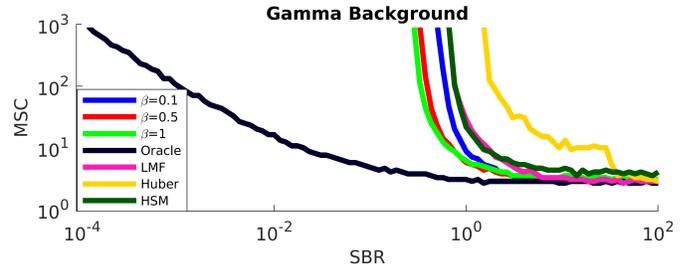


Fig. 4. Threshold of accurate detection higher than 85% for different robust methods, as function of the MSC and SBR. The outliers corresponds to background photons following a gamma distribution, and the Gaussian IRF (see Fig. 1 top) has been used to generate and analyze the data.

The estimation performance of the different methods is assessed in Fig. 4. As in Figs. 2 and 3, the MD method provides better results as β increases. The worst estimation are obtained with Huber and HSM performs similarly than LMF in this case. Note that here, the Oracle performs significantly better than in Fig. 2, due to the non-uniform nature of the noise. While the proposed method can mitigate the presence of uniform background and IRF broadening, its gain here is more limited. Consequently, using a parametric model (which accounts for the shape of the background distribution) is more adapted.

D. Computational complexity

The two pixelwise representations of the photon ToAs introduced in Section II (involving \mathbf{y} or \mathbf{s}) play a key role in the complexity of the proposed estimator. Although the use of \mathbf{y} allows us to retrieve the matched filtering formulation, the best ToA representation is still user/scenario dependent. While the

evaluation of the cost function in (5) depends linearly on the number of (non-empty) histogram bins in \mathbf{y} , which is at most T , the evaluation of the cost function in (4) depends on the number of detected photons P (assuming the all have different ToAs). Thus, the most suited representation will depend on the amount of ToAs acquired.

The quantitative comparison of the different methods used in this paper in terms of computational complexity is not included here as the complexity can be highly platform (sequential or parallel) and implementation dependent. Nonetheless, it is possible to qualitatively compare their complexity. The proposed robust method has roughly the same computational cost as the standard LMF as it reduces to computing a similar cross-correlation (LMF uses $\log(f(\cdot))$ while MD uses $f^\beta(\cdot)$). Huber is generally faster than MD and LMF for low photon counts, but is not well adapted to thousands of photon counts. Finally, HSM relies on an iterative algorithm whose number of iterations grows with the number of photons and is not adapted to large photon counts either (it becomes rapidly slower than Huber and MD/LMF).

V. CONCLUSION

In this paper, we proposed a new depth estimator for robust estimation of the range profile from single-photon Lidar data, in the presence of non negligible background. The proposed formulation of the problem significantly simplifies the estimation process as it relies on the estimation of a single parameter, i.e., the depth parameter. To alleviate the robustness issues of the classical maximum likelihood approach, β -divergences are used instead of the Kullback-Leibler divergence to quantify the similarity between the empirical data distribution and the postulated distribution. We compared the estimation performance of the proposed estimator to that of existing pixelwise approaches for different observation scenarios, and demonstrated its potential benefits over the classical pixelwise log-matched filter. Moreover, the estimation process benefits from a low complexity, similar to that of log-matched filtering. Beyond pixelwise estimation, such a robust estimation strategy can be coupled, as in [20], with prior/regularization terms to further enhance the estimation performance. However, using such β -divergences within a Bayesian framework requires further investigation to better understand and balance the relative weights of the data fidelity and regularization terms. Future work include the consideration of alternative families of divergences or similarity measures. Moreover, it would also be interesting to investigate if such robust methods could be used to mitigate dead-time detector limitations [16], [17].

REFERENCES

- [1] T. Bosch, "Laser ranging: a critical review of usual techniques for distance measurement," *Opt. Eng.*, vol. 40, no. 1, 2001.
- [2] P. Vines, K. Kuzmenko, J. Kirdoda, D. Dumas, M. Mirza, R. Millar, D. Paul, and G. Buller, "High performance planar ge-on-si single-photon avalanche diode detectors," *Nature Communications*, Sept. 2018.
- [3] A. M. Pawlikowska, A. Halimi, R. A. Lamb, and G. S. Buller, "Single-photon three-dimensional imaging at up to 10 kilometers range," *Opt. Express*, vol. 25, no. 10, pp. 11 919–11 931, May 2017.
- [4] Z.-P. Li, X. Huang, Y. Cao, B. Wang, Y.-H. Li, W. Jin, C. Yu, J. Zhang, Q. Zhang, C.-Z. Peng, F. Xu, and J.-W. Pan, "Single-photon computational 3D imaging at 45 km," 2019.
- [5] Z.-P. Li, X. Huang, P.-Y. Jiang, Y. Hong, C. Yu, Y. Cao, J. Zhang, F. Xu, and J.-W. Pan, "Super-resolution single-photon imaging at 8.2 kilometers," *Opt. Express*, vol. 28, no. 3, pp. 4076–4087, 2020.
- [6] A. Maccarone, A. McCarthy, X. Ren, R. E. Warburton, A. M. Wallace, J. Moffat, Y. Petillot, and G. S. Buller, "Underwater depth imaging using time-correlated single-photon counting," *Opt. Express*, vol. 23, no. 26, pp. 33 911–33 926, 2015.
- [7] A. Maccarone, F. M. Della Rocca, A. McCarthy, R. Henderson, and G. S. Buller, "Three-dimensional imaging of stationary and moving targets in turbid underwater environments using a single-photon detector array," *Opt. Express*, vol. 27, no. 20, pp. 28 437–28 456, 2019.
- [8] R. Tobin, A. Halimi, A. McCarthy, M. Laurenzis, F. Christnacher, and G. S. Buller, "Three-dimensional single-photon imaging through obscurants," *Opt. Express*, vol. 27, no. 4, pp. 4590–4611, Feb. 2019.
- [9] R. Tobin, Y. Altmann, X. Ren, A. McCarthy, R. A. Lamb, S. McLaughlin, and G. S. Buller, "Comparative study of sampling strategies for sparse photon multispectral Lidar imaging: towards mosaic filter arrays," *Journal of Optics*, vol. 19, no. 9, p. 094006, Aug. 2017.
- [10] Y. Altmann, A. Wallace, and S. McLaughlin, "Spectral unmixing of multispectral Lidar signals," *IEEE Trans. Signal Process.*, vol. 63, no. 20, pp. 5525–5534, Oct. 2015.
- [11] J. Tachella, Y. Altmann, M. Márquez, H. Arguello-Fuentes, J.-Y. Tourneret, and S. McLaughlin, "Bayesian 3D reconstruction of subsampled multispectral single-photon Lidar signals," *IEEE Trans. Comput. Imaging*, Aug. 2019.
- [12] A. Halimi, R. Tobin, A. McCarthy, J. M. Bioucas-Dias, S. McLaughlin, and G. S. Buller, "Robust restoration of sparse multidimensional single-photon Lidar images," *IEEE Trans. Comput. Imaging*, 2019.
- [13] J. Tachella, Y. Altmann, S. McLaughlin, and J. . Tourneret, "3D reconstruction using single-photon lidar data exploiting the widths of the returns," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, 2019, pp. 7815–7819.
- [14] A. Maccarone, A. McCarthy, A. Halimi, J. Tachella, P. S. Chhabra, Y. Altmann, A. M. Wallace, S. McLaughlin, Y. R. Petillot, and G. S. Buller, "Time-correlated single-photon counting for single and multiple wavelength underwater depth imaging," in *Advanced Photon Counting Techniques XII*, vol. 10659. International Society for Optics and Photonics, 2018, p. 106590R.
- [15] A. Gupta, A. Ingle, and M. Gupta, "Asynchronous single-photon 3D imaging," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7909–7918.
- [16] A. Gupta, A. Ingle, A. Velten, and M. Gupta, "Photon-flooded single-photon 3D cameras," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6770–6779.
- [17] J. Rapp, Y. Ma, R. M. Dawson, and V. K. Goyal, "Dead time compensation for high-flux ranging," *IEEE Trans. Signal Process.*, vol. 67, no. 13, pp. 3471–3486, 2019.
- [18] F. Futami, I. Sato, and M. Sugiyama, "Variational inference based on robust divergences," in *Intern. Conf. on Artificial Intel. and Stat., AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain*, A. J. Storkey and F. Pérez-Cruz, Eds., vol. 84. PMLR, 2018, pp. 813–822.
- [19] A. Basu, I. R. Harris, N. L. Hjort, and M. Jones, "Robust and efficient estimation by minimising a density power divergence," *Biometrika*, vol. 85, no. 3, pp. 549–559, 1998.
- [20] Q. Legros, J. Tachella, R. Tobin, A. McCarthy, S. Meignen, G. S. Buller, Y. Altmann, S. McLaughlin, and M. E. Davies, "Robust 3D reconstruction of dynamic scenes from single-photon lidar using beta-divergences," April 2020, unpublished.
- [21] R. Tobin, Y. Altmann, X. Ren, A. McCarthy, R. A. Lamb, S. McLaughlin, and G. S. Buller, "Comparative study of sampling strategies for sparse photon multispectral lidar imaging: towards mosaic filter arrays," *J. of Optics*, vol. 19, no. 9, p. 094006, aug 2017.
- [22] P. J. Huber, "Robust estimation of a location parameter," *Ann. Math. Statist.*, vol. 35, no. 1, pp. 73–101, 03 1964.
- [23] R. Wilcox, *Introduction to Robust Estimation and Hypothesis Testing*, ser. Statistical Modeling and Decision Science. Elsevier Science, 2012.
- [24] D. R. Bickel and R. Früwirth, "On a fast, robust estimator of the mode: Comparisons to other robust estimators with applications," *Computational Statistics & Data Analysis*, vol. 50, no. 12, pp. 3500 – 3530, 2006.

Optimising Network Architectures for Provable Adversarial Robustness

Henry Gouk
School of Informatics
University of Edinburgh
 Edinburgh, United Kingdom
 hgouk@inf.ed.ac.uk

Timothy M. Hospedales
School of Informatics
University of Edinburgh
 Edinburgh, United Kingdom
 thospedales@ed.ac.uk

Abstract—Existing Lipschitz-based provable defences to adversarial examples only cover the ℓ_2 threat model. We introduce the first bound that makes use of Lipschitz continuity to provide a more general guarantee for threat models based on any ℓ_p norm. Additionally, a new strategy is proposed for designing network architectures that exhibit superior provable adversarial robustness over conventional convolutional neural networks. Experiments are conducted to validate our theoretical contributions, show that the assumptions made during the design of our novel architecture hold in practice, and quantify the empirical robustness of several Lipschitz-based adversarial defence methods.

Index Terms—Artificial Neural Network, Computer Vision

I. INTRODUCTION & RELATED WORK

The robustness of deep neural networks to adversarial attack [1] is an increasingly topical issue as deep models are becoming more widely deployed in practice. This paper focuses on the problem of ensuring that once a deep network trained for image classification has been deployed, one can be confident that an adversary has only a limited ability to maliciously impact model predictions when they tamper with the system inputs. Such malicious inputs, so-called adversarial examples, appear to humans as normal images, but in reality they have undergone imperceptible modifications that cause a model to make an incorrect prediction. The majority of research into adversarial examples is still based on empirical results that have been shown to be somewhat fragile [2], [3]. In contrast, we look to the more recent trends in provable adversarial robustness, where it one is able to compute a certificate for each prediction made by the model, ensuring that it is robust to some pre-specified family of attacks, known as the threat model [4], [5].

There are several papers in the literature on deep learning that address adversarial robustness through the use of Lipschitz continuity, but they focus solely on perturbations with bounded Euclidean norm. Tsuzuku et al. [4] present an efficient method for determining whether an example could have been tampered with at test time or, conversely, certify that a prediction has not been influenced by an adversarial attack. They compare the prediction margin normalised by the Lipschitz constant of the network to the magnitude of the largest perturbation allowed by the threat model, allowing them to determine whether the input could be an adversarial example. Farnia et al. [6]

show how the adversarial risk can be bounded in terms of the training loss by adapting the bound of [7] to consider perturbations to the margin, using a similar technique to [4]. The analysis in this paper takes a similar high-level strategy—making use of margins and Lipschitz constants—but we extend this theory to threat models based on arbitrary p -norms, and provide a simpler proof than previous methods [4]. Huster et al. [8] demonstrate that current methods for regularising Lipschitz constants of networks have deficiencies when used for improving adversarial robustness. Specifically, it is shown that existing approaches for regularising the Lipschitz constant may be too restrictive because the bound on the Lipschitz constant is too loose, resulting in over-regularisation. We take an orthogonal approach: we provide theoretical and practical contributions that are compatible with arbitrary bounds on the Lipschitz constant.

Existing work that aims to provide theory-backed guarantees for adversarial robustness has resulted in several techniques able to certify whether a prediction for a particular example is immune to adversarial attack under a threat model based on ℓ_p -norm perturbation size. [9] propose a method that can only be applied to networks composed of fully connected layers with rectified linear units activation functions, and no batch normalisation. [10] present an approach based on solving an optimisation problem. While the robustness estimates they give are considerably tighter than many other certification methods, they scale very poorly to networks with large input images or feature maps. In contrast to these methods, our approach bounds the expected adversarial generalisation error, has virtually no test-time computational overhead, and can be applied to arbitrary feed-forward architectures. Bounding the expected generalisation error enables us to give guarantees about the level of robustness a model will have once it has been deployed. Existing approaches to provable robustness do not come with such guarantees, and can only provide certification for individual instances.

We begin by extending existing theory addressing the relationship between Lipschitz continuity and provable adversarial robustness. Using insights from the resulting bounds, it is shown how one can adjust network architectures in such a way that Lipschitz-based regularisation methods are more effective. Experimental results show that, while having little

difference in clean performance compared to existing Lipschitz-based defences, our approach improves the level of *provable* robustness significantly.

II. GENERALISATION UNDER ATTACK

Methods for estimating the generalisation performance of learned models typically assume examples, (\vec{x}, y) , observed at both training and testing time are independently drawn from the same distribution, \mathcal{D} . Such methods estimate or bound the expected risk,

$$R^\ell(f) = \mathbb{E}_{(\vec{x}, y) \sim \mathcal{D}}[\ell(f(\vec{x}), y)], \quad (1)$$

of a classifier, f , with respect to some loss function, ℓ . The standard technique for estimating the expected risk in deep learning is to use an empirical approximation measured on a set of held-out data. In the adversarial setting one must consider the expected risk when under the influence of an attacker that can add perturbations to feature vectors at test time,

$$\tilde{R}_{p,t}^\ell(f) = \mathbb{E}_{(\vec{x}, y) \sim \mathcal{D}} \left[\max_{\vec{\epsilon}: \|\vec{\epsilon}\|_p \leq t} \ell(f(\vec{x} + \vec{\epsilon}), y) \right], \quad (2)$$

which is known as the adversarial risk [5]. In contrast to the expected risk, $\tilde{R}_{p,t}^\ell(f)$ cannot be reliably approximated from data when f is nonlinear, as one must find the globally optimal setting of $\vec{\epsilon}$ for each data point in the held-out set.

For a hypothesis, f , that produces a vector of real-valued scores, each associated with a possible class, we define the margin function as

$$m_f(\vec{x}, y) = f_y(\vec{x}) - \max_{j \neq y} f_j(\vec{x}), \quad (3)$$

where $f_i(\vec{x})$ is the i th component of the output of $f(\vec{x})$. Typical loss functions for measuring the performance of a model via composition with the margin function include the zero-one loss and the hinge. These compositions result in the classification error rate and the multi-class hinge loss variant proposed by [11], respectively.

Proposition 1. *If f is k -Lipschitz with respect to the p -norm and $\ell: \mathbb{R} \rightarrow \mathbb{R}^+$ is a monotonically decreasing loss function, then*

$$\max_{\vec{\epsilon}: \|\vec{\epsilon}\|_p \leq t} \ell(m_f(\vec{x} + \vec{\epsilon}), y) \leq \ell(m_f(\vec{x}, y) - 2^{1/q}kt), \quad (4)$$

where q is defined such that $\|\cdot\|_q$ is the dual norm of $\|\cdot\|_p$.

Proof. The main idea behind the proof is to show that the Lipschitz constant of the network controls how much the margin can be influenced by an adversarial perturbation. Note that one can express the margin function given in Equation 3 as $m_f(\vec{x}, y) = m_{\mathbb{I}}(f(\vec{x}), y)$, where \mathbb{I} is the identity function. The Lipschitz constant of $m_{\mathbb{I}}$ with respect to its first argument when using the p -norm is $\max_{\vec{x}} \|\nabla_{\vec{x}} m_{\mathbb{I}}(\vec{x}, y)\|_q$ [12, p. 133]. The gradient of $m_{\mathbb{I}}$ is a vector with all elements set to zero, except for those corresponding to the largest and second largest components of \vec{x} . These components of the gradient take the values of 1 and -1 , respectively. Plugging these values into the

definition of vector p -norms, one arrives at a Lipschitz constant of $2^{1/q}$. From the composition property of Lipschitz functions, we can say that m_f is $(2^{1/q}k)$ -Lipschitz with respect to \vec{x} . The Lipschitz property of m_f can be used to bound the worst-case change in the output the margin function for a bounded change in the input, yielding

$$\ell\left(\min_{\vec{\epsilon}: \|\vec{\epsilon}\|_p < t} m_f(\vec{x} + \vec{\epsilon}), y\right) \leq \ell(m_f(\vec{x}, y) - 2^{1/q}kt). \quad (5)$$

From the decreasing monotonicity of ℓ , we have that

$$\max_{\vec{\epsilon}: \|\vec{\epsilon}\|_p < t} \ell(m_f(\vec{x} + \vec{\epsilon}), y) = \ell\left(\min_{\vec{\epsilon}: \|\vec{\epsilon}\|_p < t} m_f(\vec{x} + \vec{\epsilon}), y\right), \quad (6)$$

which concludes the proof. \square

This proposition bounds the worst-case change in loss for a single image in terms of prediction confidence, Lipschitz constant of the network, and the maximum allowable attack strength.

The relationship given in Proposition 1 is a more general form of the bound derived by [4], who consider only the Euclidean norm.

Proposition 1 can be extended to provide a non-trivial bound on the expected adversarial risk through the use of a held-out dataset and a simple application of McDiarmid's inequality.

Proposition 2. *If f is k -Lipschitz w.r.t. the p -norm, $\ell: \mathbb{R} \rightarrow [0, B]$ is a monotonically decreasing loss function, and $\{(\vec{x}_i, y_i) \sim \mathcal{D}\}_{i=1}^n$ is independent of f (i.e., held-out data), the following holds with probability at least $1 - \delta$:*

$$\tilde{R}_{p,t}^\ell(f) \leq \frac{1}{n} \sum_{i=1}^n \ell(m_f(\vec{x}_i, y_i) - 2^{1/q}kt) + B \sqrt{\frac{\ln(2/\delta)}{2n}} \quad (7)$$

where q is defined such that $\|\cdot\|_q$ is the dual norm of $\|\cdot\|_p$.

Proof. Constructing a mean over loss terms,

$$L = \frac{1}{n} \sum_{i=1}^n \ell(m_f(\vec{x}_i, y_i) - 2^{1/q}kt), \quad (8)$$

results in a sequence where each term is bounded by $\frac{B}{n}$, allowing McDiarmid's inequality to probabilistically bound the deviation from its expected,

$$\mathbb{P}(|L - \mathbb{E}[L]| > \gamma) \leq 2 \exp\left(\frac{-2n\gamma^2}{B^2}\right). \quad (9)$$

Setting δ equal to the right-hand side of Inequality 9 and solving for γ yields

$$\gamma = B \sqrt{\frac{\ln(2/\delta)}{2n}}. \quad (10)$$

Thus, we can say with confidence $1 - \delta$ that

$$\mathbb{E}[L] \leq L + B \sqrt{\frac{\ln(2/\delta)}{2n}}. \quad (11)$$

Applying Proposition 1 to each term of the summation, L , concludes the proof. \square

Proposition 2 extends the result of Proposition 1 from the loss on a single instance to the expected risk.

In practice, this means that a practitioner can bound the worst-case adversarial performance of their model based on its (non-adversarial) validation-set performance and its Lipschitz constant, both of which can be measured efficiently. As we show later, this can lead to non-vacuous bounds on error rate, which in turn could allow a user to deploy a model with provable confidence about its performance under adversarial attack—without the hassle and computational expense of instance-wise certification at run-time [9], [10].

III. ARCHITECTURES FOR PROVABLE ROBUSTNESS

The analysis in Section II motivates a high-level strategy for improving the adversarial robustness of neural networks: maximise the prediction margin while minimising the Lipschitz constant of the model. Several papers have proposed different methods for regularising the Lipschitz constant of a network, with various motivations, including improving robustness to adversarial examples [4], [13] and improving generalisation performance in the non-adversarial case [14].

We propose a strategy for modifying network architectures to make them more amenable to Lipschitz-based regularisers: splitting a single multi-class classification network into a collection of one-versus-all (OVA) classifiers that each produce a real-valued score. Unlike the conventional OVA method, where each component classifier is trained in isolation, the networks used in our approach are still trained jointly using a softmax composed with the cross entropy loss function. There are two requirements for this OVA scheme to have a benefit: each of the simpler binary classification subproblems must be solvable by a network with a smaller Lipschitz constant, and the Lipschitz constant of the multi-classifier system must grow slowly with the number of classes. [14] show that the Lipschitz constant is related to model capacity, so the subnetwork associated with each class should be able to achieve high accuracy with a smaller Lipschitz constant than a conventional multi-class classification network. For the second requirement, consider the vector-valued function,

$$f(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_C(\vec{x})], \quad (12)$$

where C is the number of classes, and f_i is k_i -Lipschitz. We have from the Lipschitz property of each f_i that

$$\|f(\vec{x}) - f(\vec{x} + \vec{v})\|_p \leq \| [k_1\|\vec{v}\|_p, k_2\|\vec{v}\|_p, \dots, k_C\|\vec{v}\|_p] \|_p \quad (13)$$

$$= \|\vec{v}\|_p \| [k_1, k_2, \dots, k_C] \|_p, \quad (14)$$

from which we can deduce that the Lipschitz constant of the one-versus-all classifier is the ℓ_p norm of the vector of Lipschitz constants corresponding to each binary classifier. In the case of the ∞ -norm, the largest Lipschitz constant associated with a single binary classifier dictates the Lipschitz constant of the entire OVA classifier. From this, we can conclude that the second requirement is satisfied.

A. Lipschitz Regularization Training

We investigate two approaches to controlling the Lipschitz constant of neural networks. The first approach we use is to add the bound on the Lipschitz constant as a regularisation term to the objective function, resulting in

$$\frac{1}{n} \sum_{i=1}^n \ell(f(\vec{x}_i), y_i) + \prod_{l=1}^d \|W_l\|_p, \quad (15)$$

where d is the number of layers in the network and $\|\cdot\|_p$ is the operator norm induced by the vector p -norm. In the case where p is two, the operator norm is the largest singular value of the matrix (i.e., the spectral norm). For $p = \infty$, it is the maximum absolute row sum norm [14],

$$\|W\|_\infty = \max_i \sum_j |W_{i,j}|. \quad (16)$$

IV. EXPERIMENTS

This section presents the results of numerical experiments that demonstrate the tightness of the bounds presented in Section II and provides evidence that the architecture proposed in Section III is inherently easier to optimise for provable robustness than conventional network architectures. The models used in these experiments were implemented using Keras ¹, and the adversarial attacks were performed using the CleverHans toolkit ².

A. Tightness of the Bound

The bound given in Proposition 2 provides a way to estimate the worst-case performance of a model when under the influence of an adversary. In order to validate this bound empirically, we train linear support vector machines with different levels of ℓ_2 regularisation on the MNIST dataset of hand-written digits. In the case of linear SVMs, the optimisation problem solved by iterative gradient-based attacks, such as the projected gradient descent method of [5], are convex and can therefore be solved globally. This means the the empirical adversarial risk can be computed exactly. Plots indicating the tightness of the bound for linear SVMs are given in Figure 1. These were generated by training models on the first 50,000 instances of the training set, using the other 10,000 training instances as the held-out data required for computing the bound, and using the PGD attack when evaluating the network on the test data. These plots confirm that the bound proposed in Proposition 2 is non-vacuous and has the potential to be useful in practice.

B. Provable Robustness

We first experiment on MNIST to determining whether our proposed OVA networks achieve better provable robustness than conventional convolutional neural networks. To control for the potentially confounding factor of model capacity, a series of networks with different widths are trained. We define the width of a conventional convolutional network as the number of

¹<https://github.com/keras-team/keras>

²<https://github.com/tensorflow/cleverhans>

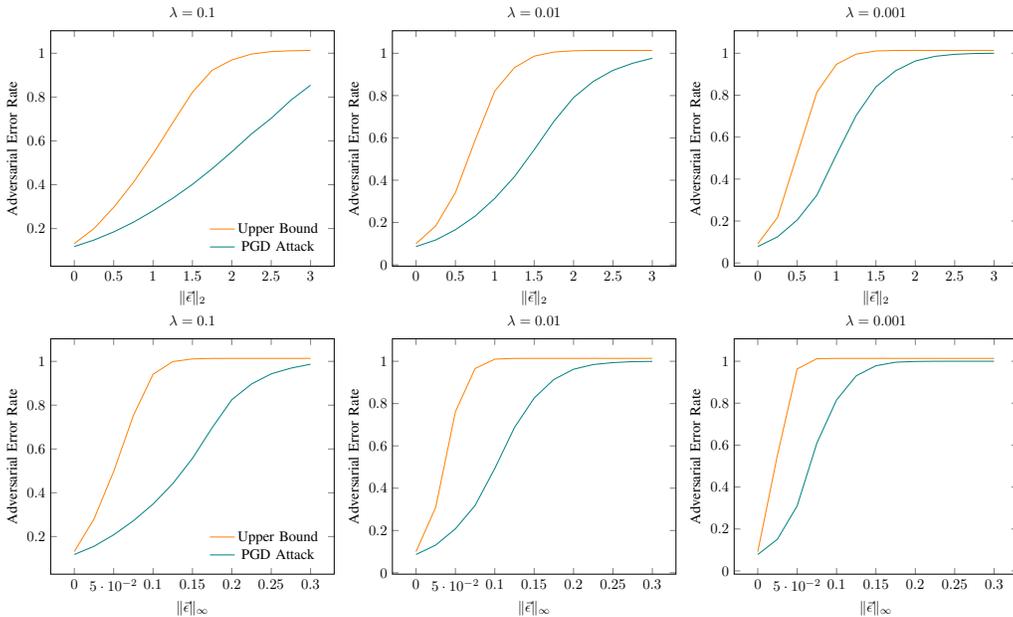


Fig. 1. Plots demonstrating the relationship between the provable upper bound on adversarial risk, and the actual misclassification rate on the test set under adversarial attack. Linear SVM recognition on MNIST with ℓ_2 (top row) and ℓ_∞ (bottom row) threat models and regularization strength λ .

feature maps produced by the first convolutional layer. For OVA networks, the width is the number of feature maps produced by the first layer in a single binary classifier, multiplied by the number of binary classifiers. For both network types, the chosen architectures contain two convolutional layers, the second of which has twice the number of feature maps as the first. Each convolutional layer contains 5×5 kernels, rectified linear unit activation functions, and is followed by a 2×2 max pooling layer. After the convolutional layers are two fully connected layers: one with 128 hidden units, and another with either 10 units (for conventional networks), or one unit (for OVA networks).

Figure 2 shows how the number of model parameters impacts the provable adversarial robustness for threat models based on the ℓ_2 and ℓ_∞ norms. The models in these plots are regularised using the Lipschitz penalty method proposed in Section III. These figures show that: (1) Regularised OVA networks exhibit superior provable robustness compared to regularized conventional CNNs at comparable model sizes, (2) The magnitude of this margin becomes more pronounced as model size increases, (3) All methods have low error rate for unperturbed examples (left plots).

To investigate how well OVANets scale to larger networks and more challenging datasets, additional experiments are run on the CIFAR-10 dataset, using VGG-style networks [15] as the base architecture. The baseline CNN uses the VGG11 architecture, and each subnetwork of the OVANet architecture is a VGG11 network with half the number of feature maps in each layer. Table I provides probabilistic (95% confidence) bounds on the worst-case adversarial error rate using Proposition 2. Table II shows the corresponding provable robustness results for SVHN benchmark. From the results we can see that: (1)

TABLE I
BOUNDS ON THE ERROR RATE FOR VGG MODELS TRAINED ON CIFAR-10. THE BOUNDS WERE COMPUTED WITH PROPOSITION 2 AT THE 95% CONFIDENCE LEVEL AND THE ℓ_2 THREAT MODEL.

Model	λ	Clean	Perturbation Size (ℓ_2)			
			1/255	2/255	3/255	4/255
VGG11-CNN	0	14.50	100.00	100.00	100.00	100.00
	0.0001	14.22	47.61	79.22	95.87	100.00
	0.0005	16.00	29.00	42.74	56.49	69.84
	0.001	17.64	26.80	35.60	44.75	53.66
VGG11-OVA	0	17.18	100.00	100.00	100.00	100.00
	0.0001	15.58	44.54	73.49	93.11	99.99
	0.0005	15.86	27.68	39.01	51.85	63.68
	0.001	17.09	25.00	32.54	40.35	48.53

Lipschitz penalty training improves the adversarial error rate for both vanilla VGG11 and VGG11-OVANet (performance improves with λ); (2) VGG11-OVANet generally has superior provable robustness compared to vanilla VGG11 for corresponding regularisation strength, especially for strong attacks. (3) Meanwhile, regularized OVANet achieves comparable results to a regularized CNN in terms of clean data performance.

V. CONCLUSIONS

This paper presents a p -norm-agnostic theoretical analysis of provable adversarial robustness via Lipschitz regularisation. A new architecture, the OVA network, is proposed, motivated by insights of how Lipschitz constants can be bounded for different architecture design choices. It is shown that OVA networks achieve similar empirical performance to conventional neural networks but, as network size increases, OVA networks are able to achieve significantly better certifiable robustness. This

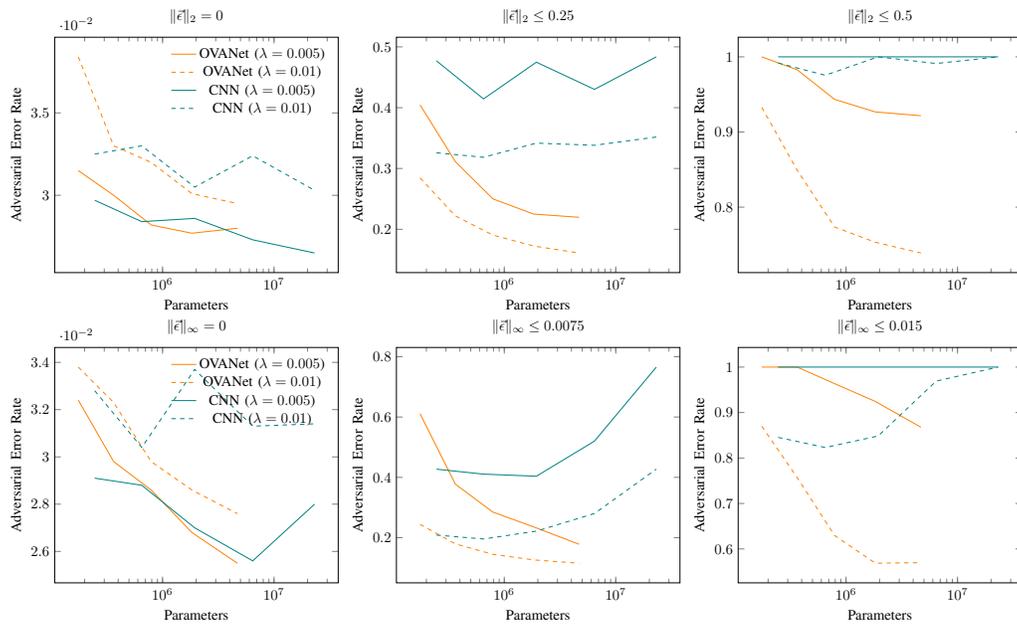


Fig. 2. Comparison of provable adversarial risk for conventional CNN versus OVANet trained with Lipschitz penalty regularization over a range of model sizes. The ℓ_2 (top row) and ℓ_∞ (bottom row) Lipschitz constants are used for regularisation and computing the bound. OVANet shows superior provable robustness, especially at larger model sizes.

TABLE II
BOUNDS ON THE ERROR RATE FOR VGG MODELS TRAINED ON SVHN.
THE BOUNDS WERE COMPUTED WITH PROPOSITION 2 AT THE 95%
CONFIDENCE LEVEL AND THE ℓ_2 THREAT MODEL.

Model	λ	Clean	Perturbation Size (ℓ_2)			
			1/255	2/255	3/255	4/255
VGG11-CNN	0	7.29	100.00	100.00	100.00	100.00
	0.0001	7.15	13.16	21.86	33.59	47.10
	0.0005	8.46	11.38	14.84	19.33	24.49
	0.001	9.41	11.76	14.95	17.76	21.48
VGG11-OVA	0	7.69	100.00	100.00	100.00	100.00
	0.0001	7.45	12.10	19.19	28.38	39.17
	0.0005	8.25	10.64	13.02	16.11	19.82
	0.001	9.02	10.92	12.86	15.05	17.81

is a useful result for practitioners, who can use a Lipschitz regulariser and our bound in order to train models with a certifiable level of robustness against adversarial attack.

REFERENCES

[1] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *arXiv:1312.6199 [Cs]*, 2014.
 [2] N. Carlini and D. Wagner, "Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods," in *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, ser. AISeC '17. New York, NY, USA: ACM, 2017, pp. 3–14.
 [3] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples," in *International Conference on Machine Learning*, Jul. 2018, pp. 274–283.
 [4] Y. Tsuzuku, I. Sato, and M. Sugiyama, "Lipschitz-Margin Training: Scalable Certification of Perturbation Invariance for Deep Neural Networks," in *Advances in Neural Information Processing Systems*, 2018.

[5] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards Deep Learning Models Resistant to Adversarial Attacks," in *International Conference on Learning Representations*, Feb. 2018.
 [6] F. Farnia, J. M. Zhang, and D. Tse, "Generalizable Adversarial Training via Spectral Normalization," *arXiv:1811.07457 [cs, stat]*, Nov. 2018.
 [7] P. L. Bartlett, D. J. Foster, and M. J. Telgarsky, "Spectrally-normalized margin bounds for neural networks," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., 2017, pp. 6240–6249.
 [8] T. Huster, C.-Y. J. Chiang, and R. Chadha, "Limitations of the Lipschitz constant as a defense against adversarial examples," *arXiv:1807.09705*, Jul. 2018.
 [9] T.-W. Weng, H. Zhang, H. Chen, Z. Song, C.-J. Hsieh, D. Boning, I. S. Dhillon, and L. Daniel, "Towards fast computation of certified robustness for relu networks," in *International Conference on Machine Learning*, 2018.
 [10] E. Wong and J. Z. Kolter, "Provable defenses against adversarial examples via the convex outer adversarial polytope," in *International Conference on Machine Learning*, 2018.
 [11] K. Crammer and Y. Singer, "On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines," *Journal of Machine Learning Research*, vol. 2, no. Dec, pp. 265–292, 2001.
 [12] S. Shalev-Shwartz, "Online Learning and Online Convex Optimization," *Foundations and Trends® in Machine Learning*, vol. 4, no. 2, pp. 107–194, Mar. 2012.
 [13] M. Cisse, P. Bojanowski, E. Grave, Y. Dauphin, and N. Usunier, "Parseval Networks: Improving Robustness to Adversarial Examples," in *International Conference on Machine Learning*, Jul. 2017, pp. 854–863.
 [14] H. Gouk, E. Frank, B. Pfahringer, and M. Cree, "Regularisation of Neural Networks by Enforcing Lipschitz Continuity," *arXiv preprint arXiv:1804.04368*, 2018.
 [15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

Tail of Distribution GAN (TailGAN): Generative-Adversarial-Network-Based Boundary Formation

Nikolaos Dionelis, Mehrdad Yaghoobi, Sotirios A. Tsaftaris

The University of Edinburgh

Edinburgh, UK

Contact email: nikolaos.dionelis@ed.ac.uk

Abstract—Generative Adversarial Networks (GANs) are a powerful methodology and can be used for unsupervised anomaly detection, where current techniques have limitations such as the accurate and robust detection of anomalies near the tail of a distribution. GANs generally do not guarantee the existence of a probability density and are susceptible to mode collapse, while few GANs use likelihood to reduce mode collapse. In this paper, we create a GAN-based tail formation model for anomaly detection, the Tail of distribution GAN (TailGAN), to generate samples on the tail of the data distribution and detect anomalies near the support boundary. Using TailGAN, we use maximum entropy regularization and leverage GANs for anomaly detection. Using GANs that learn the probability of the underlying distribution has advantages in improving the anomaly detection methodology by allowing us to devise a generator for boundary samples, and use this model to characterize anomalies. TailGAN addresses supports with disjoint components and achieves competitive performance on images. We evaluate TailGAN for identifying Out-of-Distribution (OoD) data and its performance evaluated on MNIST, CIFAR-10, Baggage X-Ray, and OoD data shows competitiveness compared to methods from the literature.

Index Terms—Anomaly detection, Generative models

I. INTRODUCTION

Generative Adversarial Networks (GANs) can capture complex data with applications in computer vision and achieve state-of-the-art image synthesis performance. Recently, GANs have been used for anomaly detection (AD) which is critical in security (contraband detection). GANs succeed convergence in distribution metrics, but suffer from mode collapse, learn distributions of low support, and do not guarantee the existence of a probability density making generalization with likelihood impossible [1, 2]. Many AD techniques perform well in low dimensions; however, there is a lack of effective methods for high-dimensional spaces, e.g. images. Important tasks are reducing false negatives and false alarms, providing boundaries for inference of within-distribution and Out-of-Distribution (OoD), and detecting anomalies near low probability regions. Unsupervised AD is examined since anomalies are not known in advance. The normal class is learned and anomalies are detected by deviating from this model. Researchers use the leave-one-out evaluation, $(K + 1)$ classes, K classes for normality, and the leave-out class for anomaly, chosen arbitrarily. This does not use the complement of the distribution support, and real-world anomalies are not confined to a finite set.

In this paper, we create a GAN-based model, the Tail of distribution GAN (TailGAN), to generate samples on the low

probability regions of the normal data distribution and detect anomalies close to the support boundary. Using TailGAN, we leverage GANs for OoD sample detection and perform sample generation on the tail using an objective cost function that forces the generated samples to lie on the boundary while optimizing an entropy-regularized loss to stabilize training. The authors of this paper have recently proposed an invertible-residual-network-based generator, the Boundary of Distribution Support Generator (BDSG) [5], for AD using invertible residual networks [6, 7]. In this paper, we include adversarial training and we create GANs to generate samples on the low probability regions of the data distribution and detect anomalies near the support boundary. We create a model to generate abnormal samples on the boundary of the normal data distribution. Our contribution is the creation of a GAN-based boundary formation model and we use GANs, such as Prescribed GAN (PresGAN) and FlowGAN [9, 10], that learn the probability of the underlying distribution and generate samples with high likelihoods. TailGAN improves the detection of anomalies and GANs that learn the probability density of the underlying multimodal distribution can improve the AD methodology by allowing us to create a generator for boundary samples and use this to characterize anomalies.

II. RELATED WORK ON AD USING GANS

A. GANs Using a Reconstruction-Based Anomaly Score

AnoGAN performs unsupervised learning to detect anomalies by learning the manifold of normal anatomical variability. During inference, it scores image patches indicating their fit into the learned distribution [11, 12]. In contrast to AnoGAN, Efficient GAN-Based AD (EGBAD) jointly learns an encoder and a generative model to eliminate the procedure of computing the latent representation of a queried test sample [13]. Its anomaly score combines discriminator and reconstruction losses. GANomaly learns the generation of the data and the inference of the latent space, \mathbf{z} . It uses an encoder-decoder-encoder in the generator, and minimizing the distance between the vectors in \mathbf{z} aids in learning the data distribution [14].

B. GANs Performing Sample Generation on the Boundary

The GAN loss learns the mass of the distribution but to perform AD, we look at the boundary. The GAN optimization problem is given by $\operatorname{argmin}_{\theta_g} \operatorname{dist}(p_{\mathbf{x}}(\mathbf{x}), p_g(\mathbf{x}))$, where the distance metric, $\operatorname{dist}(\cdot, \cdot)$, takes the specific form

$\operatorname{argmax}_{\theta_d} f(D, p_x(\mathbf{x}), p_g(\mathbf{x}))$. For example, $\operatorname{dist}(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_\infty = \max_i |x_i - y_i|$ and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$. The GAN loss is

$$\min_G \max_D \mathbb{E}_x[\log(D(\mathbf{x}))] + \mathbb{E}_z[\log(1 - D(G(\mathbf{z})))] \quad (1)$$

where $\mathbf{z} \sim p_z(\mathbf{z}) = N(\mathbf{0}, \mathbf{I})$, $\mathbf{x} \sim p_x(\mathbf{x})$, and $G(\mathbf{z}) \sim p_g(\mathbf{x})$. To generate samples on the distribution tail, MinLGAN uses minimum likelihood while FenceGAN changes (1). In contrast to the traditional GAN [17, 18], FenceGAN estimates the distance between $p_g(\mathbf{x})$ and the tail of $p_x(\mathbf{x})$. Its limitations are sampling complexity, the parallel estimation of $p_x(\mathbf{x})$ and the tail of $p_x(\mathbf{x})$, and disconnected boundary generation.

III. THE PROPOSED TAILGAN

Section II.B has presented GANs for AD that perform sample generation on the support boundary of the normal data distribution. In this section, we present the TailGAN model for sample generation on the distribution’s tail and for AD.

A. Boundary Generation: Improvement on Leave-One-Out

Before proceeding to explain TailGAN, it is important to first motivate the need to perform accurate sample generation on the tail. The leave-one-out evaluation, which can be restrictive for evaluating AD models, uses the disconnected components of the underlying multimodal distribution. For any annotated dataset with $K+1$ classes, the classification problem creates clusters and the decision criterion is a boundary for classification [20]. In high-dimensional spaces, classes form clusters and are disconnected components. Each cluster can have more than one modes. The disconnected components of the underlying distribution are usually known during the evaluation of the model while the modes of the distribution are not. Now, we denote a sample generated on the distribution’s tail by $T(\mathbf{z})$, where $\mathbf{z} \sim N(\mathbf{0}, \mathbf{I})$ is the latent space, and using the l_p -norm, $\|\cdot\|_p$, the clustering algorithm is given by

$$k(T(\mathbf{z})) = \operatorname{argmin}_{i=1, \dots, K} \operatorname{dist}(T(\mathbf{z}), \mathbf{x}_i) \quad (2)$$

$$\operatorname{dist}(T(\mathbf{z}), \mathbf{x}_i) = \min_{j=1, \dots, L} \|T(\mathbf{z}) - \mathbf{x}_{i,j}\|_p \quad (3)$$

$$R(T(\mathbf{z}), k) = \min_{j=1, \dots, L} \|T(\mathbf{z}) - \mathbf{x}_{k,j}\|_p \quad (4)$$

$$R(T(\mathbf{z}), k) < \min_{i \neq k, i=1, \dots, K} \min_{j=1, \dots, L} \|\mathbf{x}_{i,j} - \mathbf{x}_{k,j}\|_p \quad (5)$$

where we use K clusters from the leave-one-out methodology, L samples from every class/cluster, and $\mathbf{x}_{i,j} \in \mathbb{R}^d$ is the j -th sample of class i . With our boundary model, we can decide support membership to improve AD and also weight misses and false alarms. We can also generate anomalies, including adversarial anomalies, and the inequality presented in (5) states that the model’s generated samples are closer to the relevant class than any sample from any other class. Using (4) and (5), our boundary model improves the leave-one-out evaluation methodology by the margin given by $|R(T(\mathbf{z}), k) - \min_{i \neq k, i=1, \dots, K} \min_{j=1, \dots, L} \|\mathbf{x}_{i,j} - \mathbf{x}_{k,j}\|_p|$.

B. Framework for Sample Generation on the Tail

In this section, we develop our model, TailGAN, to detect anomalies near the low probability regions of the data distribution, i.e. strong anomalies. GANs generally do not guarantee

the existence of a probability density and we use the recently developed PresGAN and FlowGAN models. We leverage such models for sample generation on the tail using two steps. The first step is to train either PresGAN or FlowGAN to learn the “normal” distribution, $G(\mathbf{z}) \sim p_g(\mathbf{x})$. The random variable \mathbf{z} follows a standard Gaussian distribution, $\mathbf{z} \sim N(\mathbf{0}, \mathbf{I})$, and the mapping from the latent space, \mathbf{z} , to the data space, $\mathbf{x} \in \mathbb{R}^d$, is given by $G(\mathbf{z})$. The second step is to train a generator, $T(\mathbf{z})$, to perform sample generation on the tail by minimizing

$$L_{tot}(\theta, \mathbf{z}, \mathbf{x}, G) = w_{pr} L_{pr}(\theta, \mathbf{z}, G) + w_d L_d(\theta, \mathbf{z}, \mathbf{x}) + w_e L_e(\theta, \mathbf{z}, G) + w_{sc} L_{sc}(\theta, \mathbf{z}) \quad (6)$$

where the total cost function, L_{tot} , comprises the probability cost, L_{pr} , the distance loss, L_d , the maximum-entropy cost, L_e , and the scattering loss, L_{sc} . The total cost in (6) comprises four terms. The probability cost penalizes probability density to find the tail of the data distribution while the distance loss penalizes large distance from normality using the distance from a point to a set. The maximum-entropy loss is for the dispersion of the samples [9, 1], and the scattering cost, L_{sc} , is defined by the ratio of the distances in the \mathbf{z} and \mathbf{x} spaces to address mode collapse. Hence, L_{tot} in (6) is given by

$$\frac{1}{N} \sum_{i=1}^N \left[w_{pr} p_g(T(\mathbf{z}_i; \theta)) + w_d \min_{j=1}^M \|T(\mathbf{z}_i; \theta) - \mathbf{x}_j\|_p + w_e p_g(T(\mathbf{z}_i; \theta)) \log(p_g(T(\mathbf{z}_i; \theta))) + w_{sc} \frac{1}{N-1} \sum_{j=1, j \neq i}^N \frac{\|\mathbf{z}_i - \mathbf{z}_j\|_p^q}{\|T(\mathbf{z}_i; \theta) - T(\mathbf{z}_j; \theta)\|_p^q} \right] \quad (7)$$

where we leverage the tradeoff between probability and distance in the first two terms of the loss, i.e. L_{pr} and L_d , and where the model probability, $p_g(T(\mathbf{z}_i; \theta))$, is given by

$$p_z(G^{-1}(T(\mathbf{z}_i; \theta))) |\det \mathbf{J}_G(T(\mathbf{z}_i; \theta))|^{-1} = \exp(\log(p_z(G^{-1}(T(\mathbf{z}_i; \theta)))) - \log(|\det \mathbf{J}_G(T(\mathbf{z}_i; \theta))|)) \quad (8)$$

where $\log(p_z(G^{-1}(T(\mathbf{z}))))$ and $\log(|\det \mathbf{J}_G(T(\mathbf{z}))|)$ are estimated by an invertible GAN model such as FlowGAN.

The parameters of the generator $T(\mathbf{z})$, θ , are obtained by running Gradient Descent on L_{tot} , which can decrease to zero and is written in terms of the sample size, M , and the batch size, $N \leq M$. The distance term of the loss in (6), $L_d(\theta, \mathbf{z}, \mathbf{x})$, depends on the training data, \mathbf{x} . This distance term could use $G(\mathbf{z})$ instead of \mathbf{x} and in (7), our distance metric is defined using M by $\operatorname{dist}(T(\mathbf{z}_i), \mathbf{x}) = \min_{j=1, \dots, M} \|T(\mathbf{z}_i) - \mathbf{x}_j\|_p$.

The loss in (7) uses the l_p -norm, $\|\cdot\|_p$, and the scattering loss, L_{sc} , is based on the l_p -norm to power q , $\|\cdot\|_p^q$, where p and q are real numbers and $p, q \geq 1$. In (7), the weight w_{pr} is equal to 1 and w_d, w_e , and w_{sc} are hyperparameters. In (6)-(8), the gradient of L_{tot} with respect to $T(\mathbf{z})$ is well-defined and the change of variables formula in (8) has been used in a signal processing and nonlinear filtering algorithm in [22].

IV. EVALUATION OF TAILGAN

We evaluate the TailGAN model using (i) algorithm convergence criteria such as the value of the objective cost function



Fig. 1. Generated samples of handwritten digits when we train PresGAN on MNIST using the PresGAN hyperparameter $\lambda = 0.0002$, at 32 epochs.

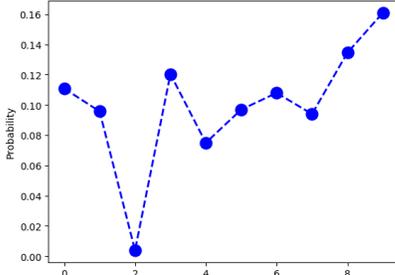


Fig. 2. Probability of occurrence of the generated images plotted against the disconnected component index using (2) and (3) when PresGAN is trained on MNIST using leave-one-out evaluation. The anomaly class is digit 2.

in (7) and the value of the distance cost, L_d , which is the distance from a point to a set and is based on the l_p -norm and the minimum operator, and (ii) the Area Under the Receiver Operating Characteristics Curve (AUROC) and the Area Under the Precision-Recall Curve (AUPRC). Experiments are performed on datasets of increasing complexity, MNIST, CIFAR, and Baggage X-Ray. In contrast to [2], we use the leave-one-out evaluation methodology and the detection of abnormal OoD data. The leave-one-out evaluation that uses the leave-out class as the anomaly leads to multimodal distributions with a support with disjoint components. The boundary of the support of the data distribution is defined using the threshold ϵ .

TailGAN performs efficient sample generation on the tail of the distribution obviating the rarity sampling complexity problem, not requiring importance sampling [20]. For distributions with disconnected components, the TailGAN model achieves better performance than the convex hull extrema points.

A. Implementation and Training of TailGAN

We implement TailGAN in PyTorch and a vectorized implementation of the individual terms of the cost function in (7) has been created. We evaluate TailGAN using $p = q = 2$ and because the choice of distance metric is important, $p = q = 1$ or $p = 2$ and $q = 1$ could be used. We use $p_g(\mathbf{x})$ from PresGAN which computes the entropy to address mode collapse. The first connection between TailGAN and our chosen base mode is $p_g(\mathbf{x})$ while the second connection is model initialization, $\theta_{t0} = \theta_g$, where θ_{t0} are the parameters of the generator $T(\mathbf{z})$ at the start of training and θ_g are the parameters of $G(\mathbf{z})$.

Using $\theta_{t0} = \theta_g$, $T(\mathbf{z})$ is trained to perform sample generation on the tail of the normal data distribution by starting from within the distribution, and this differs from the encoder-based initialization used in [2]. On the contrary, using random initial-

MNIST	L_{tot}	L_d	L_{sc}
MNIST Digits 0-9	4.73	471.96	0.82
Fashion-MNIST	14.91	1489.70	0.82
KMNIST	14.78	1476.81	0.82

TABLE I
EVALUATION OF TAILGAN COMPARING NORMALITY WITH ANOMALY CASES USING ALGORITHM CONVERGENCE CRITERIA, TOTAL LOSS AND DISTANCE LOSS. ANOMALY CASES: FASHION-MNIST, KMNIST.

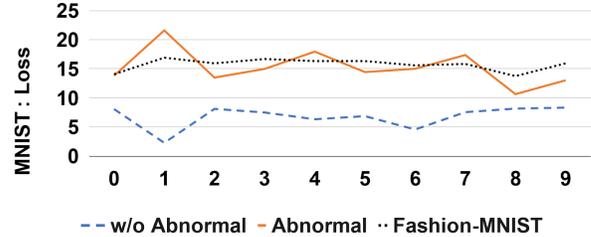


Fig. 3. Leave-one-out evaluation of TailGAN trained on MNIST using L_{tot} . The anomaly cases are the leave-out class and data from Fashion-MNIST.

ization for θ , $T(\mathbf{z})$ is trained to perform sample generation on the tail by starting from outside the distribution. To compute the probability density, $p_g(\mathbf{x})$, in (7) from the entropy which is estimated by PresGAN, we use the Lambert $W(\cdot)$ function and Newton's iterations. We evaluate TailGAN by first performing density estimation and then training $T(\mathbf{z})$ using convolutional layers with batch normalization, minimizing (6) and (7).

B. Evaluation of TailGAN on MNIST Data

We first train PresGAN on MNIST until convergence using the leave-one-out methodology and the detection of abnormal OoD data. Next, we train TailGAN applying the objective cost function in (7). We examine different values for the batch size, N , and we use the entire training set for the sample size, M . We create $T(\mathbf{z})$ using convolutional networks and we examine different architectures such as feed-forward and residual.

Figure 1 shows the generated samples at 32 epochs when we train and use a modified version of PresGAN and the PresGAN hyperparameter $\lambda = 0.0002$. As qualitative evaluation and visual measure, the MNIST canvas with the generated images in Fig. 1 shows that our chosen base model is trained to create realistic images of handwritten digits. All the digits from 0 to 9 are present in the canvas. Figure 2 depicts the probability of occurrence of the generated images against the disconnected component index when PresGAN is trained on MNIST using the leave-one-out evaluation, when the anomaly class is digit 2. The frequency of the generated data is computed using (2) and (3) based on the clustering algorithm in Sec. III.A.

The convergence of the values of the proposed cost function in (7) is examined. We train TailGAN on MNIST image data until convergence and we observe that all the individual terms of the proposed objective cost function, i.e. L_{pr} , L_d , L_e , and L_{sc} , decrease over training epochs achieving convergence.

For the evaluation of TailGAN, Table I shows the algorithm convergence criteria in (6)-(8) produced by TailGAN trained on MNIST data. TailGAN is evaluated using MNIST data as the normal class and abnormal data originating from Fashion-MNIST and KMNIST. The values of the objective cost function,

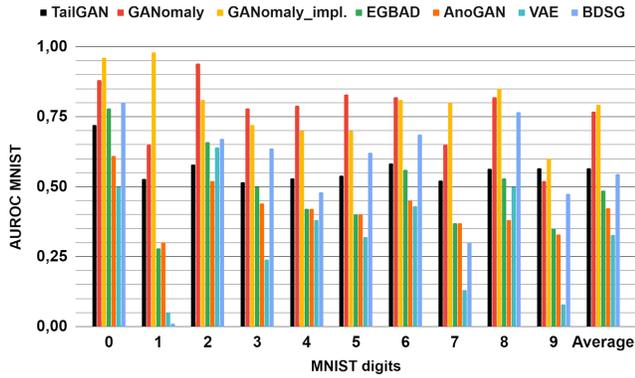


Fig. 4. Evaluation of TailGAN using AUROC on MNIST data.

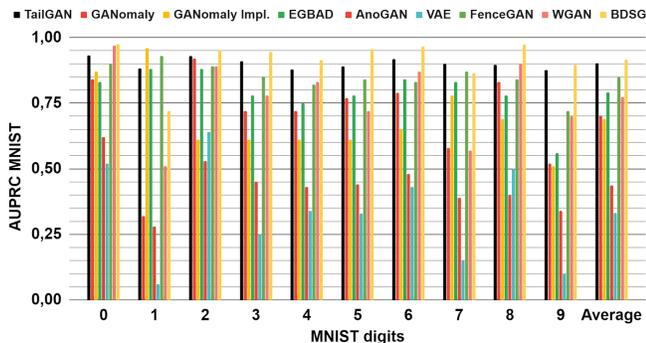


Fig. 5. Evaluation using AUPRC for AD on MNIST image data.

total loss, and distance loss, L_d , for abnormal OoD data are higher than the corresponding values for the normal data. The OoD anomalies are from Fashion-MNIST and KMNIST, and the total and distance losses are indicators of anomalies.

Using the leave-one-out evaluation on MNIST, we compute the cost function in (7), the total cost, the distance loss, and the scattering cost of $T(\mathbf{z})$ of TailGAN. Figure 3 depicts the total loss, L_{tot} , for every MNIST digit for three different cases: (i) without the abnormal digit, (ii) the abnormal digit, and (iii) Fashion-MNIST. The anomaly cases are the abnormal leave-out digit and Fashion-MNIST data. The algorithm convergence criteria, L_{tot} and L_d , deviate from normality for the abnormal digits and the OoD cases as they are higher compared to the corresponding normal case values. Comparing TailGAN to the GANomaly and FenceGAN baselines trained on MNIST and evaluated on Fashion-MNIST, the l_2 -norm distance loss in the \mathbf{x} space is 3.1, 1.7, and 1.1 times as much compared to the corresponding values for the normal case for TailGAN (i.e. Table I), GANomaly, and FenceGAN [18], respectively.

Figures 4 and 5 present the AUROC and AUPRC scores, respectively, for the evaluation of TailGAN on MNIST data. The performance of TailGAN is examined using the leave-one-out evaluation methodology. The proposed TailGAN model is evaluated using the AUROC and AUPRC metrics and is compared to several baselines, GANomaly, the implementation of GANomaly (GANomaly Impl.), EGBAD, AnoGAN, VAE, FenceGAN, WGAN, and BSG [5]. Compared to the baselines, TailGAN shows competitive results in AUROC and AUPRC. For the anomaly score during inference, the TailGAN



Fig. 6. Generated samples when we train PresGAN on CIFAR-10 images.

CIFAR-10	L_{tot}	L_d	L_{sc}
CIFAR-10	2.22	217.72	4.60
CIFAR-100	5.21	516.49	4.60
SVHN	7.76	771.02	4.60
STL-10	5.74	569.21	4.60
CelebA	6.31	626.26	4.60
Baggage X-Ray	51.67	5161.90	4.60

TABLE II

EVALUATION OF TAILGAN COMPARING NORMAL WITH ABNORMAL CASES USING ALGORITHM CONVERGENCE CRITERIA, TOTAL LOSS AND DISTANCE LOSS. NORMALITY: CIFAR-10 CLASSES 0-9. ANOMALY CASES: CIFAR-100, SVHN, STL-10, CELEBA, BAGGAGE X-RAY.

model uses the estimated probability density, the first term of the objective cost function presented in (7) which addresses if a sample is within or out of the distribution support, and the second term in (7) which computes l_p -norm distances.

C. Evaluation of TailGAN on CIFAR-10 Data

To scale up the dimensions of the problem, we use the CIFAR-10 dataset and we hence go from 28×28 dimensions of MNIST to $3 \times 32 \times 32$ dimensions of CIFAR. First, we train PresGAN for density estimation until convergence on CIFAR data and, then, we train TailGAN using convolutional neural networks with batch normalization. We train TailGAN using the entire training set for M and our aim is to use TailGAN to accurately detect atypical, aberrant, abnormal samples.

Figure 6 shows the generated samples when we use a modified version of PresGAN and the PresGAN hyperparameter $\lambda = 0.0002$. As qualitative evaluation, the CIFAR-10 canvas with the generated images in Fig. 6 shows that our chosen base model is trained to create realistic images. All the classes are present in the canvas. Next, we train the proposed TailGAN by minimizing the cost function presented in (6)-(8). TailGAN achieves convergence on CIFAR-10 and L_{pr} , L_d , L_e , and L_{sc} in (6) and (7) decrease over iterations, epochs, and time.

For model evaluation during inference, Table II compares the values of the cost terms for the normal class, CIFAR-10 data, with the corresponding values of the loss terms for the abnormal OoD data, CIFAR-100, SVHN, STL-10, CelebA, and Baggage X-Ray. Table II shows that the distance loss is an indicator of anomalies and of the anomaly score, and both the total and distance losses are indicators of anomalies. The distance loss and the cost function values, L_d and L_{tot} , are higher for the OoD data than the corresponding values for normality: $L_{tot} = 2.22$ for normal data from CIFAR-10 and $L_{tot} = 6.31$ for abnormal data from CelebA. Comparing



Fig. 7. (a) Normal class image from the Baggage X-Ray dataset. (b) Anomaly class image from the Baggage X-Ray dataset containing knives and bottles.

Baggage X-Ray	L_{tot}	L_d	L_{sc}
Baggage X-Ray Normal	3.21	319.22	2.83
Baggage X-Ray Abnormal	7.45	741.39	2.83
CIFAR-10	12.77	1274.32	2.83
CIFAR-100	13.14	1309.87	2.83
SVHN	9.87	983.72	2.83
STL-10	17.11	1706.81	2.83

TABLE III

EVALUATION OF TAILGAN ON BAGGAGE X-RAY DATA COMPARING NORMALITY WITH ABNORMAL OOD CASES USING L_{tot} FROM (7).

TailGAN with GANomaly and FenceGAN trained on CIFAR-10 (Normal) and evaluated on CIFAR-100 (Abnormal), the l_2 -norm distance loss in the \mathbf{x} space is 2.3, 1.1, and 1.1 times as much compared to the corresponding values for normality for TailGAN (i.e. Table II), GANomaly, and FenceGAN.

D. Evaluation of TailGAN on Baggage X-Ray Data

In this section, we evaluate TailGAN on Baggage X-Ray data. Figure 7 shows two example images, one from the normal class and one from the anomaly class, from the Baggage X-Ray dataset, where the abnormal image contains 2 knives and 3 bottles. Table III uses Baggage X-Ray data and compares the values of the losses for the normal class with the values of the losses for the abnormal OoD data. The distance loss and the cost function values, L_d and L_{tot} , are higher for the Baggage X-Ray abnormal OoD data than the corresponding values for the normal Baggage X-Ray data, i.e. $L_{tot} = 3.21$ for normal class data and $L_{tot} = 7.45$ for abnormal OoD image data.

V. CONCLUSION

In this paper, we have proposed TailGAN to perform AD and sample generation on the tail of the distribution of typical samples. The proposed TailGAN uses adversarial training and GANs, and minimizes the objective cost function in (7). Using GANs that can explicitly compute the probability density, we perform sample generation on the tail of the data distribution, we address multimodal distributions with disconnected components, and we also address the mode collapse problem. The main evaluation outcomes on MNIST, CIFAR-10, Baggage X-Ray, and OoD data using the leave-one-out evaluation show that TailGAN achieves competitive AD performance.

VI. ACKNOWLEDGMENT

This work was supported by the Engineering and Physical Sciences Research Council of the UK (EPSRC) Grant number

EP/S000631/1 and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing.

REFERENCES

- [1] E. Abbasnejad, J. Shi, A. van den Hengel, and L. Liu. A Generative Adversarial Density Estimator. In Proc. IEEE CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.
- [2] L. Ruff, R. Vandermeulen, N. Görnitz, L. Deecke, S. Siddiqui, A. Binder, E. Müller, and M. Kloft. Deep One-Class Classification. In Proc. 35th International Conference on Machine Learning, Stockholm, 2018.
- [3] L. Deecke, R. Vandermeulen, L. Ruff, S. Mandt, and M. Kloft. Image Anomaly Detection with Generative Adversarial Networks. In Machine Learning and Knowledge Discovery in Databases. Springer, 2019.
- [4] B. Schölkopf, J. Platt, J. Shawe-Taylor, A. Smola, and R. Williamson. Estimating the Support of a High-Dimensional Distribution. In Proc. Neural Computation, vol. 13, pp. 1443-1471, July 2001.
- [5] N. Dionelis, M. Yaghoobi, and S. A. Tsafaris. Boundary of Distribution Support Generator (BDSG): Sample Generation on the Boundary. Paper Accepted for Publication in the Conference IEEE International Conference on Image Processing (ICIP), Oct. 2020.
- [6] J. Behrmann, W. Grathwohl, R. Chen, D. Duvenaud, and J. Jacobsen. Invertible Residual Networks. In Proc. 36th International Conference on Machine Learning (ICML), pp. 573-582, June 2019.
- [7] R. Chen, J. Behrmann, D. Duvenaud, and J. Jacobsen. Residual Flows for Invertible Generative Modeling. In Proc. Advances in Neural Information Processing Systems (NIPS), Dec. 2019.
- [8] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, June 2016.
- [9] A. Dieng, F. Ruiz, D. Blei, and M. Titsias. Prescribed Generative Adversarial Networks. Preprint arXiv:1910.04302 [stat.ML], 2019.
- [10] A. Grover, M. Dhar, and S. Ermon. Flow-GAN: Combining Maximum Likelihood and Adversarial Learning in Generative Models. In Proc. Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
- [11] T. Schlegl, P. Seeböck, S. Waldstein, U. Schmidt-Erfurth, and G. Langs. Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. In Proc. Information Processing in Medical Imaging (IPMI), Computer Science vol. 10265. Springer, June 2017.
- [12] H. Zenati, C. Foo, B. Lecouat, G. Manek, and V. R. Chandrasekhar. Efficient GAN-Based Anomaly Detection. Workshop Track in International Conference on Learning Representations (ICLR), April 2018.
- [13] J. Donahue, P. Krähenbühl, and T. Darrell. Adversarial Feature Learning. In Proc. International Conference Learning Representations (ICLR), 2017.
- [14] S. Akçay, A. Atapour-Abarghouei, and T. Breckon. GANomaly: Semi-Supervised Anomaly Detection via Adversarial Training. arXiv preprint, arXiv:1805.06725 [cs.CV], Nov. 2018.
- [15] S. Akçay, A. Atapour-Abarghouei, and T. Breckon. Skip-GANomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection. arXiv preprint, arXiv:1901.08954 [cs.CV], Jan. 2019.
- [16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. In Proc. Advances in Neural Information Processing Systems (NIPS), 2014.
- [17] C. Wang, Y. Zhang, and C. Liu. Anomaly Detection via Minimum Likelihood Generative Adversarial Networks. In Proc. 24th International Conference on Pattern Recognition (ICPR), Aug. 2018.
- [18] C. Ngo, A. Winarto, C. Li, S. Park, F. Akram, and H. Lee. Fence GAN: Towards Better Anomaly Detection. In Proc. IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), Nov. 2019.
- [19] I. Haloui, J. Sen Gupta, and V. Feuillard. Anomaly detection with Wasserstein GAN. Preprint arXiv:1812.02463 [stat.ML], Dec. 2018.
- [20] R. Devon Hjelm, A. Paul Jacob, T. Che, A. Trischler, K. Cho, and Y. Bengio. Boundary-Seeking Generative Adversarial Networks. arXiv preprint, arXiv:1702.08431 [stat.ML], Feb. 2018.
- [21] D. Hendrycks, M. Mazeika, and T. Dietterich. Deep Anomaly Detection with Outlier Exposure. In Proc. International Conference on Learning Representations (ICLR), New Orleans, US, May 2019.
- [22] N. Dionelis. Modulation-Domain Kalman Filtering for Single-Channel Speech Enhancement, Denoising and Dereverberation. Chapter 5: Phase-Sensitive Joint Noise Suppression and Dereverberation. PhD Thesis, Imperial College London, UK, Sept. 2019.
- [23] D. Hendrycks and K. Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In Proc. International Conference on Learning Representations (ICLR), 2017.

Electrical device classification using deep learning

R.O. Lane
QinetiQ
Great Malvern, UK

Abstract—Electrical devices unintentionally inject signals onto the mains power supply that can be measured using commercially available equipment designed for electromagnetic compatibility testing. This allows the identification of devices from their current time series when switched on or off. This paper outlines measurements of several devices and a process for simulating large amounts of annotated data based on those measurements. A convolutional neural net (CNN) algorithm for classifying signals is described. Performance of the CNN is compared to that of dynamic time warping (DTW) and correlation algorithms, with respect to varying training data set size and noise level. The CNN has the best performance for almost all cases considered.

Keywords—convolutional neural net; dynamic time warping; electrical signals; statistical model; time series classification

I. INTRODUCTION

Electrical devices unintentionally inject signals onto the mains electricity supply. Current time series produced by various devices are visually distinct when switched on or off [1] and algorithms can classify time series events when devices operate individually [2]. Useful information about the “pattern of life” of equipment could thus be deduced from the mains current and reveal what is happening inside a building, with applications for smart metering or intelligence gathering. This paper summarizes results of a measurement trial, describes a process to simulate large amounts of data, specifies an algorithm to classify on/off events, and compares the performance of deep learning, dynamic time warping, and correlation classification algorithms when applied to devices operating in a noisy environment.

II. ELECTRICAL DEVICE DATA

Measurements of several devices were made as described in [1]. Root mean-square (RMS) current signals were measured at a sample rate of 50 Hz. Data was recorded for the cycle: power off, power on, power off, or standby, on, standby for devices with a standby mode. The signals were recorded with a consumer unit typically used with building electrical supplies and a 15 m extension lead. These measurements are representative of what could be obtained in the field with a passive measurement device. The following devices were tested: compact fluorescent lamp (CFL), cathode ray tube (CRT) television, laptop and charger (two different models of charger), power drill, desk fan, filament lamp, flat screen television, halogen lamp, light emitting diode (LED) lamp, and Nokia and Samsung phone chargers. These devices were selected to cover a broad range of characteristics and have significant variability in the amount of power consumed. Details about the make and model of each device are given in [1].

III. GENERATING REALISTIC SIMULATED DATA

A. Single device simulations

The above measurements were made for devices operating individually on different days to test the consistency of device signatures over time. On/off events were manually annotated. For analyzing data from devices operating simultaneously, it is impractical to annotate the large amounts of data required to measure every combination of device. Therefore, a process for simulating device signatures based on measurements was developed in [2]. This process analyzed the nonlinear variations in time exhibited by each signature, as well as local variations in signal amplitude, to construct a statistical model that can be used to generate signals randomly for each device.

One issue with the data simulation model in [2] is that amplitude and time were both assumed to vary according to a multi-variate Gaussian probability distribution. This is a reasonable assumption for much of the data but some parts of the signals have significantly skewed distributions and a Gaussian model is not suitable. This can be seen by comparing Fig. 1 and Fig. 2, which show measured data of a switch-on event for a CRT TV and data simulated using the Gaussian model. At the switch-on time of sample index 100, some simulated data examples exhibit significant negative relative current values while the real data does not have this feature. The distribution of the peak current at sample 106 in the measured data is skewed and sometimes has values as large as 0.3 A. In the simulated data the peak is symmetrically distributed about a value of 0.26 A. These observations motivate the need for a non-Gaussian distribution.

A key aspect of the simulated data model is that it incorporates correlations between signal values at different times. This is necessary to deal with the fact that significant correlations exist in the measured data. For most parts of the signal, the data is positively correlated – slowly varying signals are likely to have a high sample value if the previous sample was also high. However, some negative correlations also exist. For example, when there is a sharp spike in the data, this often is spread over adjacent samples. Since the total energy of the spike is approximately constant, if one sample is high then the adjacent one will be low. Modelling asymmetric correlated non-Gaussian data is not straightforward, as not many suitable probability distribution families exist. However, it is possible to model such data using copulas.

Following [3], a d -dimensional copula C is a multivariate distribution function on the unit hypercube $[0,1]^d$ with uniform marginal distributions for each dimension.

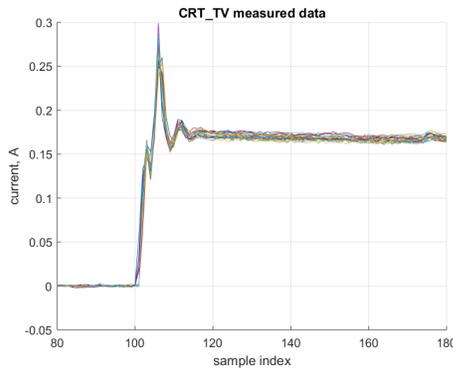


Fig. 1. Segments of measured CRT TV data for a switch-on event.

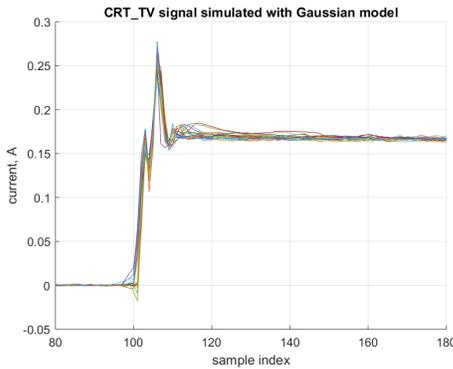


Fig. 2. Segments of CRT TV data simulated using a Gaussian amplitude and time-warp model.

Sklar's theorem says that if H is an arbitrary multidimensional distribution function with one-dimensional marginals F_1, \dots, F_d then $H(x_1, \dots, x_d) = C(F_1(x_1), \dots, F_d(x_d))$ for some copula C , which always exists. Thus, the one-dimensional marginals and multivariate dependence structure of a multi-dimensional distribution can be separated, with dependence represented by a copula. This means that when constructing a model, one has complete freedom in choosing different distributions for each dimension, including distributions from different families. This is important for the signal-modelling task as it allows non-Gaussian distributions to be assigned to the signal values at each time step. The following discussion outlines how these marginal distributions are independently defined. Attention is then given to choosing a copula dependence model.

A key observation about the device data is that the distribution of different parts of each signal can be symmetric, positively skewed, or negatively skewed. While the copula model in principle allows an arbitrary distribution to be assigned to each sample, it is more convenient in practice to use a single distribution family, but with varying parameters. It is therefore necessary to select a distribution with enough flexibility to model signals with different characteristics, while being simple enough that its parameters can be estimated from limited data sets. A suitable choice is the four-parameter beta distribution, as in addition to being able to vary the skewness it is possible to set minimum and maximum bounds on the signal, which prevents spurious large or small values being produced. The probability density function of the four-parameter beta distribution is:

$$p(x) = \frac{\Gamma(\alpha + \beta)(x - L)^{\alpha-1}(U - x)^{\beta-1}}{\Gamma(\alpha)\Gamma(\beta)(U - L)^{\alpha+\beta-1}} \quad (1)$$

This definition is valid for $L \leq x \leq U$, where L is the lower bound and U is the upper bound. Outside of this range $p(x) = 0$. The parameters $\alpha > 0$ and $\beta > 0$ are shape parameters and $\Gamma(\cdot)$ is the well-known gamma function.

Given measured data, it is necessary to estimate parameters of the distribution to form a concrete model of how the signal varies at each time sample index. Two widely used generic methods for parameter estimation are maximum likelihood and the method of moments. However, both methods have drawbacks when naively applied to the four-parameter beta distribution. The global maximum of the likelihood function in unconstrained parameter space can correspond to infeasible solutions, where the estimated range $[L, U]$ is narrower than the range of the data [4]. Useful parameter estimates can be obtained at a local maximum, but this is sometimes difficult to locate when the sample size is small, which is the case for the data set at hand. Direct moment matching can also lead to infeasible solutions. A feasibility-constrained moment matching technique was developed in [5], where the first four moments of the solution only approximately match those of the data.

To avoid the above issues, a robust modification of the hybrid procedure in [6] was implemented. This decomposes the problem into two tractable tasks: estimate the lower and upper bounds first and then given these bounds, estimate the shape parameters. A robust estimate of the bounds is given by:

$$\hat{L} = \min(x) - range(x)/(n - 1) \quad (2)$$

$$\hat{U} = \max(x) + range(x)/(n - 1) \quad (3)$$

This guarantees that the data lies within the bounds and sets each bound at a distance from its nearest data point equal to the mean spacing between all points in the set. In [6], linear interpolation on order statistics is used to define different distances for the lower and upper bound. However, this method is highly sensitive to the spacing of the lowest and highest two data points and, in practice, the estimators (2) and (3) have been found by the present author to be more robust. Once the lower and upper bounds have been set, the data can linearly be rescaled from $[\hat{L}, \hat{U}]$ to $[0, 1]$. Maximum likelihood estimation of α and β then proceeds as usual for the two-parameter beta distribution, a well-conditioned problem.

As an alternative to the above hybrid procedure, a fully Bayesian approach to estimating all four parameters of the beta distribution using Markov chain Monte Carlo (MCMC) was implemented. MCMC represents uncertainty in the parameter estimates by a set of samples, which could, in principle, be used in the signal simulation process. However, signals simulated using the hybrid procedure were found to be of sufficient fidelity and it was judged that the extra run time associated with MCMC outweighed the benefit gained in more accurate modelling of uncertainty. Therefore the results presented in this paper are based on the hybrid maximum likelihood procedure.

The dependence structure of the copula model is now defined. A large proportion of the discussion surrounding copulas is based on two-dimensional distributions, but the signal

simulation task requires models with more than 100 dimensions. A convenient copula for large dimension problems is the Gaussian copula [7]. This is defined as:

$$C(u_1, \dots, u_d) = \Phi_G(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_d); R) \quad (4)$$

where Φ is the cumulative distribution function (CDF) of the standard normal distribution, and Φ_G is the standard multivariate normal distribution with correlation matrix R . The Gaussian copula is flexible as it allows for varying degrees of positive or negative correlation for each pair of variables.

The Gaussian copula parameter R is estimated using a two-step maximum likelihood technique known as Inference Functions for Margins (IFM) [8]. In the first step, parameters of the marginal four-parameter beta distributions F_i for each dimension (time sample index) $i = 1, \dots, d$ are estimated from the data using the hybrid procedure described above. The data x_{ij} (for $j = 1, \dots, n$ with n data points) are then transformed to a standard multivariate normal distribution by:

$$g_{ij} = \Phi^{-1}\left(F_i(x_{ij}|\hat{\alpha}_i, \hat{\beta}_i, \hat{L}_i, \hat{U}_i)\right) \quad (5)$$

The estimated correlation matrix \hat{R} is simply the covariance of $[g_{ij}]$.

Data can be generated from the copula model by first drawing samples \tilde{g}_{ij} from a multivariate Gaussian distribution with covariance matrix \hat{R} . The samples from each dimension are then transformed using:

$$\tilde{x}_{ij} = F_i^{-1}\left(\Phi(\tilde{g}_{ij})|\hat{\alpha}_i, \hat{\beta}_i, \hat{L}_i, \hat{U}_i\right) \quad (6)$$

The resulting \tilde{x}_{ij} represent a set of constant-length randomly simulated beta distribution signals aligned to the same time base.

To model temporal variation of the signals, the non-linear warp function model described in [2] can be applied. That warp model is also based on a multivariate Gaussian distribution. The model can be upgraded to use copulas in the same manner as for signal amplitudes, as described above. Therefore, a warp model using four-parameter beta distribution marginals and a Gaussian copula was implemented.

A copula model for amplitude and temporal variation was estimated for each device independently. Each model was then used to simulate new device signals.

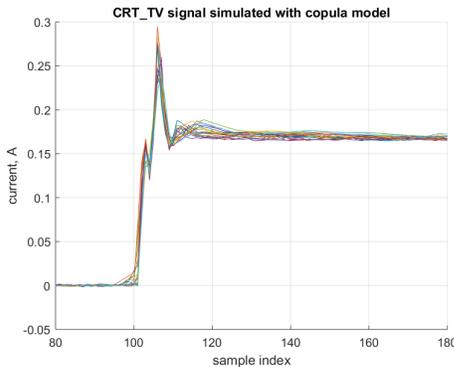


Fig. 3. Segments of CRT TV data simulated using an amplitude and time-warp copula model.

A subset of simulated data is shown in Fig. 3. Comparing to Fig. 1, the generated signals look realistic. Furthermore, comparison with Fig. 2 shows that signals generated from the copula model have a higher fidelity than those generated using the Gaussian model. In particular, the copula model does not produce spurious large negative values at the switch-on time of sample index 100, and the peak current at sample 106 shows the correct amount of skew.

B. Simultaneous device simulations

The measured and simulated data described in sections II and III-A pertain to single devices operating at a time. This data is useful for isolating the effect of each device and determining the intrinsic ability of machine learning algorithms to classify the data. However, a real system would measure a single long time series with multiple signals potentially superimposed on each other. It would be necessary to detect and extract relevant signals before classification. Detection of an on event is simple as all devices either produce a spike or step change in the current. A section of the data can then be extracted from 1 s before the event to 15 s after it, which covers the transient signal of all devices. That section of data is used as input to the classifier.

The effect of multiple devices operating simultaneously on classification performance can be tested in two ways. The first is to add zero-mean Gaussian noise with a certain variance. Due to the central limit theorem, this is a reasonable approximation when a large number of devices are in simultaneous use and their noise is additive. The total noise power is related to the steady-state noise introduced by each device while it is operating.

The second method for multiple device simulation is to simulate individual device signatures, using the procedure from section III-A, and superimpose them to produce a composite signal. It is assumed that linear superposition applies so that time series may simply be added. This has been verified via measurement. The signal aggregation process proceeds as follows. The initial state of the system is a time series with Gaussian noise with a variance defined by the background noise power. At a specified point in time, a device is randomly chosen and a simulated switch-on signal is superimposed on the background noise. Once the transient part of the switch-on signal has finished, a steady state model for that device is used to persist its signal until a switch-off event for that device is simulated or the end of the simulation is reached. The above process is repeated for multiple devices, cumulating the signal and steady state noise from each device over time. When a switch-off event is simulated, the device switch-off transient is applied to the aggregate signal and the effect of the steady-state model for that device is removed. This method for multiple device simulation allows one to test the effect of non-Gaussian and structured noise produced during the steady state on period of devices. It also facilitates the production of arbitrarily long time series with automatic annotation of the time of switch-on and switch-off events, along with the label of the device associated with each event. This enables the production of large amounts of training data, which is required by deep learning algorithms [9], and test data to obtain statistically significant results. This paper uses the simple Gaussian noise method for simultaneous device analysis. The use of simulated superimposed signals will be the subject of future work.

A steady-state noise model is required to pad short simulated signals to constant length. The model was constructed as follows. First, steady-state parts of each measured signal were extracted in the following manner. All signals for a single device were warped to a reference time-base using the dynamic time warping (DTW) procedure from [2]. The start and end times of the steady state portion of the mean reference signal were manually annotated. The inverse of the warp function for each signal was then used to transform these reference start/end times to actual start/end times for the original signal. These times were used to extract segments of steady state noise for each device. These segments were concatenated and their mean spectrum estimated using the Welch method [10], with an FFT size of 256 samples (5.12 s). The spectrum was used to filter a white Gaussian noise signal of the required length. The mean and standard deviation of the colored noise were then adjusted by linear scaling to match those of the steady state portion of the signal to which the noise was being concatenated. A comparison of measured and simulated steady-state noise for a flat-screen TV is given in Fig. 4 and Fig. 5. It is seen that the simulation correctly captures the time-varying structure of the measured data spectrum.

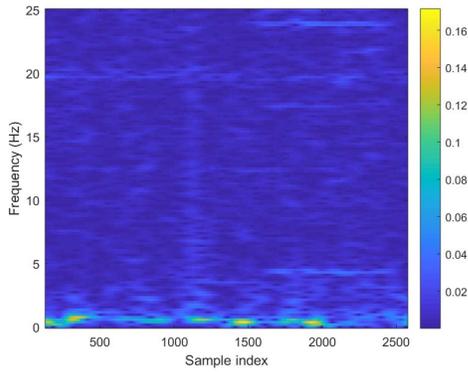


Fig. 4. Spectrogram of measured steady state noise from a flat-screen TV.

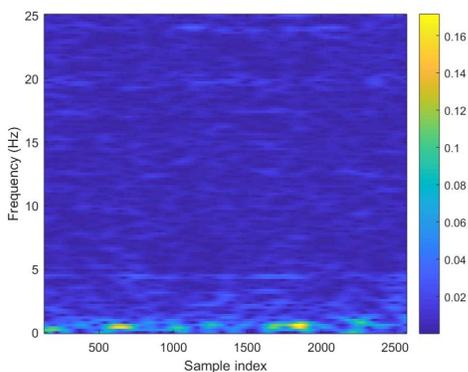


Fig. 5. Spectrogram of simulated steady state noise from a flat-screen TV.

IV. DEEP LEARNING CLASSIFICATION

Previous work in [2] developed a dynamic time warp algorithm for signal classification. While performance was shown to be good in noise-free environments, it is likely that

advances in deep learning could be used to develop algorithms with performance that is more robust in a range of scenarios.

One particular form of deep learning is a convolutional neural net (CNN). As with dynamic time warping, this type of classifier is robust to linear and non-linear signal variations in time, which are present in the data. A CNN classifier was implemented with structure shown in Table 1, where n_f is the number of filters per layer, n_l is the number of max-pool layers, and n_h is the number of hidden nodes in the dense layer. This structure is inspired by the VGG-16 image recognition architecture [11], but adapted to one-dimensional signals with some modifications. At the signal edges, zero padding is applied before convolution so that the signal remains the same length. The Adam optimizer and a categorical cross-entropy loss function were used. All experiments used 100 training epochs, which was sufficient for accuracy on the training set to converge. In many specific cases, training for fewer epochs would have been sufficient. A simple architecture search procedure, which assumed the effect of each parameter could independently be analyzed, found optimum values of the network parameters to be $n_f = 32$, $n_l = 6$, and $n_h = 32$. This results in a network with 28,364 parameters. It is likely that a more comprehensive search could improve the architecture further.

Layer Type	Filter size	#Filters or nodes	Activation function	#Parameters
Conv1D	3	n_f	ReLU	128
MaxPool1D	2	N/A	None	0
Repeat the above two layers ($n_l - 1$) times				$3104(n_l - 1)$
Dense	N/A	n_h	ReLU	12320
Dropout	N/A	0.5	None	0
Dense	N/A	n_d	Softmax	396

Table 1. Base convolutional neural net structure. The number of devices n_d is fixed at 12. The #Parameters column assumes $n_f = 32$ and $n_h = 32$.

V. PERFORMANCE ASSESSMENT

The single-device simulation procedure described in section III-A produces simulated signals of different lengths, due to the temporal variations in the warp model. For input to the classifiers, all signals were standardized to be the same length of 800 samples (16 s). Raw signals longer than this were truncated. Signals shorter than this were padded with device-specific steady state noise using the Welch filter process described in section III-B. Two hundred switch-on event signals per device were simulated and the data were split into training (70%) and test (30%) sets. For comparison, three classifiers were applied to the data: correlation and dynamic time warping (DTW), as described in [2], and a CNN, as described in section IV.

The effect of making measurements with multiple simultaneous devices on classifier performance was analyzed by adding zero-mean white Gaussian noise with varying standard deviation σ as described in section III-B. In the initial set of experiments, the same amount of noise was added to both the training and test data. Experiments were repeated ten times and

the mean and standard deviation of classifier accuracy were computed from the Monte Carlo runs. The results are shown in Fig. 6. All classifiers have perfect performance in a noise-free environment and this degrades as the noise level increases. The CNN classifier has the best performance for all noise levels.

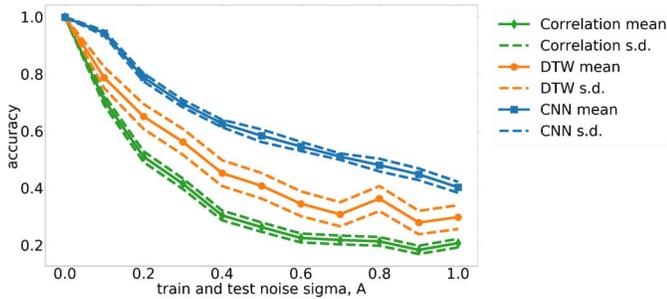


Fig. 6. Accuracy vs. noise level when test and train noise levels are the same.

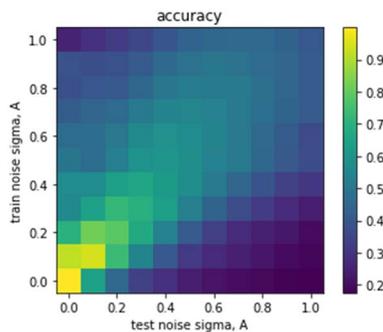


Fig. 7. CNN accuracy vs. noise level for different test and train noise levels.

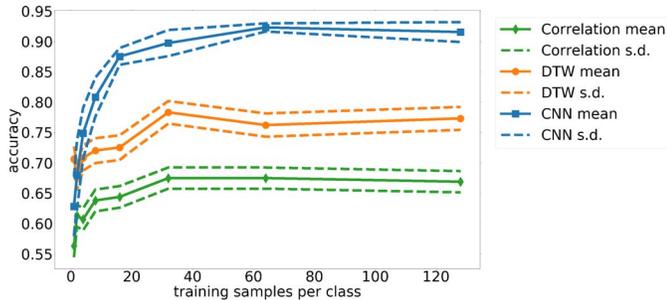


Fig. 8. Accuracy as a function of training data set size, with noise set to 0.1 A.

A second experiment was carried out to determine the effect of testing a classifier where the noise conditions are different to those present during training. The results for the CNN classifier are shown in Fig. 7. It is seen that performance is generally best when test conditions match training conditions. It is further noted that there is an asymmetry in Fig. 7 – a classifier trained with a certain specified level of noise performs better in noise-free test conditions than when trained in a noise-free environment but tested with the same specified amount of noise. Therefore, to be useful in practice, classifiers should be trained with multiple levels of noise.

Increasing training data volume allows classifiers to construct a more accurate model. Accuracy as a function of training set size is shown in Fig. 8, where a noise level of 0.1 A

was used for both train and test. Performance increases rapidly with training set size, but plateaus once 32–64 training samples per class are available. When only one or two training samples are available, DTW outperforms the CNN. However, the CNN gains the largest benefit from training data and outperforms other classifiers with four or more training samples per class.

VI. CONCLUSIONS

A sophisticated data simulation process has been designed that can generate unlimited amounts of single-device event data. This can be used to test classifier performance as a function of training data set size or noise level. At least 64 training examples per device were required to achieve maximum performance. Classifier performance drops with increasing noise levels but the CNN classifier maintains a fair accuracy even with Gaussian noise levels of 1.0 A. This compares favorably to the DTW and correlation classifiers, whose performance is the same or worse than the CNN classifier for any given noise level.

The data simulation process enables the creation of long time series with multiple devices operating simultaneously and annotated switch-on or off events. This could be used to analyze signal detection algorithms, which should be the subject of future work. The effect of structured noise on classifier performance should also be analyzed.

REFERENCES

- [1] R.O. Lane, T.A. Badran, G.F. Marshall, “Measurements for remote identification of electrical equipment”, *Electronic Letters* 53(15):1001-1005, 2017.
- [2] R.O. Lane, Remote identification of electrical devices using dynamic time warping, *Sensor Signal Processing for Defence Conference*, London, 6-7 December 2017.
- [3] P. Embrechts, F. Lindskog, A. McNeil, Modelling dependence with copulas and applications to risk management, In S. T. Rachev (Ed.) *Handbook of heavy tailed distributions in finance*, Elsevier, Rotterdam, pp. 329-384, February 2003.
- [4] J.V. Carnahan, Maximum likelihood estimation for the 4-parameter beta distribution, *Communications in Statistics - Simulation and Computation* 18(2):513-536, 1989.
- [5] S.M. AbouRizk, D.W. Halpin, J.R. Wilson, Fitting beta distributions based on sample data, *J. Construction Engineering and Management* 120(2):288-305, June 1994.
- [6] D.E. Bertrand, Comparison of estimation techniques for the four parameter beta distribution, Masters Thesis, Air Force Institute of Technology, December 1981.
- [7] N. Channouf, P. L'Ecuyer, Fitting a normal copula for a multivariate distribution with both discrete and continuous marginals, *Winter Simulation Conference*, Austin, TX, USA, December 2009.
- [8] P.K. Trivedi, D.M. Zimmer, Copula modeling: an introduction for practitioners, *Foundations and Trends in Econometrics* 1(1):1-111, 2007.
- [9] C.P. Moate, S.D. Hayward, J.S. Ellis, L. Russell, R.O. Timmerman, R.O. Lane, T.J. Strain, Vehicle Detection in Infrared Imagery using Neural Networks with Synthetic Training Data, 15th Int. Conf. on Image Analysis and Recognition (ICIAR 2018), Póvoa de Varzim, Portugal, June 27-29, 2018.
- [10] P.D. Welch, The use of Fast Fourier Transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms, *IEEE Trans. on Audio and Electroacoustics* 15(2):70-73, 1967.
- [11] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, *Int. Conf. on Learning Representations (ICLR)*, San Diego, CA, USA, May 2015.

Multimodal Learning for Early Detection of Explosive Sounds using Relative Spectral Distribution

Vishwajeet Shukla
Samsung Research Institute
 Noida, India
 v.shukla@samsung.com

Mayank Singour
Samsung Research Institute
 Noida, India
 m.singour@samsung.com

Abstract—Human safety is a major challenge in the modern world. Bomb blasts and public shooting incidents are on the rise. Early recognition of such occurrences by detecting corresponding blaring audio events could help in providing swifter aid and reducing overall devastation. Hence rapid and accurate identification of related audio events is a crucial task for public safety. In this paper we propose a novel approach of Multimodal learning for the automated detection of explosive sounds such as Gunfire, Machine gun, Fusillade, Artillery Fire, Cap Gun and Explosion sounds for audio surveillance task. We present a deep feature stacking based method to classify explosive and non-explosive sounds substantially outperforming existing methods. We also discuss the challenges involved in explosive sound detection based on the inferences observed.

Index Terms—Artificial Intelligence, Audio signal processing, Explosive sound detection, Multimodal learning

I. INTRODUCTION

The problem of inadequate automated audio surveillance has been contemplated by the research community recently. Extreme inhuman events have increased drastically nowadays such as gunfire in schools, religious and other sensitive places. A study shows that the number of people killed by gunfire in USA alone in 2017 was almost 40,000 [1]. Myriads of innocent lives are lost due to unpreparedness, hence early detection of explosive sounds can play a vital role in combating with these situations. Acoustic sensors have been used in battlefields since World War I to accurately deduce weaponry strength. Modern world calls for cutting edge technology in audio signal processing for safeguarding [2] [3].

Explosive sound detection falls under the domain of Audio Event Detection (AED). AED has a large number of applications such as Animal care [4], Human disease symptoms detection [5] [6], audio detection based surveillance [7] [8], monitoring living beings such as baby crying [9] or pet making sound, etc. AED can be complementary to video and image based event detection and on top of that it can be advantageous in the scenarios where visual signals are not clear or unavailable such as snowfall, rain or fog.

In this paper, we mainly concentrate on audio surveillance using explosive sound detection. Explosive sounds have slightly

We would like to acknowledge and thank SRI-Noida for the hardware support

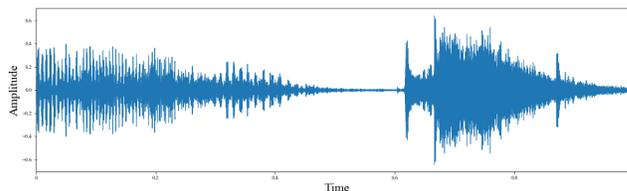


Fig. 1. Representation of sample audio waveform of a gunshot sound.

different amplitude and variant spectral characteristics as compared to the normal audio clips. The main contributions of this paper are condensed as follows:

- (a) A novel multimodal network with deep feature stacking using deep convolutional neural network accompanied by blending of features extracted from raw waveform
- (b) Method to utilize octave based relative spectral distribution and average spectral envelop of an audio signal for explosive sound detection
- (c) A state of the art method for feature extraction using temporal and spectral characteristics of impulsive sound.

II. RELEVANT WORK

Some of the earliest work done in the field of Audio Event Detection(AED) are content based classification and retrieval of audio recording [10] [11] [12]. Other relevant work [13] on Acoustic Event Detection used deeper CNN architecture and data augmentation [14] [15] to classify 14 classes of environmental sounds. Recent work [7] in 2018 provides audio surveillance system for detecting hazardous road incidents such as tire skidding, car crashes. In [16] authors have classified the audio clips of Audioset (a dataset of audio events that was released by Google [17]) into 600 different classes using log mel spectrogram as input feature. DCASE launches competition based on audio event classification and detection every year since 2013 [18] [19] [20].

Subsequent work in relevant area is focused on specific sound event detection such as gunshot and scream detection [21]. Authors in [22] have used features such as MFCC, 8th order LPC coefficients etc. and HMM classifier to classify the audio

excerpts into gunshots and other sounds. Reference [23] utilizes a sliding window Hidden Markov Model (HMM), it tries to identify abnormal sounds such as gunshot, glass breaking, and explosion using different spectral and harmonics features. Although there are a lot of exploratory research available on gunshot and impulsive sound detection, but precise results are still missing.

The paper is structured as follows: Section III describes the Feature Representation of audio signal used in our Multimodal network. In Section IV we provide an overview of the dataset used for training and validation purposes. Section V describes the deep Multimodal network and in the further sections we discuss results and observations.

III. FEATURE REPRESENTATION

Audio features are extracted using raw audio waveforms (Fig.1). Most popularly used audio features are STFT, spectrogram, MFCC, LPC etc. Since gunshots and explosions have variant nature of spectral characteristics hence average spectral envelope based features are not sufficient for explosive sound detection [24]. Therefore, we have used relative spectral characteristics along with spectrogram to capture the consolidated spectral properties of an audio signal (Fig.2). Other audio features such as MFCC, spectral roll off either closely resemble or perform poorly in comparison with above said features in terms of explosive sound detection task.

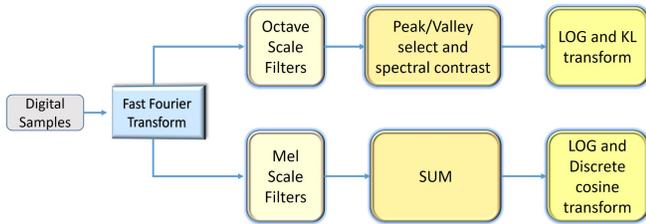


Fig. 2. Flow of Spectrogram and Spectral Contrast calculation

A. Spectral Contrast

Spectral contrast is a feature which was originally presented in “Music type classification by spectral contrast feature” [25] [26] which classifies Music type among different classes such as baroque, romantic, pop, rock and jazz music. Since it uses the idea of peak and valley spectrum, it is a representation of relative distribution in a signal. To the best of our knowledge this feature has not been used for explosive sound detection. We have used this feature to capture the relative spectral characteristics for highly varying nature of explosive sounds such as gunshots.

Feature Extraction :

For each frame of 2048 samples of an audio signal, FFT is applied to get spectral components and then divided into six sub-bands based on octave scale. Let, FFT of k -th sub-band is $\{x_{k,1}, x_{k,2}, \dots, x_{k,N}\}$. After sorting this FFT vector by magnitude, suppose we get a new vector $\{x'_{k,1}, x'_{k,2}, \dots, x'_{k,N}\}$

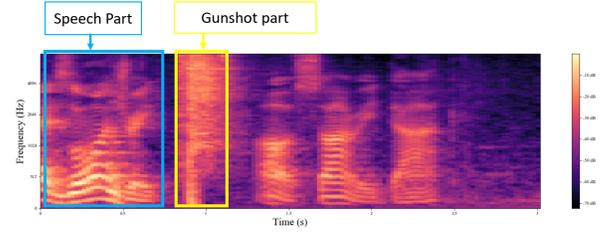


Fig. 3. Spectrogram of a gunshot sound

such that $x'_{k,1} > x'_{k,2} > \dots > x'_{k,N}$. Now, spectral peaks and valleys are calculated as :

$$Peak_k = \log \left\{ \frac{1}{\alpha N} \sum_{i=1}^{\alpha N} x'_{k,i} \right\} \quad (1)$$

$$Valley_k = \log \left\{ \frac{1}{\alpha N} \sum_{i=1}^{\alpha N} x'_{k,N-i+1} \right\} \quad (2)$$

where $\alpha = 0.02$ (standard value based on experiments). Spectral Contrast (SC) is defined as :

$$SC_k = Peak_k - Valley_k, \text{ where } k \in [1, 6] \quad (3)$$

After Log transform on obtained vector, Karhunen-Loeve transform is applied to eliminate the relativity among different dimensions. The above mentioned process can be similarly applied to all the frames obtained from raw audio digital samples to get complete Octave-based Spectral Contrast representation for the audio.

B. Log-mel Spectrogram

Mel Spectrogram is used for audio classification primarily as an input to neural networks [16]. Mel scale is based on the variant frequency of human ear’s critical bandwidth to capture important characteristics of speech signal. Notice the peaks (formants) in the spectrum as shown in Fig.3, sounds can be identified much better by the Formants and by their transitions.

IV. DATASET

Generating dataset for explosive sounds is difficult as these are one of the rare events. We have used Audioset [17], a dataset containing 632 classes of different categories of sounds such as car engine, dog bark etc. (released by Google in 2017).

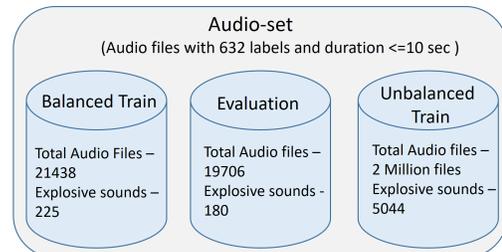


Fig. 4. Given containers shows internal distribution of Audioset and the extracted number of audio files from each sub component.

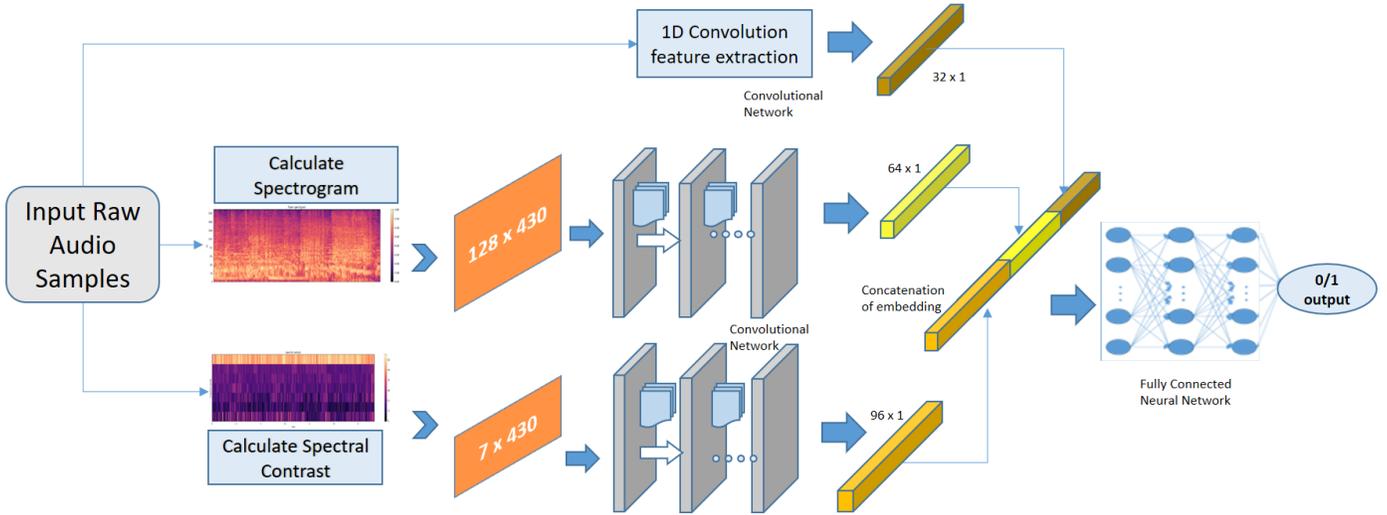


Fig. 5. Multimodal architecture for explosive sound detection using spectral contrast, spectrogram and feature extraction using 1D convolutional operations on raw waveform followed by a fully connected network which outputs the probability of audio signal belonging to explosive sound category

We have filtered the explosive sounds for our task such as Gunshot, Machine gun, Fusillade, Artillery Fire, Cap Gun and Explosion from the Audioset. We have used 4510 non-explosive sounds from the ‘unbalanced train set’ randomly, containing excerpts from sounds such as speech, music, car engine and other perceptually similar impulsive sounds such as door slam, bang, yell etc. to train alongside the mentioned explosive sound data (Fig.4). These audio excerpts are of less than or equal to 10s in duration, as a result we have around 25 hours of dataset in total.

UrbanSound8K [27], a dataset for Environmental Sound Classification also contains gunshot sounds, but it has relatively smaller number and less variety of gunshot sounds (342 samples of ≤ 4 sec each). We have also benchmarked our method on this dataset achieving substantial improvement as compared with previous existing methods (CNN and LSTM) [28] while using five fold cross validation. We have also used convolution based architectures which are faster in execution as compared to LSTMs and other memory cell based networks.

V. MODEL ARCHITECTURE

We have used custom made lighter version of ResNet [29] based convolutional architecture for both baseline and proposed multimodal network due to its ability to counter vanishing gradient problem. Although Audioset has comparatively diverse set of samples of explosive sounds, it has not been utilized specifically for the explosive sound detection task. Hence, we also present a baseline model on this dataset using log mel spectrogram as input feature.

A. ResNet Network using spectral features

Our model contains three Residual blocks where each block consists of two convolutional layers each accompanied with batch-normalization and Relu activations (Fig.6). The input of this network is spectral representation of audio i.e. spectrogram

or spectral contrast and has a binary output denoting explosive or non-explosive sound.

B. Multimodal Network

The Multimodal network consists of two 2D ResNet networks and a 1D convolutional network running in parallel. It is a Multi input one output network (Fig.5). Two of these networks take spectrogram and spectral contrast as input and the other network take raw spectrogram as input and extract features corresponding to the same audio clip using 1D convolutional neural network consisting of 3 residual blocks same as Fig.6 but replacing 2D convolution with 1D convolutional kernels.

We concatenate the output of intermediate layers of these three networks. Note that batch normalization before merging helps in normalization of outputs of all the networks and thus reduce the chance of adverse impact of one network on the other. A fully connected neural network is built on the top of merged layer to learn the feature representation of combination of these three networks.

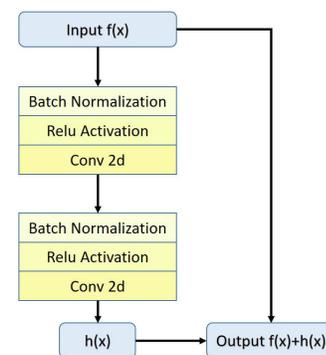


Fig. 6. Internal schematic representation of one residual block used in Multimodal architecture

VI. RESULTS AND DISCUSSIONS

TABLE I
PERFORMANCE OF DIFFERENT MODELS BASED ON CHANGE IN INPUT OR ARCHITECTURE

Architecture and dataset	Accuracy	F1 Score
Spectrogram for Gunshot	85.3 %	0.87
SpectralContrast for Gunshot	89.8%	0.91
Spectrogram for Explosion	86.5%	0.91
SpectralContrast for Explosion	77.9%	0.84
Spectrogram for all data	85.3%	0.86
SpectralContrast for all data	87.9%	0.89
Bi-modal Architecture for all data	90.0%	0.91
Bi-modal with data augmentation	91.2%	0.92
Multimodal Architecture for all data	91.6%	0.92

TABLE II
PERFORMANCE ON URBANSOUND 8K DATASET

Architecture	Accuracy	F1 Score
CNN architecture	NA	0.91
LSTM architecture	NA	0.94
Proposed Multimodal	95.9%	0.98

Using individual features : We have captured relative and average spectral distribution using spectral contrast and log mel spectrogram with the results as shown in Table I. It is evident that spectral contrast performs better in case of gunshot detection due to larger relative spectral variation in gunshot sounds and spectrogram performs better for explosion sounds.

Using Bi-modal architecture with spectral features: We have extracted deep features using Multimodal deep feature stacking network to capture all the spectral variation of explosive sounds in frequency domain. This bimodal architecture takes spectral features represented by spectrogram and spectral contrast of an audio signal as input. We have shown a considerable improvement in performance metrics by using bimodal network as compared to the network with only spectrogram as input, as shown in Table I.

Using Multimodal network: For capturing temporal information from raw audio waveform we have extracted features using custom made ResNet based 1D convolutions. The impressive performance of spectral contrast for gunshot sound and spectrogram for the explosion sound detection prompted us to develop a Multimodal network which can account for

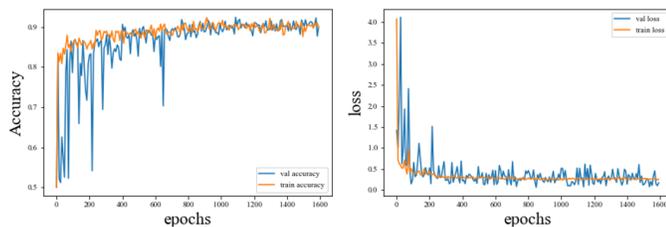


Fig. 7. Comparison of loss and accuracy vs epochs

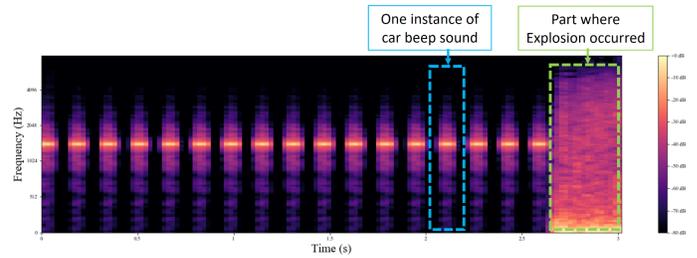


Fig. 8. Difference in spectral distribution as evident from spectrogram in car beep and explosion sounds

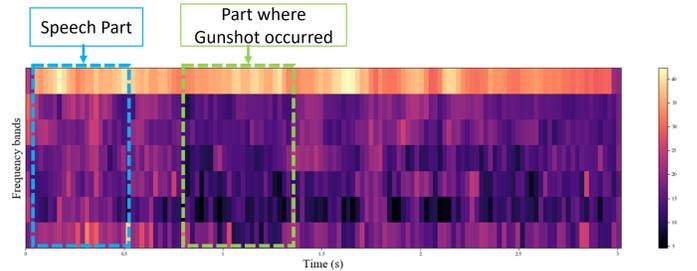


Fig. 9. Difference in relative spectral distribution as evident from spectral contrast in gunshot and speech sounds

temporal, relative and average spectral distribution for an audio. Our Multimodal network substantially outperforms the baseline method as evident from Table II. Model improvement and convergence is shown in Fig.7.

VII. OBSERVATIONS

- 1) Spectral contrast being a relative spectral distribution performs better for gunshot detection (Fig.9) and spectrogram performs better for explosion sounds (Fig.8). Feature extraction using 1D convolutions on raw waveform captures the basic properties such as amplitude variation in time domain. Therefore, our Multimodal architecture captures all the aspect of an audio signal and is able to classify various kind of explosive sounds with great precision.
- 2) Although we have reasonable number of data points for the explosive sounds, we can generate more quality data points to improve the accuracy and reduce the error rate. Artificial data generation using Generative adversarial networks corresponding to similar scenarios might be one of the solutions to fulfill data scarcity.
- 3) Similar architecture and method can be applied for other impulsive sound detection task such as door slam, glass cracking etc.

VIII. CONCLUSION

In this paper, we proposed a novel Multimodal Residual neural network for explosive sound detection using deep feature stacking. We have shown the importance of relative and average spectral distribution of an audio signal with blending of features extracted using 1D convolutions on raw audio waveform for

explosive sound detection. We have achieved close to accurate result of 0.98 F1-Score on UrbanSound8k dataset and an improvement of 6.3% as compared to baseline method on Audioset. Our Future work will be directed towards applying the proposed method to different applications such as health care and audio-visual surveillance.

REFERENCES

- [1] CDC Wonder "Underlying Cause of Death," The Trace, Statistics that tell the story of Gun Violence in 2018, newsroom, 2018.
- [2] Balakrishnan Kaushik, Donald Nance and K.K. Ahuja, "A Review of the Role of Acoustic Sensors in the Modern Battlefield," Collection of Technical Papers - 11th AIAA/CEAS Aeroacoustics Conference, volume 4, 2005.
- [3] Charles Jokel, Kurt Yankaskas, and Martin B. Robinette, "Noise of military weapons, ground vehicles, planes and ships," The Journal of the Acoustical Society of America, pages 3832-3838, volume 146, 2019.
- [4] Sasmaz, Emre, Tek F., "Animal Sound Classification Using A Convolutional Neural Network," 3rd International Conference on Computer Science and Engineering (UBMK), Sarajevo, pages 625-629, 2018 .
- [5] Wang, Jia-Ming Liu, Guo-Zheng L. and Guo-Zheng Li, "Audio signals encoding for cough classification using convolutional neural networks: A comparative study," IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pages 442-445, Washington, DC, 2015.
- [6] Rubin, Jonathan and Abreu, Rui and Ganguli, Anurag and Nelaturi, Saigopal and Matei, Ion and Sricharan, Kumar, "Recognizing Abnormal Heart Sounds Using Deep Learning," CoRR Journal, volume abs/1707.04642, 2017.
- [7] Noor Almaadeed, Somaya Al-Maadeed, Ahmed Bouridane, Muhammad Asim, Azeddine B., "Automatic detection and classification of audio events for road surveillance applications," Sensors Journal, volume 18, page 1858, 2018 .
- [8] Faycal Ykhlef, Sarah Ahmed Hamada, Farid Y., Abdeladhim Derbal and Djamel Bouchaffra, "Real-time detection of impulsive sounds for audio surveillance systems," 3rd edition of the National Study Day on Research on Computer Sciences (JERI 2019), volume 2351, paper 49, 2019.
- [9] Lavner Yizhar, Cohen Rami, "Infant Cry Analysis and Detection," IEEE 27th Convention of Electrical and Electronics Engineers in Israel, pages 1-5, 2012.
- [10] Guodong Guo and Stan Z. Li, "Content-Based Audio Classification and Retrieval by Support Vector Machines," IEEE Transactions on Neural Networks, volume 14, pages 209-215, 2003.
- [11] E. Wold, T. Blum, D. Keislar, J. Wheaton, "Content-based classification, search, and retrieval of audio," IEEE MultiMedia Journal, volume 3, pages 27-36, 1996.
- [12] Clavel, Chloé and Ehrette, T. and Richard, Gaël, "Events Detection for an Audio-Based Surveillance System," IEEE International Conference on Multimedia and Expo, pages 1306-1309, 2005.
- [13] Naoya Takahashi, Michael Gygli, Beat Pfister, Luc Van Gool, "Deep Convolutional Neural Networks and Data Augmentation for Acoustic Event Detection," Interspeech 2016, pages 2982-2986, 2016.
- [14] Justin Salamon and Juan Pablo Bello, "Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification," IEEE Signal Processing Letters Journal, volume 24, pages 279-283, 2016.
- [15] Daniel S. Park , William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, Quoc V. Le, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," CoRR Journal, volume abs/1904.08779, 2019.
- [16] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, Kevin Wilson, "CNN Architectures for large-scale audio classification," International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017.
- [17] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, Marvin Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," IEEE ICASSP, pages 776-780, 2017.
- [18] Annamaria Mesaros, Aleksandr Diment, Benjamin E., Toni Heittola, Emmanuel Vincent, Bhiksha Raj, Tuomas Virtanen, "Sound event detection in the DCASE 2017 challenge," IEEE/ACM Transactions on Audio, Speech, and Language Processing, volume 27, pages 992-1006, 2017.
- [19] Qiuqiang Kong, Yong Xu, Iwona Sobieraj, Wenwu Wang, Mark D. Plumbley, "Sound Event Detection and Time-Frequency Segmentation from Weakly Labelled Data," IEEE/ACM Transactions on Audio, Speech and Language Processing, volume 27, no. 4, pages 777-787, 2019.
- [20] Turab Iqbal, Yong Xu, Qiuqiang Kong and Wenwu Wang, "Capsule Routing for Sound Event Detection," European Signal Processing Conference (EUSIPCO), pages 2255-2259, Rome, Italy, 2018.
- [21] G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci, A. Sarti, "Scream and gunshot detection and localization for audio-surveillance systems," IEEE Conference on Advanced Video and Signal Based Surveillance, pages 21-26, 2007.
- [22] Izabela L. Freire, Jose A. Apolinario Jr., "Gunshot detection in noisy environments," 7th International Telecommunications Symposium (ITS), 2010.
- [23] Cheung-Fat Chan, Eric W.M., "An abnormal sound detection and classification system for surveillance applications," European Signal Processing Conference, 2010.
- [24] Matti E. Ylikoski, Jussi O. Pekkarinen, Jukka P. Starck, Rauno J. Pääkkönen and Jukka S. Ylikoski, "Physical Characteristics of Gunfire Impulse Noise and its Attenuation by Hearing Protectors," Scandinavian Audiology Journal, volume 24, pages 3-11, 1995.
- [25] Dan-Ning Jiang, Lie Lu, Hong-Jiang Zhang, Jian-Hua Tao and Lian-Hong Cai, "Music type classification by spectral contrast feature," IEEE International Conference on Multimedia and Expo, volume 1, pages 113-116, 2002.
- [26] Chang-Hsing Lee, Jau-Ling Shih, Kun-Ming Yu, Jung-Mau Su, "Automatic music genre classification using modulation spectral contrast feature," IEEE International Conference on Multimedia and Expo, pages 204-207, 2007.
- [27] Justin Salamon, Christopher Jacoby, Juan Pablo Bello, "A Dataset and Taxonomy for Urban Sound Research," 22nd ACM International Conference on Multimedia, pages 1041-1044, Orlando, USA, 2014.
- [28] I. Lezhenin, N. Bogach and E. Pyshkin, "Urban Sound Classification using Long Short-Term Memory Neural Network," Federated Conference on Computer Science and Information Systems (FedCSIS), pages 57-60, 2019.
- [29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Deep residual learning for image recognition," Conference on Computer Vision and Pattern Recognition (CVPR), pages 770-778, 2016.

Detection of Linear Frequency Modulation, Phase-Coded and Multicarrier Radar Waveforms in Electronic Warfare Context

Antoine FOUCAULT

THALES Defense Mission Systems
Lab STICC, ENSTA Bretagne, CNRS
Brest, France
antoine.foucault@ensta-bretagne.org

Ali KHENCHAF

Lab STICC, ENSTA Bretagne, CNRS
Brest, France
ali.khenchaf@ensta-bretagne.fr

Cédric CORNU

THALES Defense Mission Systems
Brest, France
cedric.cornu@fr.thalesgroup.com

Fabrice COMBLET

Lab STICC, ENSTA Bretagne, CNRS
Brest, France
fabrice.comblet@ensta-bretagne.fr

Abstract—Interception of radar signals is analyzed. The waveforms of interest are the Linear Frequency Modulation (LFM), the Phase-Coded (PC) and multicarrier (MC) waveforms. The problematic is to perform detection with real-time requirement and the possibility to integrate on the longest pulse width (LFM). The well-known ambiguity function is proposed as quadratic time-frequency detector which is able to help waveform recognition among the analyzed set of waveforms. Its detection performances are evaluated and some estimators are proposed to facilitate the intrapulse estimation step.

Index Terms—Electronic Warfare, Electronic Support Measure, detection, radar waveforms.

I. INTRODUCTION

The context of the study is the Electronic Support Measure (ESM) applied on radar signal. That consists in intercepting the electromagnetic emissions and to identify the nature of the intercepted signal to identify the emitting radar [1].

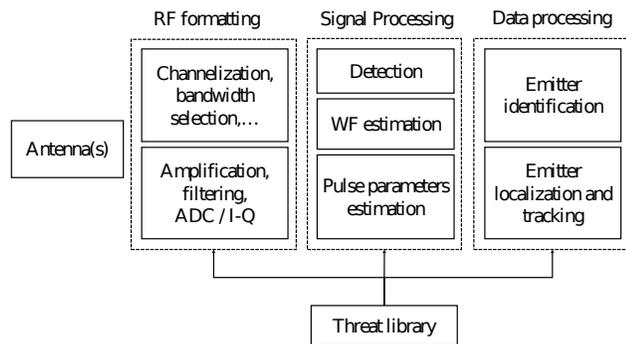


Fig. 1. General functions architecture of an ESM system.

The general functions of an ESM system are presented in Fig. 1. We deal with signal processing function and in particular detection one. The detection has to initiate the

following steps, particularly the waveform (WF) recognition of interest and intrapulse estimation. To deal with them, time-frequency (TF) detectors are chosen. Short-time FT or Fourier-based synchrosqueezing transform (FSST) [2] show great interest in LPI radar waveform recognition, in particular with the use of CNN [3]. However, detection has to be performed. Or the time integration (N samples) of these TF detectors is limiting to quasi-stationary WF, forcing to integrate on subpart of pulse duration. On the other hand, optimal time-frequency detector formulation is proposed in [4], using the cross Wigner-Ville distribution (WVD). The ambiguity function (AF) is linked to WVD but conserves causal property [5]. We will show in section II that AF performs energy concentration of LFM, PC and MC WF contrary to WVD. Section III will present the detection performances of the AF.

The set of WF $\mathcal{W} = \{1:\text{LFM}, 2:\text{PC}, 3:\text{MC}\}$ contains elementary LPI radar WF from which more complex radar WF could be designed [6]–[8]. The general expressions are, for all $k \in \llbracket 1, N \rrbracket$:

$$x_1[k] = A e^{i2\pi f_0 \frac{k}{f_s}} e^{i\phi_0} e^{i\pi\alpha \frac{k^2}{f_s^2}} \quad (1)$$

$$x_2[k] = A e^{i2\pi f_0 \frac{k}{f_s}} e^{i\phi_0} \sum_{p=0}^{N_m-1} \text{rect}\left(\frac{k - pN_{sm}}{N_{sm}}\right) e^{i\Phi[k]} \quad (2)$$

$$x_3[k] = \frac{A}{\sqrt{N_f}} e^{i2\pi f_0 \frac{k}{f_s}} \sum_{p=1}^{N_f} e^{i2\pi f_p \frac{k}{f_s}} e^{i\phi_p} \quad (3)$$

Where A is the amplitude, f_0 the carrier frequency, f_s the sampling frequency, ϕ_0 the origin phase. LFM: α is the slope of the chirp. PC: N_m is the number of moments of PC signals, N_{sm} the number of samples per moment, Φ is the vector of size $1 \times N_m$ containing the phase codes. MC: N_f is the num-

ber of subcarriers, the agility frequency $f_p = \left(p - \frac{N_f}{2}\right) \delta f$ and phase ϕ_p characterize the p^{th} subcarrier. The PC expression incorporates all the PC WF. The LFM and MC expressions are the simplest expressions of their own set.

II. AMBIGUITY FUNCTION PROPERTIES ON WAVEFORMS OF INTEREST

A. Definition and first results of the detector

Let consider N the number of observed samples as even. $N_I = N - n_\tau$ defines the number of integrated samples, with n_τ as delay. The detector based on AF is:

$$D_{\mathcal{W}}(n_\tau, f) = \sum_{k=0}^{N_I-1} x_{\mathcal{W}}[k + n_\tau] \times x_{\mathcal{W}}^*[k] e^{-i2\pi f \frac{k}{f_s}} \quad (4)$$

The AF is implementable with the FFT algorithm and calculation cost is $\mathcal{O}(N_f^2 \ln N_I)$. The results of the detector applied on \mathcal{W} are presented in the table I. The entire energy of LFM

TABLE I
MODULUS OF THE DETECTOR FOR THE \mathcal{W} SET OF WF

\mathcal{W}	f_m	$ D_{\mathcal{W}}(n_\tau, f_m) $
LFM	$n_\tau \frac{\alpha}{f_s}$	$N_I A^2$
PC	0	$A^2 \left N - N_m n_\tau + n_\tau \sum_{k=0}^{N_m-2} e^{i(\phi_{k+1} - \phi_k)} \right $
MC	0	$\frac{N_I}{N_f} A^2 \left \frac{\sin\left(\frac{\delta f}{f_s} \pi N_f n_\tau\right)}{\sin\left(\frac{\delta f}{f_s} \pi n_\tau\right)} \right $

signal is concentrated in a single frequency canal. It remains true for PC if $n_\tau \ll N_I$. For MC, it is periodically true. More explanations will be given in the next subsection. Simulations for the LFM and PC WF are presented in figure 2. The noise is supposed to be a Complex Circularly Symmetric Gaussian (CCSG) random variable $\mathcal{CN}(0, \sigma^2)$. We suppose the integration time to be inferior to the pulse width (PW). A threshold is added to illustrate the performances (calculation in section III).

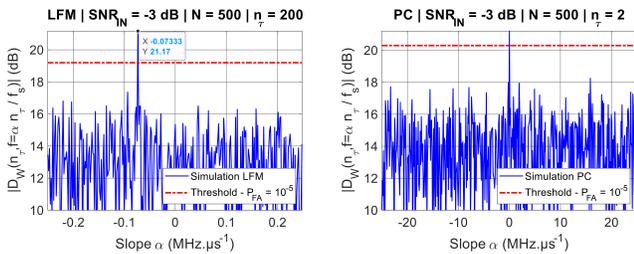


Fig. 2. Output of AF detector of noisy chirp and Barker signal

The estimator of $\hat{\alpha}$ is $\hat{\alpha} = \frac{f_s}{n_\tau} \max_f |D_{\mathcal{W}}(n_\tau, f)|$. Consequently, all WF with $\alpha \neq 0$ is directly recognized at the output of the detector. More n_τ is important, more the resolution and precision on $\hat{\alpha}$ is important.

B. Separate the PC and MC waveforms

The detection of the MC WF needs several calculations of the AF for different n_τ . Simulation is performed for $f_m = 0$ in figure 3. The output detector is periodically above the detection threshold for MC WF. Increasing n_τ for PC deteriorates the performances of the detector. A simple

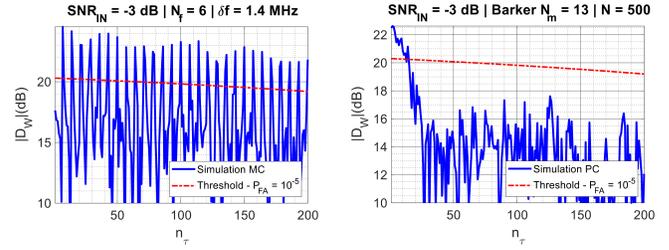


Fig. 3. Output of AF detector for noisy PC and MC signal - $N = 500$

criteria to separate WF based on frequency agility and on PC consists in verifying that detections occurs or not when n_τ increases. The main drawback of the detector is the necessity to calculate various n_τ contrary to the recognition of LFM. For the proposed simulation, about 1.55 million of operations have to be done for the calculation of the detector for a single n_τ . However, thanks to the causal property of the AF, it is possible to operate the calculation as soon as the sample arrive in the receiver channel. It can be shown that for a FPGA with a clock of 100 MHz, several tens of AF are calculable in parallel in real-time for $f_s = 50$ MHz and $N = 500$.

Outside $f_m = 0$, the MC WF has symmetrical artefacts that could be detected, due to interferences between subcarriers. If $p' \neq p$ with $(p', p) \in \llbracket 1, N_f \rrbracket^2$, then:

$$\left| D_{\mathcal{W}}\left(n_\tau, (p - p')\delta f\right) \right| = \frac{N_I}{N_f} A^2 \quad (5)$$

The symmetrical property allows to distinguish them from potential LFM signals.

C. Estimation of the carrier and subcarrier frequency

The detector $D_{\mathcal{W}}$ is complex. Analytical calculations on \mathcal{W} show that the arguments of the detector allow the estimation of frequency (cf. table II).

TABLE II
ARGUMENT OF THE DETECTOR FOR THE \mathcal{W} SET OF WF

\mathcal{W}	f_m	$\angle D_{\mathcal{W}}(n_\tau, f_m)$
LFM	$n_\tau \frac{\alpha}{f_s}$	$2\pi \left(f_0 + \frac{N}{2} \frac{\alpha}{f_s}\right) \frac{n_\tau}{f_s}$
PC	0	$2\pi f_0 \frac{n_\tau}{f_s}$
MC	0	$2\pi \left(\frac{\delta f}{2} (N_f - 1) + f_0\right) \frac{n_\tau}{f_s}$

Let notice that a single measure $D_{\mathcal{W}}$ for a unique n_τ is non-sufficient to unwrap the phase without ambiguity. To estimate the frequency, the detector $D_{\mathcal{W}}$ must be reiterated several times. The FT with n_τ as variable of integration could be proposed. We define the set \mathcal{S}_{12} in which the

$|D_{\mathcal{W}}(n_{\tau}, f_m)|$ for each n_{τ} is superior to the detection threshold. The carrier frequency estimator is, for LFM and PC signals:

$$\hat{f}_0 = \max_{\eta} \left(\text{FT} \left(\frac{D_{\mathcal{W}}(n_{\tau}, f_m) \in \mathcal{S}_{12}}{|D_{\mathcal{W}}(n_{\tau}, f_m)| \in \mathcal{S}_{12}} \right) (\eta) \right) - \frac{N\hat{\alpha}}{2f_s} \quad (6)$$

For the MC WF, the estimation of δf could be done with two methods. The first one consists in using the symmetrical artefacts seen in equation (5), but this is possible only if they could be detected. The second method uses only the detection on $f_m = 0$ thanks to the periodic maxima of the detector in function of the n_{τ} . On a periodical detection, several detections could be performed (1, 2 or 3). The maximum of each periodical detection is collected in the set called \mathcal{S}_3 .

$$\widehat{\delta f} = \max_{\eta \neq 0} \left(\text{FT} \{ |D_{\mathcal{W}}(n_{\tau}, 0)| \in \mathcal{S}_3 \} \right) (\eta) \quad (7)$$

The estimation of N_f is performed through several hypothesis of N_f . A final FFT gives the estimation of f_0 .

$$\hat{f}_0 = \max_{\eta} \left(\text{FT} \{ D_{\mathcal{W}}(n_{\tau}, 0) \in \mathcal{S}_3 \} \right) (\eta) - \frac{\widehat{\delta f}}{2} (\widehat{N}_f - 1) \quad (8)$$

The estimation of the MC parameters requires several periodical detections to work. Contrary to estimation of LFM and PC carrier frequency which could be performed with 10 calculations of AF, 30 to 40 AF calculations are required.

D. Integrate on longer duration

To evaluate the impact of the increasing of the time duration for ESM receivers, the calculation on the Signal to Noise Ratio (SNR) is useful, since the following steps will need a minimal SNR to work. Assuming that the integration time is equal to the pulse width and centered on it, the SNR at the output of the detector is equal to:

$$\text{SNR}_{\text{out}} = \frac{N_I \text{SNR}_{\text{in}}^2}{2\text{SNR}_{\text{in}} + 1} \quad (9)$$

where $\text{SNR}_{\text{in}} = \frac{A^2}{\sigma^2}$. Considering that integration is superior to the PW with $N_{I,\text{signal}} < N_I$, we have:

$$\text{SNR}_{\text{out}} = \frac{(N_{I,\text{signal}}^2 \text{SNR}_{\text{in}}^2)}{(N_I + 2N_{I,\text{signal}} \text{SNR}_{\text{in}})} \quad (10)$$

$$N_{I,\text{signal}} = \left\lfloor \frac{N_I}{N_{\text{PW},\text{sample}} \cdot r_{dc}} \right\rfloor$$

where r_{dc} is the duty cycle and $N_{\text{PW},\text{sample}}$ is the number of samples constituting the pulse width of the signal of interest. The formula is illustrated in figure 4.

Each break line characterizes the beginning or the ending of a pulse. In this example, to obtain the SNR_{out} of 17 dB that will be reached if the integration time is equal to the pulse width $N_{\text{PW},\text{sample}}$ and synchronizes with it, we need to increase the integrated samples from $N_I = 100$ to about $N_I = 10000$ integrated samples for a duty cycle $r_{dc} = 0.10$.

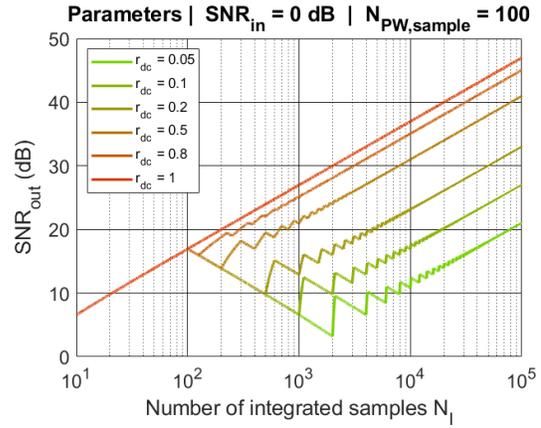


Fig. 4. Graph representing the variation of the output SNR in function of the chosen number of integrated samples, in function of the duty cycle r_{dc} with given number of sample for coding for a single pulse signal $N_{\text{PW},\text{sample}}$.

E. Multisignal detection

Increasing the time integration implies that several signals could be present in the receiver channel. If the signals are LFM, it could be easily showed that they are separated with any interference if their slope are different and could be distinguished. Their carrier frequency is measured with the previous method. Multi-PC signals could be detected thanks to the apparition of symmetrical artefacts that allow to estimate the number of superposed signals in $f_m = 0$. The multisignal case with the presence of one or several MC WF could be resolved thanks to the estimation of the δf . This requires more calculations with different hypotheses (symmetrical artefacts of PC waveforms, one MC, several MC).

F. Synthesis of the detector

The detector based on AF is able to detect \mathcal{W} WF. Two calculations of AF for $n_{\tau} \rightarrow 1$ and $n_{\tau} \in \left[\frac{N}{4}, \frac{N}{2} \right]$ provide the PC and LFM detections and the LFM recognition with adequate precision and resolution on the slope conjoint estimation. To be able to separate PC and MC WF and estimations of carrier and subcarrier frequencies, the AF detector has to be performed on tens of n_{τ} . These calculations could be done in parallel. Estimations are based on FT calculation, which is efficient estimators for the complex exponentials. Knowing that PC WF have $r_{dc} \sim 0.1$ and relative short PW compared to LFM and MC, the increase of the time integration benefits for that last two WF. In order to perform at least equal performances in detection with PC, the time integration has to be set with the longest expected PC PW and with its minimal expected r_{dc} . The multisignal detection could be implemented in an algorithm which demands more computations and so moves off real-time requirement.

III. DETECTION PERFORMANCES

This section focuses on the detector performances on \mathcal{W} WF.

A. Detection problematic of a single signal

Let suppose a single noised received signal. We suppose that the thermal noise is the main noise of the ESM receiver and that it could be modeled by a CCSG r.v. $n[k] \mathcal{CN}(0, \sigma^2)$ where $k \in \llbracket 1, N_I \rrbracket$. \mathbf{n} is supposed to be identically and independantly distributed. The signal is deterministic with unknown parameters. Under all these assumptions, the detection problematic consists in choosing between:

$$\begin{aligned} H_0 : & y[k] = n[k] \\ H_1 : & y[k] = x_W[k] + n[k] \end{aligned}, \quad k \in \llbracket 1, N_I \rrbracket \quad (11)$$

To evaluate detection performances, the probability of false alarm PFA and the probability of detection PD expressions are required.

B. PFA and PD expressions and analyses

The detector $|D_W|$ is applied on the received signal y . Thanks to the definition presented in equation (4), we have:

$$d = |D_W(n_\tau, f)| = \left| \sum_{j=0}^{N_I-1} y_1[j] y_2^*[j] \right| \quad (12)$$

where $y_1[j] = y[j + n_\tau]$ and $y_2[j] = y[j] e^{i2\pi f \frac{j}{f_s}}$. Let begin with the hypothesis H_0 .

1) PFA calculation:

a) *Exact analytical calculation:* Analytical calculations could be done in the H_0 hypothesis, since the random variable d could be recognized as a marginal probability of the off-diagonal elements of the Wishart distribution. The PDF for hypothesis H_0 is [9]:

$$p_{H_0}(d) = \frac{4d^{N_I}}{\Gamma(N_I) \sigma_d^2} K_{N_I-1} \left(\frac{2d}{\sigma_d^2} \right) \quad (13)$$

The PFA is then derived from the PDF thanks to the Mathematica software:

$$P_{FA}(u) = \int_u^{+\infty} p_{H_0}(x) dx = \frac{2u^{N_I}}{\Gamma(N_I)} K_{N_I} \left(\frac{2u}{\sigma_d^2} \right) \quad (14)$$

Where:

- K_ν is the modified Bessel function of second kind of order ν
- Γ is the gamma function
- $\sigma_d^2 = \frac{N_I}{2\sigma^4}$

However, the calculus consoms time to be performed, and for N_I higher than some hundreds, $\Gamma(N_I)$ and K_{N_I} are huge numbers, with for instance:

$$\Gamma(1000) \simeq 10^{3428} \quad K_{1000}(0.1) \simeq 10^{3865} \quad (15)$$

Consequently, to be closer to real-time application, a comparison is proposed between tables of exact PFA expression, calculated thanks to Mathematica software, and the approximate based on the modulus of the Central Limit Theorem (CLT) for d .

b) *CLT approximate:* The CLT applied on the detector D_W gives a complex Gaussian distribution. Then, in H_0 case, the modulus of that zero-mean complex Gaussian distribution $|D_W|$ is the well-known Rayleigh distribution:

$$p(d | H_0) = \frac{d}{\sigma_d^2} e^{-\frac{d^2}{2\sigma_d^2}} \quad (16)$$

Then, the PFA is:

$$P_{FA}(u) = e^{-\frac{u^2}{2\sigma_d^2}} \quad (17)$$

We propose a comparative simulation with the exact expression with the following characteristics, when effectively calculable:

- $N_I = [20, 50, 100, 250, 500, 1000, 2000]$
- The PFA of interest, called $P_{FA, \text{target}}$, is equal to the vector $P_{FA, \text{target}} = [10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}]$.
- The number of draws N_d is 30 millions.

That number of draws allows to have at least about 300 draws in the histogram area of interest, for the smallest PFA of interest, in order to allow a quite reasonable approximation of the PDF and then of the cumulative density function (CDF).

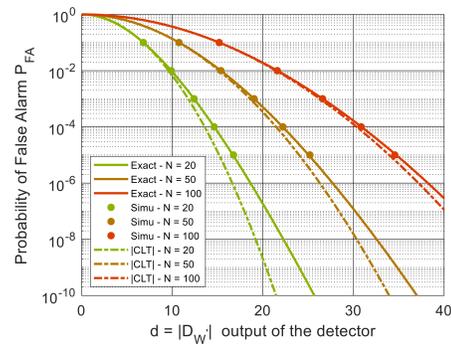


Fig. 5. Probability of false alarm calculated with exact expression (solid line), CLT approximate expression (dashed line) and results of the five simulated PFA represented by points, in function of the output of the detector / threshold.

The exact expression is then verified by the simulation (cf. Fig. 5), and error with the approximate expression are visible.

The CLT could be considered as a valid approximation if the relative error made on the PFA is acceptable. For instance, if we accept a relative error of 10%, then for an expected PFA of 10^{-5} , it is possible to integrate on the interval $N_I \in \llbracket 1000, \infty \rrbracket$ (cf. Fig. 6).

2) PD calculation:

Any exact expression is proposed for PD, since the required precision on the detection probability is not as important as false alarm. Indeed, 0.8, 0.90 or 0.99 could be acceptable values for detection probability. However, as we have a functional simulation, we will propose to evaluate the error introduce by the modulus of CLT approximate versus simulation PD.

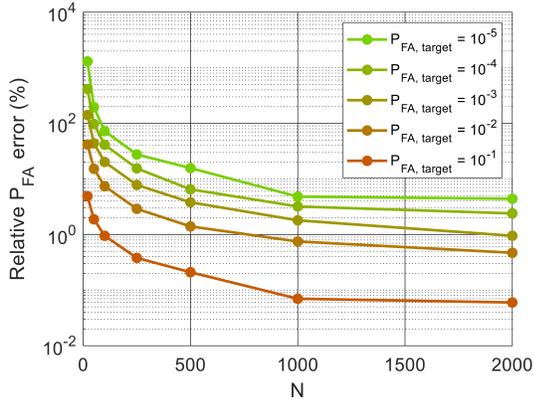


Fig. 6. Evaluation of the PFA error in function of the integrated sample N

The modulus of a non-zero-mean complex Gaussian distribution is the well-known Rician distribution:

$$p(d | H_1) = \frac{d}{\sigma_d^2} e^{-\frac{(d+\mu)^2}{\sigma_d^2}} I_0\left(\frac{\mu d}{\sigma_d^2}\right) \quad (18)$$

Where $\mu = E(|y_1|) = E(|y_2|)$. Consequently, the PD is:

$$P_D(u) = Q_1\left(\frac{\mu}{\sigma_d}; \frac{d}{\sigma_d}\right) \quad (19)$$

Where Q_1 is the Marcum Q-function.

Using the previous parameters of the simulation, (9) given in property 3 and the same relative error for detection probability this time, we obtain Fig. 7.

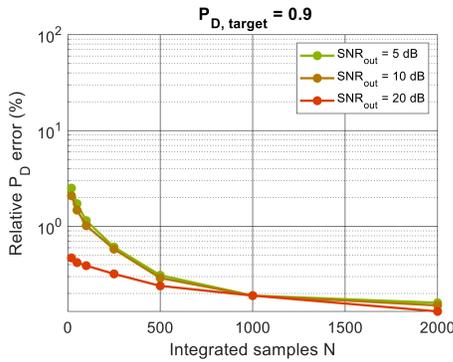


Fig. 7. Graph representing the relative error on the detection probability calculus in function of the expected SNR at the output.

The SNR at the input is calculable using (9). The relative error on the detection probability remains under several percent. Consequently, the approximate detection probability could be used without any restriction on N_I in ESM context.

C. Receiver Operating Characteristic (ROC)

The ROC graph of the detector is drawn in Fig. 8 thanks to the previous calculations.

If the requirements specification are limited by the detector, then the ROC and Fig. 8 is used to define the minimal number of integrated samples to obtain required maximal false alarm with consequent detection probability.

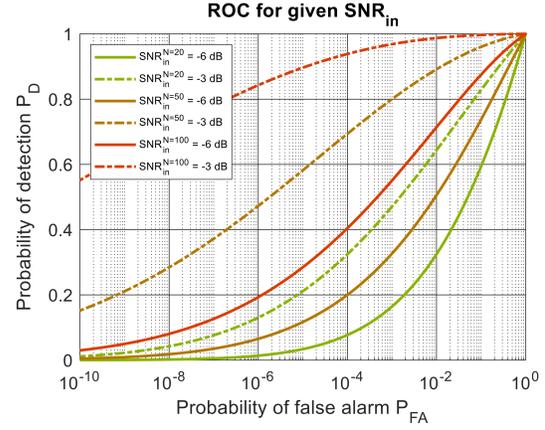


Fig. 8. ROC of $|D_W|$ with the input SNR_{in} (solid line for $SNR_{in} = -6dB$ and dashed line for $SNR_{in} = -3dB$) and the number of integrated N_I as parameters (colors)

CONCLUSION

This work is done in ESM context. The ambiguity function was chosen as quadratic time-frequency detector to perform detection on LFM, PC and MC waveforms. AF is chosen for its properties for allowing WF recognition. Conjoint estimation of the slope parameter is very useful for LFM recognition, and could be done on a single calculation. However, to go further, several AF calculations are required. Estimations of carrier and subcarrier frequencies could be done thanks to the argument of the detector. Real-time computation could be performed since the AF calculation for the required delays could be done in parallel and at the arrival of each sample thanks to the causal property of AF. AF detector allows to increase the time integration and the conservation of its properties on the analyzed waveforms. In multisignal case, the AF detector is able to distinguished the analyzed waveforms in detriment of the real-time omputation. Detection performances of the detector are analyzed.

REFERENCES

- [1] A. De Martino, "Introduction to Modern EW Systems," 2nd ed., Artech House, Norwood CA, 2018.
- [2] T. Oberlin and S. Meignen, V. Perrier, "The Fourier-Based Synchrosqueezing Transform", IEEE ICASSP 2014.
- [3] G. Kong and V. Koivunen, "Radar Waveform Recognition using Fourier-Based Synchrosqueezing Transform and CNN", IEEE 8th CAMSAP 2019.
- [4] P. Flandrin, "A Time-Frequency Formulation of Optimum Detection", IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 36, No. 9, pp 1377-1384, September 1988.
- [5] P. E. Pace, "Detecting and Classifying Low Probability of Intercept Radar", 2nd ed., Artech House, 2009.
- [6] S. D. Blunt and E. L. Mokole, "Overview of Radar Waveform Diversity", IEEE A&E Systems Magazine, November 2016.
- [7] N. Levanon and E. Mozeson, "Radar Signals", John Wiley & Sons, 2004.
- [8] A. De Maio, A. Farina, "Waveform Diversity: Past, Present, and Future", Unclassified Report RTO-EN-SET-119(2009), November 2009.
- [9] J-S. Lee and K.W. Hoppel, S.A Mango and A.R. Miller, "Intensity and Phase Statistics of Multilook Polarimetric and Interferometric SAR Imagery", IEEE Transactions on Geoscience and Remote Sensing, Vol.32, No.5, pp 1017-1028, September 1994.

Robust Source Number Estimation Based on Denoising Preprocessing

Koichi Ichige, Shohei Hamada
Dept. Elec. & Comp. Eng., Yokohama Nat'l Univ.
Yokohama-shi, Kanagawa, Japan.

Katsuhisa Kashiwagi, Nobuya Arakawa, Ryo Saito
Murata Manufacturing Co., Ltd.
Nagaokakyo-shi, Kyoto, Japan.

Abstract—This paper presents an accurate source number detection method for array antennas or MIMO (Multi-Input Multi-Output) radars. DOA (Direction of arrival) estimation is one of the significant techniques in high-speed wireless communication and radar imaging. Most of the representative DOA estimation methods requires the source number information in advance, however the classical methods AIC or MDL are often used. Those methods often fails to estimate source number in severe environments like low SNR or large transmission power difference. Recall that received signals are often bandlimited or narrowband signals, the proposed method first applies denoising preprocessing by removing undesired components, and compare the original and denoised signal information. Performance of the proposed method is evaluated through computer simulation.

Index Terms—source number detection, array antenna, MIMO radar, DOA estimation

I. INTRODUCTION

DOA (Direction of arrival) estimation is one of the key techniques in high-speed wireless communication and radar imaging [1] – [3]. Array antenna and MIMO (Multi-Input Multi-Output) are key techniques which need to know wave propagation environment for estimating DOAs, and to suppress interference waves in order to establish high-speed wireless communication technology. For array antenna techniques, high resolution DOA estimation algorithms like MUSIC (Multiple Signal Classification) method [4] or ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) method [5] have attracted attention because of their brilliant properties in estimating DOAs. However there exists an inherent problem that those methods require a correct number of arrival signals in advance, and they give wrong directions if the number is incorrect.

As for estimation algorithms of the number of arrival signals, AIC (Akaike Information Criterion) [6] and MDL (Minimum Description Length) [7] are regarded as classical estimation algorithms which are still effective, meanwhile the bootstrap method based on statistics [8] and PDL (Predictive Description Length) method based on information theory [9] have been recently studied for better estimation. Bootstrap method aims at obtaining more accurate data by resampling observed signals but it does not always give accurate estimates in spite of long computation time; indeed it takes quite long time for resampling itself and evaluating the accuracy of resampled data. PDL method aims at approximating time-varying wave propagation environment by smoothing param-

eter. But this method will be corrupted by some sudden occasional data like impulse due to the PDL principle composed by the successive sum of its log-likelihood functions from the past. We have already presented a source number estimation method using pre-estimated signal subspace [10], but its computational cost is still large due to beamformer processing and eigendecomposition. That is why AIC and MDL methods are still often used for source number detection problem. In fact their computation procedures are simple and therefore source numbers are estimated within a very short time, but the estimation accuracy of them becomes worse in severe communication environment like lower SNR, smaller number of snapshots, or closely spaced sources. We desire an accurate and robust estimation algorithm while preserving low computational cost.

Recall that we often deal with bandlimited or narrowband signals in array antenna or radar applications or FMCW (Frequency Modulated Continuous Wave) signals for radar applications, and assume wideband noise environments like AWGN (Additive White Gaussian Noise) or Random noise which cover wide frequency range. There will be a difference between the original and denoised signal information if we apply denoising preprocessing by removing undesired components. Such difference could be more clearly observed if we modify the signal information in the covariance matrix and its eigenvalues.

In this paper, we develop an accurate and robust source number detector for array antenna and FMCW-MIMO radars. We first apply denoising preprocessing by removing undesired frequency components by bandpass filtering, and then compare the original and denoised signal information through the covariance matrix and its eigenvalues. Then we detect the number of arrivals based on the difference of the eigenvalues. The proposed method is evaluated through computer simulation for various environments of snapshots, SNR and incident waves from closely spaced sources in comparison with the conventional methods.

II. PRELIMINARIES

This section prepares mathematical models and representations of signals, vectors and spaces.

A. Signal Model

Assume that K incident waves are impinged from the direction θ_k ($k = 1, 2, \dots, K$) and received by M -element uniform linear array (ULA) under an AWGN environment. Using the input signal $x_m(t)$ for the m -th element at the time t , the input signal vector $\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_M(t)]^T$ can be written as

$$\begin{aligned} \mathbf{x}(t) &= \sum_{\ell=1}^K \mathbf{a}(\theta_\ell) s_\ell(t) + \mathbf{n}(t) \\ &= \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t), \end{aligned} \quad (1)$$

where $\mathbf{a}(\theta_\ell)$, $s_\ell(t)$ and $\mathbf{n}(t)$ denote the array steering vector of k -th incident wave, the complex waveform of k -th wave, and the noise vector whose elements are with zero mean, respectively. The array steering matrix \mathbf{A} and the waveform vector $\mathbf{s}(t)$ in (1) are given as

$$\begin{aligned} \mathbf{A} &= [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \dots, \mathbf{a}(\theta_K)], \\ \mathbf{s}(t) &= [s_1(t), s_2(t), \dots, s_K(t)]^T, \end{aligned}$$

where $[\cdot]^T$ denotes the transpose of vectors or matrices. The covariance matrix \mathbf{R}_{xx} of $\mathbf{x}(t)$ is given by

$$\begin{aligned} \mathbf{R}_{xx} &= E[\mathbf{x}(t)\mathbf{x}^H(t)] \\ &= \mathbf{A}E[\mathbf{s}(t)\mathbf{s}^H(t)]\mathbf{A}^H + E[\mathbf{n}(t)\mathbf{n}^H(t)] \\ &= \mathbf{A}\mathbf{S}\mathbf{A}^H + \sigma^2\mathbf{I}, \end{aligned}$$

where $[\cdot]^H$ denotes the conjugate transpose of vectors or matrices, $\mathbf{S} \triangleq E[\mathbf{s}(t)\mathbf{s}^H(t)]$, $E[\cdot]$ denotes the statistical expectation, σ^2 is the noise power, and \mathbf{I} is the identity matrix. The covariance matrix \mathbf{R}_{xx} is an Hermitian matrix, and its eigenvalues λ_{x_k} are defined to satisfy the following inequality:

$$\lambda_{x_1} \geq \lambda_{x_2} \geq \dots \geq \lambda_{x_K} > \lambda_{x_{K+1}} = \dots = \lambda_{x_M} = \sigma^2.$$

B. Conventional Source Number Estimation Methods

As the representative source number estimation methods, AIC [6] and MDL [7] are formulated in this subsection.

The cost function of AIC is given by

$$\begin{aligned} \text{AIC}(k) &= -2N \log \left\{ \left(\prod_{i=k+1}^M \lambda_i \right) / \left(\frac{1}{M-k} \sum_{i=k+1}^M \lambda_i \right)^{(M-k)} \right\} \\ &\quad + k(2M - k). \end{aligned} \quad (2)$$

The AIC estimate for the number of sources is the value of k when (2) takes the smallest value.

In a similar manner, MDL gives its estimation value k when (3) takes the smallest value.

$$\begin{aligned} \text{MDL}(k) &= N \log \left\{ \left(\prod_{i=k+1}^M \lambda_i \right) / \left(\frac{1}{M-k} \sum_{i=k+1}^M \lambda_i \right)^{(M-k)} \right\} \\ &\quad + \frac{k(2M - k)}{2} \log N. \end{aligned} \quad (3)$$

III. PROPOSED METHOD

This section explains the proposed source number estimation method for the cases of general narrowband and FMCW signals.

A. Case of Narrowband Signals

First we assume the case of K narrowband sources. As prepared in Section II, we have the input signal vector $\mathbf{x}(t)$, its covariance matrix \mathbf{R}_{xx} and its eigenvalues λ_{x_k} . We try to denoise the input signal $x_m(t)$ of m -th element into $y_m(t)$ by the following frequency domain filtering procedure.

$$\begin{aligned} X_m(f) &= \text{FFT}[\{x_m(t)\}_{t=1}^N], \quad m = 1, 2, \dots, M, \\ Y_m(f) &= \text{BPF}[X_m(f)], \quad m = 1, 2, \dots, M, \\ y_m(t) &= \text{IFFT}[\{Y_m(f)\}_{f=1}^N], \quad m = 1, 2, \dots, M, \end{aligned}$$

where $\text{FFT}[\cdot]$ and $\text{IFFT}[\cdot]$ denotes the fast Fourier transform (FFT) and the inverse FFT operators, respectively. $\text{BPF}[\cdot]$ is a band-pass filtering operator which extracts only desired frequency components while suppressing undesired components. The denoised array input vector $\mathbf{y}(t)$ and its covariance matrix \mathbf{R}_{yy} are respectively written as

$$\begin{aligned} \mathbf{y}(t) &= [y_1(t), y_2(t), \dots, y_M(t)]^T, \\ \mathbf{R}_{yy} &= E[\mathbf{y}\mathbf{y}^H] \simeq \frac{1}{N} \sum_{i=1}^N \mathbf{y}\mathbf{y}^H. \end{aligned}$$

where N denotes the number of time snapshots. The eigenvalues λ_{y_k} of the matrix \mathbf{R}_{yy} are defined to satisfy the following inequality:

$$\lambda_{y_1} \geq \lambda_{y_2} \geq \dots \geq \lambda_{y_K} > \lambda_{y_{K+1}} = \dots = \lambda_{y_M}.$$

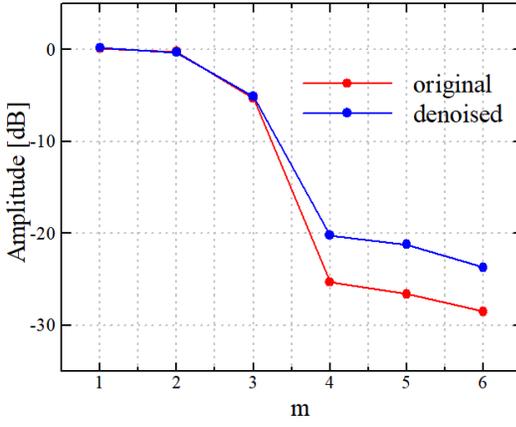
Comparing the eigenvalues λ_{x_k} and λ_{y_k} , they take almost the same signal eigenvalues for $k = 1, 2, \dots, L$ but take different noise eigenvalues for $k = L + 1, \dots, K$ because of denoising. Figure 1 shows an example distribution of the eigenvalues. We see from Fig. 1 that the signal eigenvalues are almost the same but the noise eigenvalues are different. Based on this observation, we define δ_m as the ratio of the eigenvalues, i.e.,

$$\delta_m = \frac{\lambda_{x_m}}{\lambda_{y_m}}, \quad m = 1, 2, \dots, M, \quad (4)$$

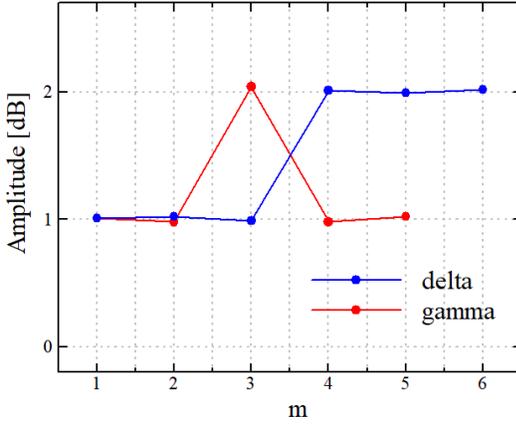
which will be almost one for $m = 1, 2, \dots, K$ because the signal eigenvalues are almost the same, but will be more than one for $m = K + 1, K + 2, \dots, M$ because the noise eigenvalues are different between λ_{x_m} and λ_{y_m} . Furthermore, we prepare the ratio of the parameters δ_{m+1} and δ_m as

$$\gamma_m = \frac{\delta_{m+1}}{\delta_m}, \quad m = 1, 2, \dots, M - 1, \quad (5)$$

which will be more than one when m is equal to the source number, but will be almost one when m is not equal to the source number. Therefore, we can estimate the source number by m where the γ_m in (5) takes the maximum value.



(a) The eigenvalues λ_{x_m} and λ_{y_m}



(b) δ_m and γ_m

Fig. 1. Example of (a) the original eigenvalues λ_{x_m} and the denoised eigenvalues λ_{y_m} , and (b) the parameters δ_m and γ_m as a function of m , in case of $M = 6$ and $K = 3$.

B. Case of FMCW Signals

In case of FMCW signals used for radar modules, the source number estimation procedure is different from the case of narrowband cases. We employ a downsampling scheme as follows. Note that we assume to use the annihilating filter (AF) [11] for DOA estimation.

First we prepare the downsampled version $\tilde{x}_m(t)$ of the original input $x_m(t)$ at the m -th element, i.e.,

$$\begin{aligned} \tilde{x}_k(t) &= [x_m(t)]_{\downarrow W} = x_m(Wt), \quad m = 1, 2, \dots, M, \\ \tilde{X}_m(f) &= \text{FFT}[\{\tilde{x}_m(t)\}_{t=1}^{N/W}], \quad m = 1, 2, \dots, M, \end{aligned}$$

where W is the downsampling rate, and $[\cdot]_{\downarrow W}$ denotes the W -times downsampling operator. After that we extract the target distance bin f_0 by the spectrum peak search of $X_m(f)$. Collect the peak values $X_m(f_0)$ for $m = 1, 2, \dots, M$, we formulate

the $(M - K) \times (K + 1)$ matrix \mathbf{C} as

$$\mathbf{C} = \begin{bmatrix} X_{K+1}(f_0) & X_K(f_0) & \cdots & X_1(f_0) \\ X_{K+2}(f_0) & X_{K+1}(f_0) & \cdots & X_2(f_0) \\ \vdots & \vdots & \ddots & \vdots \\ X_M(f_0) & X_{M-1}(f_0) & \cdots & X_{M-K}(f_0) \end{bmatrix},$$

Applying the singular value decomposition (SVD) to the matrix \mathbf{C} , we have the singular values

$$\lambda_{x_1} \geq \lambda_{x_2} \geq \dots \geq \lambda_{x_K} > \lambda_{x_{K+1}} = \dots = \lambda_{x_M},$$

where $\tilde{K} = \max\{M - K, K + 1\}$.

On the other hand, we denoise the original input $x_m(t)$ similarly to the case of narrowband signals:

$$\begin{aligned} X_m(f) &= \text{FFT}[\{x_m(t)\}_{t=1}^N], \quad m = 1, 2, \dots, M, \\ Y_k(f) &= \text{BPF}[X_m(f)], \quad m = 1, 2, \dots, M, \\ y_k(t) &= \text{IFFT}[\{Y_m(f)\}_{f=1}^N], \quad m = 1, 2, \dots, M. \end{aligned}$$

Then the denoised signal $y_m(t)$ is downsampled, i.e.,

$$\begin{aligned} \tilde{y}_m(t) &= [y_m(t)]_{\downarrow W} = y_m(tW), \quad m = 1, 2, \dots, M, \\ \tilde{Y}_m(f) &= \text{FFT}[\{\tilde{y}_m(t)\}_{t=1}^{N/W}], \quad m = 1, 2, \dots, M. \end{aligned}$$

Similarly to the noisy case, we extract the target distance bin f_0 by the spectrum peak search of $Y_m(f)$. Collect the peak values $Y_m(f_0)$ for $m = 1, 2, \dots, M$, we formulate the $(M - K) \times (K + 1)$ matrix $\tilde{\mathbf{C}}$ as

$$\tilde{\mathbf{C}} = \begin{bmatrix} Y_{K+1}(f_0) & Y_K(f_0) & \cdots & Y_1(f_0) \\ Y_{K+2}(f_0) & Y_{K+1}(f_0) & \cdots & Y_2(f_0) \\ \vdots & \vdots & \ddots & \vdots \\ Y_M(f_0) & Y_{M-1}(f_0) & \cdots & Y_{M-K}(f_0) \end{bmatrix}.$$

Applying the singular value decomposition (SVD) to the matrix $\tilde{\mathbf{C}}$, we have the singular values

$$\lambda_{y_1} \geq \lambda_{y_2} \geq \dots \geq \lambda_{y_K} > \lambda_{y_{K+1}} = \dots = \lambda_{y_M}.$$

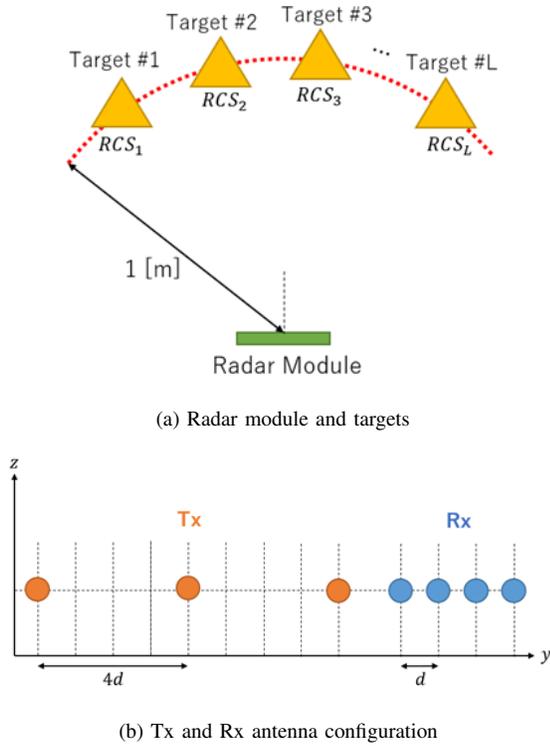
Similarly to the narrowband case, we can estimate the number of sources by using δ_k in (4) and γ_k in (5).

IV. NUMERICAL EXAMPLES

A. Simulation Specifications

The proposed source number detection method is evaluated through computer simulation of FMCW signals. Radar module and targets are located as in Fig. 2(a). Multiple targets are placed on a circle arc, which are at the same distance from the radar module.

Specifications of simulation is as shown in Table I. We assume 3×4 FMCW-MIMO radar system which is equivalent to $3 \times 4 = 12$ element ULA as in Fig. 2(b). The first target is used as the reference to SNR, and the other targets are with the radar cross section (RCS) difference as in Table I.



(a) Radar module and targets

(b) Tx and Rx antenna configuration

Fig. 2. Simulation model.

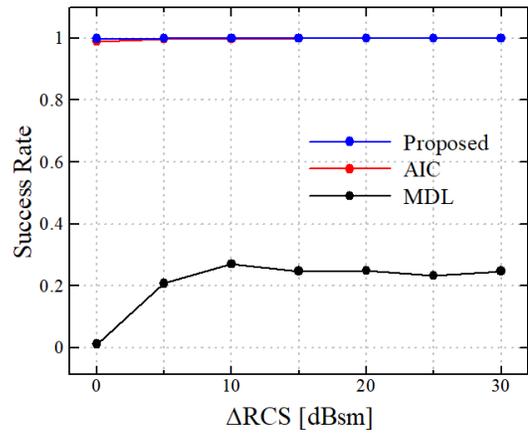
TABLE I
SPECIFICATIONS OF SIMULATION.

Scenario	#1	#2	#3	#4
Number of Tx elements	3			
Number of Rx elements	4			
Carrier frequency	79 GHz			
Frequency bandwidth	3.16 GHz			
Rx element interval	0.5λ			
Distance to targets	1.0 m			
Sampling frequency	2.75 MHz			
Downsampling rate	32			
Number of snapshots	128			
Number of trials	1,000			
Number of Targets	2	4	2	5
RCS [dB]	0 to 30	10	30	
SNR [dB]	-10	-20 to 0		
DOAs [deg] ($K = 2$)	-32, 38			
DOAs [deg] ($K = 4$)	-51, -28, 14, 56			
DOAs [deg] ($K = 5$)	-44, -25, -8, 14, 26			

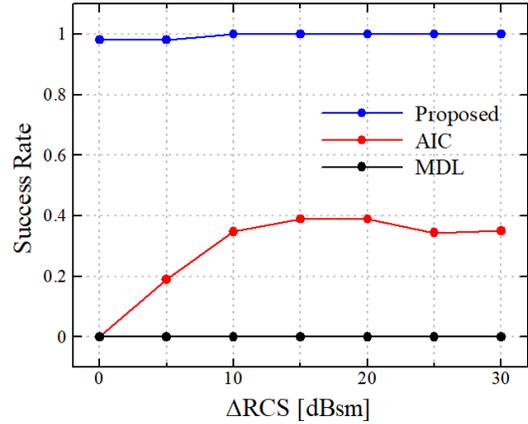
B. Source Number Estimation Accuracy

Figure 3 shows the behavior of the estimation success rate in case of the scenario #1 and #2 for various values of ΔRCS . We see from Fig. 3 that the proposed method gives almost 100% success rate while the conventional AIC and MDL methods often fail to estimate source number in severe conditions. AIC can accurately estimate in the scenario #1 but does not work well in the scenario #2.

Figure 4 shows the behavior of the estimation success rate in case of the scenario #3 and #4 for various values of SNR. We



(a) Case of scenario #1



(b) Case of scenario #2

Fig. 3. Behavior of the estimation success rate as a function of ΔRCS .

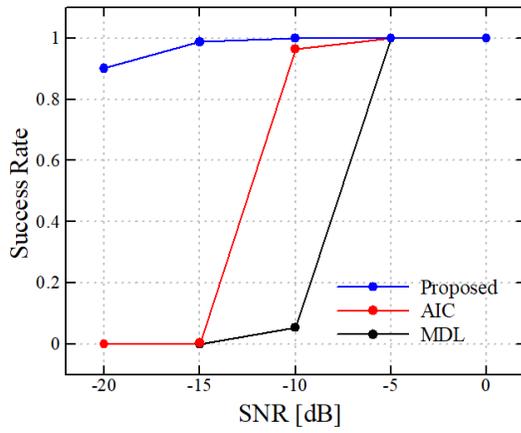
see from Fig. 4 that the proposed method gives almost 100% success rate in high SNR, and holds more than 90% success rate in low SNR environments. The conventional AIC and MDL methods work well in high SNR but usually fail in low SNR environments. We overall confirmed that the proposed method gives much higher success rate than the conventional methods.

V. CONCLUDING REMARKS

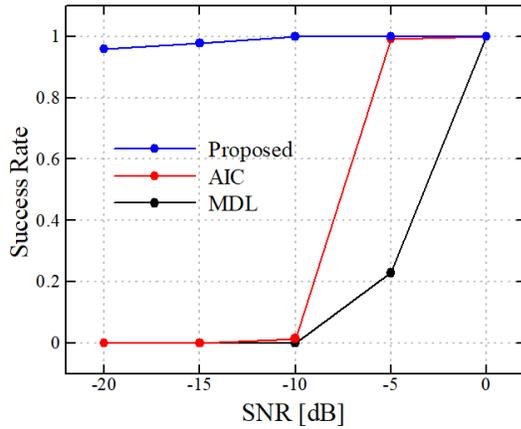
We proposed a robust source number estimation method based on denoising preprocessing. The proposed method achieves much higher estimation accuracy in severe environments like large ΔRCS or low SNR, where the conventional methods often fail to estimate. Further enhancing the estimation accuracy by more minute denoising remains as one of future studies.

ACKNOWLEDGMENT

This work is partly supported by Japan society for the promotion of science (JSPS) grant-in-aid for scientific research #20K04500. The authors sincerely thank to their support.



(a) Case of scenario #3



(b) Case of scenario #4

Fig. 4. Behavior of the estimation success rate as a function of SNR.

REFERENCES

- [1] T. E. Tuncer and B. Friedlander, "Classical and modern direction of arrival estimation," Academic Press, 2009.
- [2] M. I. Skolnik, "Introduction to radar systems, 3rd Ed.," McGraw-Hill, 2001.
- [3] H. L. Van Trees, "Optimum array processing: Part IV of detection, estimation, and modulation theory," Wiley, 2002.
- [4] R. O. Schmidt, "Multiple Emitter Location and Signal Parameter Estimation," IEEE Trans. Antennas and Propagation, vol.34, pp.276-280, Mar. 1986.
- [5] R. Roy and T. Kailath, "ESPRIT—Estimation of Signal Parameters via Rotational Invariance Techniques," IEEE Trans. Acoust., Speech, and Signal Processing, vol.37, pp.984-995, July 1989.
- [6] M. Wax and T. Kailath, "Detection of Signals by Information Theoretic Criteria," IEEE Trans. Acoust., Speech, and Signal Processing, vol.33, pp.387-392, Apr. 1985.
- [7] M. Wax and I. Ziskind, "Detection of the Number of Coherent Signals by the MDL Principle," IEEE Trans. Acoust., Speech, and Signal Processing, vol.37, pp.1190-1196, Aug. 1989.
- [8] R. F. Bricich, A. M. Zoubir and P. Pelin, "Detection of Sources Using Bootstrap Techniques," IEEE Trans. Signal Processing, vol.50, pp.206-215, Feb. 2002.
- [9] S. Valaee and P. Kabal, "An Information Theoretic Approach to Source Enumeration in Array Signal Processing," IEEE Trans. Signal Processing, vol.52, pp.1171-1178, May 2004.
- [10] Y. Ishikawa, K. Ichige, H. Arai, "Accurate Source Number Detection Using Pre-Estimated Signal Subspace," IEICE Trans. Communications, vol. E89-B, no. 12, pp. 3257-3265, Dec. 2006.
- [11] Y. Pan, G. Q. Luo, H. Jin and W. Cao, "Direction-of-Arrival Estimation With ULA: A Spatial Annihilating Filter Reconstruction Perspective," IEEE Access, vol. 6, pp. 23172-23179, May 2018.

Learning Entropy of Adaptive Filters via Clustering Techniques

Ivo Bukovsky*, Gejza Dohnal, Pavel Steinbauer, Ondrej Budik
Czech Technical University in Prague
Prague, Czech Republic
*ivo.bukovsky@fs.cvut.cz, orcid.org/0000-0002-7715-2841

Kei Ichiji, Homma Noriyasu
Tohoku University
Department of Health Science
980-8575, Sendai, Japan

Abstract — Learning Entropy (LE) was initially introduced as a measure for sample point novelty by unusually large learning effort of an online learning system. The key concept is that LE is based on pre-training and further online learning, and the novelty measure is not necessarily correlated to the prediction error. Most recently, the idea of LE was revised as a novel non-probabilistic, i.e., machine-learning-based information measure. This measure is high when a learning system is not familiar with a given data point, so the learning activity to learn novel data points is unusual (regardless of the prediction error), i.e., the learning increments display unusual patterns during adaptation. In this paper, we propose concepts of the learning state and the learning state space so that LE can be approximated via neighbourhood analysis in the learning space. Further, two novel clustering-based techniques for approximation of sample point LE are proposed. The first one is based on the sum of K nearest neighbour distances. The second one is based on multiscale neighbourhood cumulative sum. Also, we preprocess the learning space with dimensionality reduction that is promising for research of LE even with neural networks and potentially with deep neural networks. The performance of novelty detection with the clustering-based sample point LE with dimensionality reduction is compared to the original algorithms of LE, and its potentials are discussed.

Keywords — novelty detection, unsupervised anomaly detection, time series, adaptive filters, dynamic detection scheme, clustering

1 INTRODUCTION

Novelty detection has gained severe importance in all areas where sensor signals are measured and evaluated. E.g., in vibration analysis for structural anomaly and construction damage detection, predictive maintenance, network or device intrusion detection, biomedical data analysis, i.e., in sensor measurements and data processing from dynamical systems in general. Also, instant novelty detection may be important when tracking and prediction depend on actual sensor data.

The unsupervised anomaly detection methods, i.e., where the type of anomaly is not known and where the anomaly can not be annotated before its detection, can be distinguished according to many various principles and types of applications. Statistical methods, e.g., [1]–[3] or machine learning methods, e.g., [4], [5], or their variations, e.g., [6], can be utilized. Also, the environments are non-stationary and complicated, so dynamical detection schemes need to be utilized as it has been apparent for the last decades, e.g., [6]–[8].

In this paper, we introduce a significant extension to the Learning Entropy (LE), i.e., the machine-learning based anomaly detection for adaptive filters, [9]–[13] and references therein. The new extension consists of dimensionality reduction and clustering in learning state space for the estimation of LE. Thus, the concept of clustering for the estimation of LE is proposed, and this also

makes LE estimation better suitable for neural networks and deep networks than the originally proposed algorithms.

The paper is organized as follows. Section 2 recalls the original concept and algorithms of LE in connotation to adaptive predictors (filters) in general. Subsection 3.1 introduces the concepts of learning state and learning state space. Subsections 3.2 and 3.3 present two clustering methods for simpler predictors with a low number of adaptive parameters such as for linear or polynomial adaptive filters. Then, subsection 3.4 proposes dimensionality reduction as pre-processing for more complicated learning systems with a large number of adaptive parameters, i.e., such as for neural networks and deep learning. Section 4 presents typical results on novelty detection performance on a dynamical system and on respiration time series.

In mathematical notations, k as in $y(k)$ stands for the discrete index of time that is dropped unless necessary for clarity, i.e. $y = y(k)$, vectors are in small bold, matrices are in capital bold, and n generally denotes the count of elements such as n_w stands for the count of elements of \mathbf{w} , etc.

The full term Approximate Individual Sample Point Learning Entropy (AISLE as in [11]) is shortened further as the sample point LE or just the LE in this paper.

2 RELATED BACKGROUND

The main idea behind LE is to detect the novelty of individual data samples by quantification how unusual is the actual learning behaviour of a learning system. The typical learning systems can be incrementally learning adaptive filters, e.g., via gradient descent (LMS, NLMS, e.g. [14]) or Kalman filter based update rules (RLS [15]). Also, the learning system can be a more complicated model, such as a neural network.

In this paper, a learning system is an adaptive filter that can be written as a predictor of a general form as

$$\tilde{y}(k+h) = f(\mathbf{x}(k), \mathbf{w}(k)), \quad (1)$$

where \tilde{y} stands for the predicted value, k is the discrete time index (for constant sampling), h is the prediction horizon [samples], \mathbf{w} represents a vector of all adaptive parameters (weights) whose count is n_w , and \mathbf{x} is the input vector to a mapping function $f(\cdot)$. The general incremental learning is the adaptation of the parameters \mathbf{w} as follows

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \Delta \mathbf{w}(k), \mathbf{w} \in \mathbb{R}^{n_w} \quad (2)$$

where $\Delta \mathbf{w}(k)$ represents the vector of all learning increments at sample time k , n_w is the length of $\Delta \mathbf{w}(k)$, and $\Delta \mathbf{w}(k)$ can be calculated by some applicable incremental supervised learning algorithm.

Originally in [10], [11], LE was practically approximated via a cumulative count of unusually large magnitudes of weight increments over all adaptive parameters and over various detection sensitivities (because practically its optimal value is unknown) as follows

$$E_A(k) = \frac{1}{n_\alpha \cdot n_w} \sum_{j=1}^{n_\alpha} \sum_{i=1}^{n_w} h(|\Delta w_i(k)| > \alpha_j \cdot \overline{|\Delta w_i|}), \quad (3)$$

where $E_A(k) \in \langle 0, 1 \rangle$ denotes the sample point LE approximated by original algorithm in [10], [11] (i.e. AISLE), α_j is a scalar from the user-proposed vector of real positive detection sensitivities $\alpha_j \in \alpha \in \mathbb{R}_{>0}^{n_\alpha}$, n_w is the length of vector $\Delta \mathbf{w}$, $\overline{|\Delta w_i|}$ is the recent average magnitude of i^{th} weight increment over its recent M -sample history, and $h(\cdot)$ is a binary detection function so that

$$h(\text{TRUE}) = 1 \text{ and } h(\text{FALSE}) = 0. \quad (4)$$

Later in [12], the alternative approximation of LE was proposed directly via Z-scoring of weight update magnitudes and their cumulative count over multiple biases (similarly to detection sensitivities) as follows

$$E_{ADirect}(k) = \frac{1}{n_\beta \cdot n_w} \sum_{j=1}^{n_\beta} \sum_{i=1}^{n_w} h(z(|\Delta w_i(k)|) > \beta_j), \quad (5)$$

where $E_{ADirect}$ is the sample point LE approximated by the direct z-scoring algorithm, β_j is one instance of detection biases that are heuristically defined by vector β of real positive biases, i.e. $\beta_j \in \beta \in \mathbb{R}_{>0}^{n_\beta}$, $z(\cdot)$ denotes the z-scoring operation where the mean and standard deviation are drawn from the recent M -sample long learning history \mathbf{S} , and $h(\cdot)$ is according to (4).

We kindly refer interested readers to see work [11] for details on AISLE (3) and to see [12] for details on direct algorithm (5). Further, novel approaches based on nearest neighbours for the approximation of LE are introduced and investigated as they allow us for evaluation of more complicated patterns of learning intensity.

3 PROPOSED METHOD

3.1 Learning State Vector and Learning State Space

Arising from the original concept of Learning Entropy, which is the measure of how unusually large is the learning activity at sample time k , we propose to extend the concept by defining the learning state (vector) and the learning (state) space as follows

$$\mathbf{s} = \mathbf{s}(k) = |\Delta \mathbf{w}(k)| \in \Lambda^{n_w} \in \mathbb{R}_{\geq 0}^{n_w} \quad (6)$$

where \mathbf{s} is the **learning state (vector)**, Λ^{n_w} denotes the n_w -dimensional **learning (state) space** of non-negative real values and $|\Delta \mathbf{w}(k)|$ is the n -sample long vector of absolute values of actual weight updates for the last measured sample point at time k . Generally and according to (6), the learning engine displays high learning activity when $\|\mathbf{s}\|$ is high, the low learning activity is when $\|\mathbf{s}\|$ is low, and because of the proposed clustering, \mathbf{s} can represent a universal basis for estimating LE via neighbourhood analysis in Λ^{n_w} . Thus, we focus on the fundamental definition of the learning state as given in (6), which is our new proposal for LE estimation via

the clustering of unusual learning intensity patterns in this paper.

Further, let us define the contemporary learning history \mathbf{S} of M previous data samples in n_w -dimensional learning state space Λ^{n_w} as follows

$$\mathbf{S} = \mathbf{S}(k) = \begin{bmatrix} \mathbf{s}(k-M) \\ \mathbf{s}(k-M+1) \\ \vdots \\ \mathbf{s}(k-1) \end{bmatrix}, \quad (7)$$

where M is the adjustable length of contemporary learning history, so the contemporary learning history \mathbf{S} is $M \times n_w$ matrix of non-negative real values.

In further subsections, two novel techniques for the estimation of LE, i.e., a K-NN based technique and a multiscale neighbourhood analysis technique, are introduced utilizing \mathbf{s} and \mathbf{S} .

3.2 Approximation of Sample Point LE by K-NN Distances

For learning systems with a low number of neural weights such as linear adaptive filters, LE can be estimated according to the location of the learning state $\mathbf{s}(k)$ directly in the n_w -dimensional learning (state) space Λ^{n_w} according to the contemporary learning history \mathbf{S} . Naturally, the location of the actual learning state $\mathbf{s}(k)$ is the more distinctive the fewer neighbours have the learning state within radius r in the n_w -dimensional learning state space within the contemporary history of learning \mathbf{S} . In principle, we may think of an n_w -dimensional heat map based on \mathbf{S} , where its hot spots would represent unusual patterns of learning intensity (high LE), and its cold spots would represent usual learning patterns (low LE).

To avoid building a full heat map in n-D, we propose to utilize K-NN based efficient algorithms, e.g., KD tree, to find the nearest neighbours of $\mathbf{s}(k)$ in \mathbf{S} . Thus, we can estimate LE via a function whose argument is the sum of distances from K nearest neighbours to the actual learning state $\mathbf{s}(k)$. The principle of the method is sketched in Fig.1, and the most straightforward function for approximation (estimation) of LE is then as follows

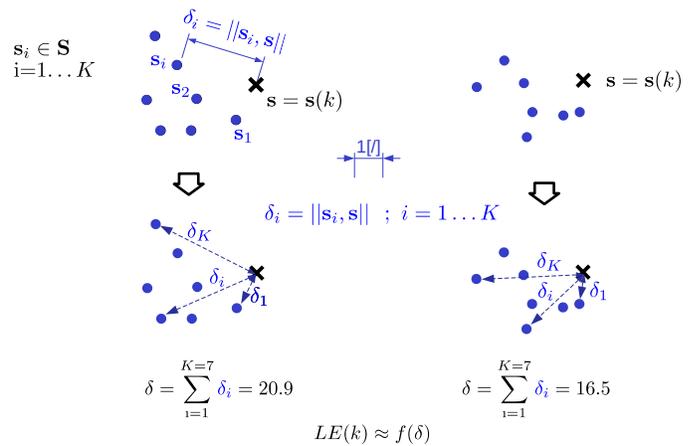


Fig. 1: E_{AK-NN} : The principle of K-NN based approximation of sample point LE (a simplified sketch in 2-D); the actual learning state \mathbf{s} (the cross) on the left side is more unusual (higher LE) than the actual state \mathbf{s} on the right side (lower LE), and $f(\cdot)$ denotes a function in general.

$$E_{AK-NN}(k) = \sum_{i=1}^K \delta_i(k); E_{AK-NN} \in \langle 0, +\infty \rangle, \quad (8)$$

where E_{AK-NN} denotes the approximated sample point LE by summation of K nearest neighbours and $\delta_i(k)$ is the distance to i -th nearest neighbour from state \mathbf{s} within the contemporary learning history \mathbf{S} at sample time k .

3.3 Approximation of Sample Point LE by Multiscale Nearest Neighbour Count

Alternatively to K-NN based technique as introduced in Subsection 3.2, we propose to quantify the unusual location of learning state in learning space via multiscale technique (based on our original research in [16]). This technique explores neighbourhood of the actual learning state \mathbf{s} along with the set of radii in learning space and cumulatively counts the neighbours within the given radii. This naturally resolves the issue of otherwise practically unknown optimal radius ρ , and also, the distribution of neighbouring points is considered (see Fig. 2).

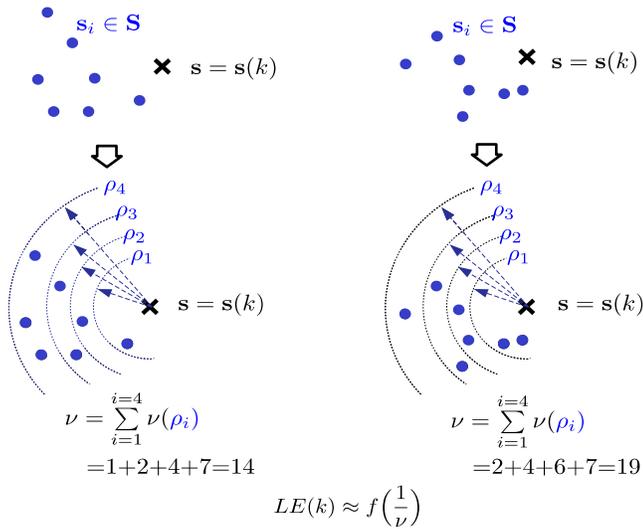


Fig. 2: E_{AMNN} : The principal sketch of multiscale nearest neighbours based approximation of sample point LE (a simplified sketch in 2-D); the actual learning state \mathbf{s} (the cross sign) on the left side is more unusual $f(\cdot)$ (higher LE) than the actual state \mathbf{s} on the right side (lower LE), and $f(\cdot)$ denotes a function in general.

$$\nu = \sum_{i=1}^{i=n_\rho} \nu_i = \sum_{i=1}^{i=n_\rho} \nu(\rho_i) \quad (9)$$

where $\nu_i = \nu(\rho_i)$, i.e. also $\nu = \nu(k)$, is the count of learning states that are neighbours with the actual learning state $\mathbf{s}(k)$ within the actual radius ρ_i in the contemporary learning history \mathbf{S} , and ν is the cumulative count of neighbours over the whole set of the nearest radii at actual sample time k .

Thus we propose a straightforward formula for the approximation of LE for individual point samples as follows

$$\tilde{E}_{AMNN}(k) = \frac{1}{1 + \nu} ; \tilde{E}_{AMNN} \in \langle E_{Amin}, 1 \rangle \quad (10)$$

where \tilde{E}_{AMNN} denotes the approximated sample point LE by multiscale nearest neighbours, and the cumulative neighbour count ν is sketched in Fig.2 and given in (9). The lowest value of estimated LE (10) depends on user-defined configuration parameters as follows

$$E_{Amin} = \frac{1}{1 + M \cdot n_\rho} > 0, \quad (11)$$

where M is length of contemporary learning history (7) and n_ρ is the number of user defined nearest radii (e.g. $n_\rho = 4$ in Fig.2). Further, we can apply min-max normalization as follows

$$E_{AMNN}(k) = \frac{\tilde{E}_{AMNN}(k) - E_{Amin}}{1 - E_{Amin}}, \quad (12)$$

so we obtain the final standardized estimated sample point LE via multiscale nearest neighbours as follows

$$E_{AMNN}(k) = \frac{M \cdot n_\rho - \nu}{M \cdot n_\rho \cdot (1 + \nu)} ; E_{AMNN}(k) \in \langle 0, 1 \rangle \quad (13)$$

and these novel sample point LE estimations, i.e., E_{AK-NN} in (8) and E_{AMNN} in (13), are the funding contributions of this paper.

Furthermore, due to the clustering principle, the LE estimation becomes more suitable for more complicated predictors and tracking learning systems, such as neural networks and potentially deep ones, and this is introduced in the next subsection.

3.4 LE with Dimensionality Reduction for Deep Learning

The above-introduced clustering techniques of LE estimation are mainly intended and investigated for adaptive linear filters and for nonlinear polynomial filters, whose mathematical structure is rather simple and where the number of adaptive parameters is low.

In this subsection, we propose to apply dimensionality reduction preprocessing via classical methods such as PCA or SVD for learning systems where the number of adaptive parameters is high. Thus, the LE can become more suitable for research with neural networks and possibly also with deep neural networks.

The idea of this learning state-space dimensionality reduction is straightforward as follows. At every sample time during detection, the count of n_w columns of learning history $\mathbf{S}(k)$ in (7) can be reduced to a smaller number of columns via some dimensionality reduction technique. Let's denote

$$n_w > n_{PCA} \text{ or } n_w > n_{SVD}, \quad (14)$$

where n_w is the total number of learning weights, i.e., it is the dimension of original learning space $\Lambda \in \mathcal{R}_{\geq 0}^{n_w}$, and n_{PCA} or n_{SVD} are new dimensions of the learning state-space after dimensionality reduction via PCA or SVD, respectively. According to the applied reduction technique, the actual learning state $\mathbf{s}(k)$ is then also reduced to the corresponding dimension. Then, the clustering estimation of LE, as proposed in the above subsections, can be applied in the same way.

The main objective of this paper is the clustering concept for practical estimation of LE; the proper investigation of dimensionality reduction techniques that would focus sparsity and other aspects of learning systems and learning states exceeds the scope of this paper. Furthermore, other dimensionality reduction techniques can be investigated, including auto-encoders in the future.

Experiments with adaptive filters and LE estimation with PCA and SVD dimensionality are also included in the following experiments in the next section.

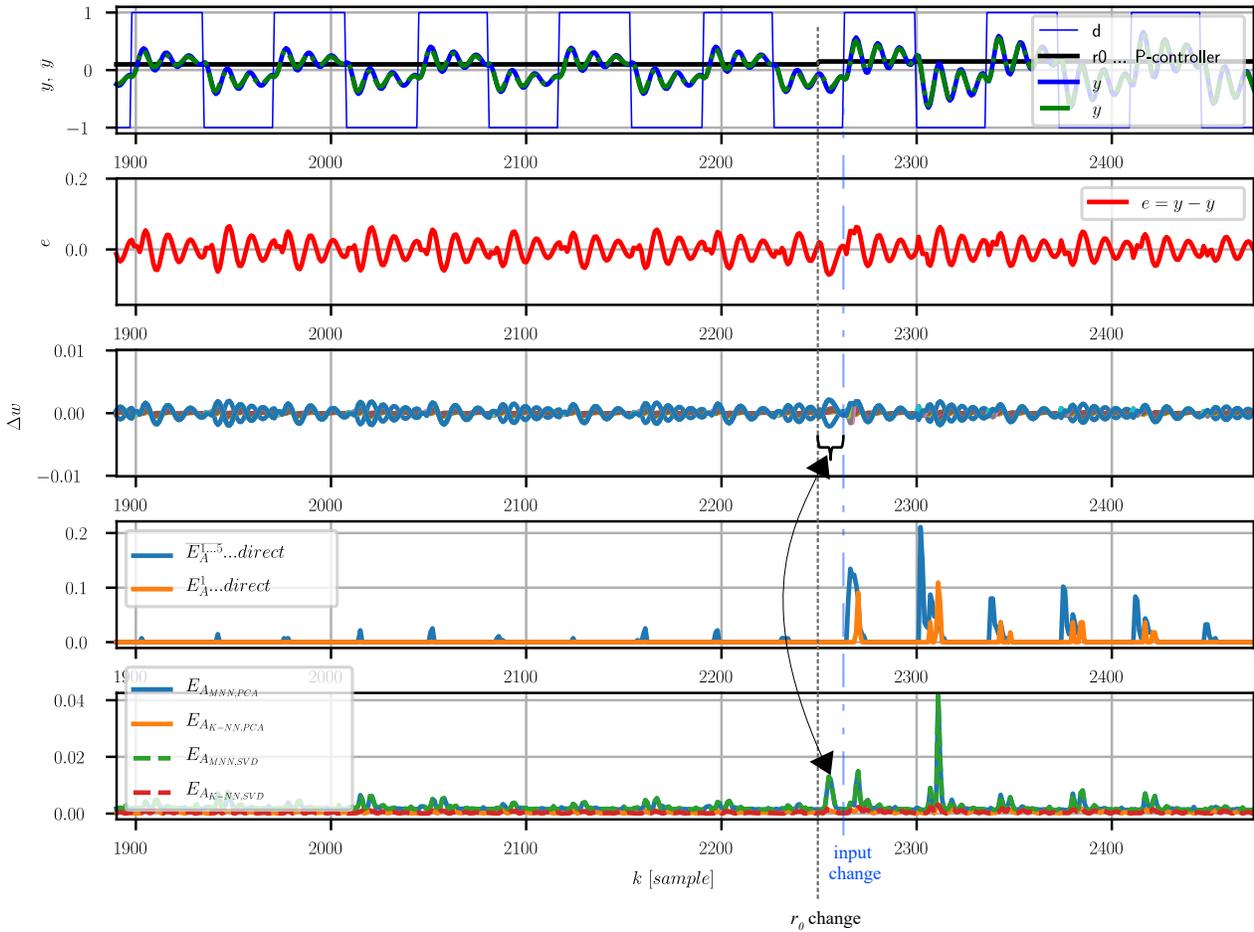


Fig. 3: Anomaly detection in a closed-loop dynamical system (15)(16) via LE, where the gain r_0 suddenly changes at $k = 2250$ (top axes); this sudden perturbation is not immediately apparent in plant output (top axes), nor in the actual error of the adaptive filter (second axes from top), nor it is so apparent via adaptive weight increments (middle axes); the original LE algorithms (3)(5)(second from bottom axes) detect the anomaly later than the new clustering based techniques with dimensionality reduction (bottom axes).

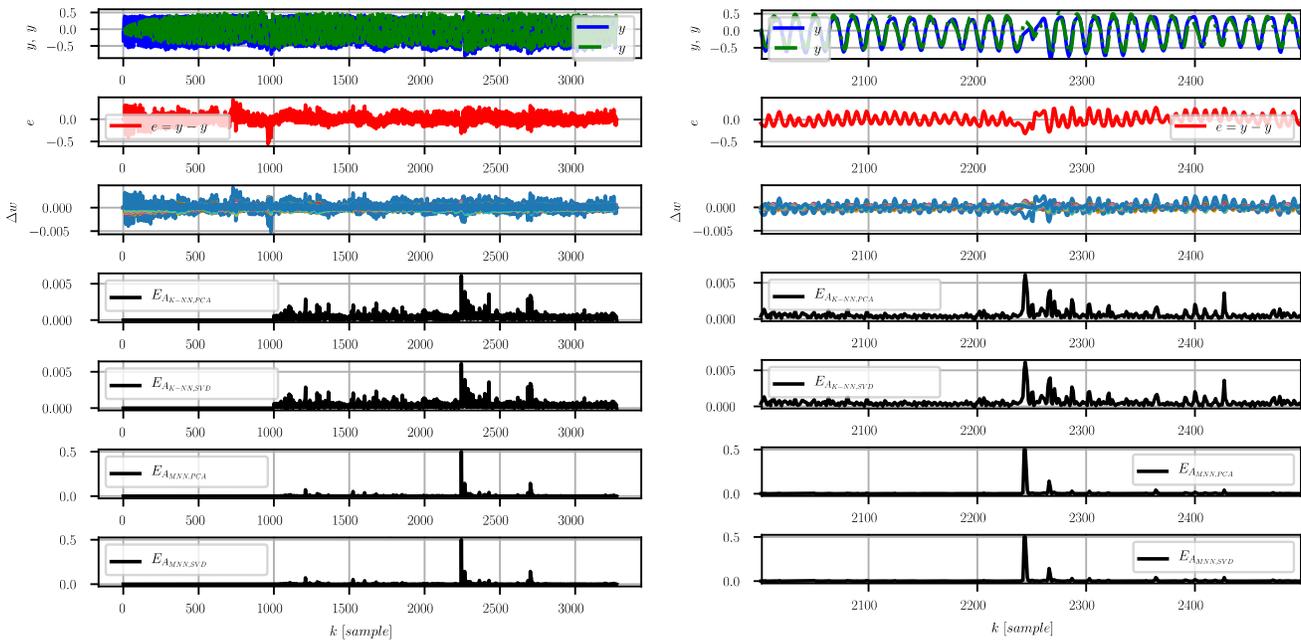


Fig. 4: Respiration motion time series (top row plots, blue...measured, green...adaptive filter (linear learning predictor by online least mean squares, $h = 1$, sampling 6 s.p.s.) with a significant anomaly (details in left column plots); the normalized multiscale neighbour technique E_{AMNN} (12) is more robust and detects clearly more significant anomalies for this time series; here, the K-NN based technique E_{AK-NN} (8) is more sensitive and also detects less significant anomalies for this data (and this predictor, its learning, and the particular setups). See Fig.5 for details of the anomaly around $k \approx 2690$.

4 EXPERIMENTAL ANALYSIS

4.1 Anomaly Detection in a Dynamical System

First, the performance of the proposed methods was examined by the anomaly detection in a single-input single-output linear oscillating dynamical system that was defined as follows

$$\ddot{y}(t) + 2 \cdot \eta \cdot \Omega_0 \cdot \dot{y}(t) + \Omega_0^2 \cdot y(t) = 2 \cdot \Omega_0^2 \cdot u(t), \quad (15)$$

$$u(t) = r_0(t) \cdot (d(t) - y(t)), \quad (16)$$

where d was the periodic step input (top axes in Fig.3), y was the plant output, and r_0 was the proportional gain of a P-controller. The closed loop (15)(16) was simulated with an ODE solver, i.e., as a continuous-time dynamical system. The only measured variables were input $d(t)$ and output $y(t)$, and these were sampled with a constant sampling interval. The anomaly was caused by a sudden change of a proportional gain r_0 , i.e., r_0 suddenly changed at $k = 2250$ (see the top axes in Fig.3). The real-time learning predictor was a linear (FIR) adaptive filter whose input vector \mathbf{x} in (1) involved step delayed values of d and step delayed outputs y . Fig. 3 shows already the performance of the converged filter; however, its output error does not converge to zero (second top axes in Fig. 3), perhaps due to imperfect filter configuration and learning setups. As the learning rule, the normalized least squares rule [14] was applied, i.e., the gradient descent with normalized learning rate. Then, the bottom axes in Fig. 3 show that the new method from subsection 3.3, and even with dimensionality reduction via PCA or by SVD, detects the anomaly in the dynamical behaviour earlier than the non-clustering techniques (3) and (5).

4.2 Time Series and Anomaly Detection

In this experimental subsection, the two new clustering-based methods from subsections 3.2 and 3.3 were mutually compared on novelty detection in nonstationary biomedical data, i.e., in the lung-tumour motion time series during patient respiration¹. Again, the weights of linear (FIR) adaptive filter with NLMS incremental learning were used for anomaly detection via LE. The results are shown in Fig. 4. In this case, the normalized multiscale neighbour technique E_{AMNN} (12) is more robust and detects clearly only the most significant anomalies (at $k \approx 2250, 2700$ for this time series, while the K-NN based technique E_{AK-NN} (8) is more sensitive and detects also less significant anomalies in this data (and for this predictor, its learning, and the particular setups). The detail of the detected anomaly at $k \approx 2690$ is shown in Fig.5.

5 CONCLUSIONS

The estimation of Learning Entropy for anomaly detection was proposed with the dimensionality reduction and clustering techniques in learning state space. The combination of the approaches appeared suitable for novelty detection with adaptive filters, and it can outperform the original non-clustering algorithms of LE. Further, the new techniques open new frontiers of LE for more complicated learning systems, such as neural networks and deep learning.

Acknowledgements

The work was supported by the Ministry of Education, Youth and Sports of the Czech Republic under OP RDE grant number CZ.02.1.01/0.0/0.0/16_019/0000753 "Research Centre for Low-Carbon Energy Technologies, and by the Japanese project Smart aging research center grant and JSPS Kakenhi #17H04117 and #18K19892.

¹ Anonymous respiration data courtesy of Hokkaido University Hospital

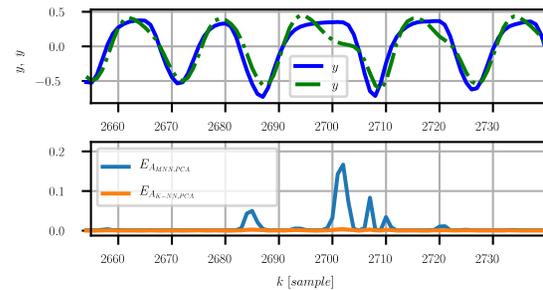


Fig. 5: Detail for the second significant anomaly from Fig. 4; notice that E_{AK-NN} is not a standardized measure, so its magnitude is much smaller than the one of E_{AMNN} .

REFERENCES

- [1] M. Markou and S. Singh, "Novelty detection: a review—part 1: statistical approaches," *Signal Process.*, vol. 83, no. 12, pp. 2481–2497, Dec. 2003, doi: 10.1016/j.sigpro.2003.07.018.
- [2] J. S. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy," *Am. J. Physiol. Heart Circ. Physiol.*, vol. 278, no. 6, pp. H2039–2049, Jun. 2000.
- [3] S. M. Pincus, "Approximate entropy as a measure of system complexity," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 88, no. 6, pp. 2297–2301, Mar. 1991.
- [4] M. Markou and S. Singh, "Novelty detection: a review—part 2: neural network based approaches," *Signal Process.*, vol. 83, no. 12, pp. 2499–2521, Dec. 2003, doi: 10.1016/j.sigpro.2003.07.019.
- [5] C. Alippi, S. Ntalampiras, and M. Roveri, "A Cognitive Fault Diagnosis System for Distributed Sensor Networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 8, pp. 1213–1226, Aug. 2013, doi: 10.1109/TNNLS.2013.2253491.
- [6] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, Jul. 2009, doi: 10.1145/1541880.1541882.
- [7] C. Alippi and M. Roveri, "Just-in-Time Adaptive Classifiers #x2014;Part I: Detecting Nonstationary Changes," *IEEE Trans. Neural Netw.*, vol. 19, no. 7, pp. 1145–1153, Jul. 2008, doi: 10.1109/TNN.2008.2000082.
- [8] R. Elwell and R. Polikar, "Incremental Learning of Concept Drift in Nonstationary Environments," *IEEE Trans. Neural Netw.*, vol. 22, no. 10, pp. 1517–1531, Oct. 2011, doi: 10.1109/TNN.2011.2160459.
- [9] I. Bukovsky, W. Kinsner, and J. Bila, "Multiscale analysis approach for novelty detection in adaptation plot," in *Sensor Signal Processing for Defence (SSPD 2012)*, Sep. 2012, pp. 1–6, doi: 10.1049/ic.2012.0114.
- [10] I. Bukovsky, C. Oswald, M. Cejnek, and P. M. Benes, "Learning entropy for novelty detection a cognitive approach for adaptive filters," in *Sensor Signal Processing for Defence (SSPD), 2014*, Sep. 2014, pp. 1–5, doi: 10.1109/SSPD.2014.6943329.
- [11] I. Bukovsky, "Learning Entropy: Multiscale Measure for Incremental Learning," *Entropy*, vol. 15, no. 10, pp. 4159–4187, Sep. 2013, doi: 10.3390/e15104159.
- [12] I. Bukovsky, W. Kinsner, and N. Homma, "Learning Entropy as a Learning-Based Information Concept," *Entropy*, vol. 21, no. 2, p. 166, Feb. 2019, doi: 10.3390/e21020166.
- [13] J. Vrba and J. Mareš, "Introduction to Extreme Seeking Entropy," *Entropy*, vol. 22, no. 1, p. 93, Jan. 2020, doi: 10.3390/e22010093.
- [14] D. P. Mandic, S. Kanna, and A. G. Constantinides, "On the Intrinsic Relationship Between the Least Mean Square and Kalman Filters [Lecture Notes]," *IEEE Signal Process. Mag.*, vol. 32, no. 6, pp. 117–122, Nov. 2015, doi: 10.1109/MSP.2015.2461733.
- [15] X. Hong and Y. Gong, "A constrained recursive least squares algorithm for adaptive combination of multiple models," in *2015 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2015, pp. 1–6, doi: 10.1109/IJCNN.2015.7280298.
- [16] I. Bukovsky, W. Kinsner, V. Maly, and K. Krehlik, "Multiscale Analysis of False Neighbors for state space reconstruction of complicated systems," in *Merging Fields of Computational Intelligence and Sensor Technology (CompSens), 2011 IEEE Workshop On*, Apr. 2011, pp. 65–72, doi: 10.1109/MFCIST.2011.5949517.

Classifying LPI signals with transfer learning on CNN architectures

Bunlong Lay, Alexander Charlish
 Dept. Sensor Data and Information Fusion
 Fraunhofer FKIE, Wachtberg, Germany
 Email: bunlong.lay@fkie.fraunhofer.de

Abstract—Due to the increased deployment of low probability of intercept radar systems, recognition and classification of low probability of intercept signals has developed an increased importance for electronic warfare systems. Recent results showed that combining time-frequency transformations such as Choi-Williams distribution with convolutional neural networks yield high accuracy. Since training convolutional neural networks is a time consuming task, we propose to use transfer learning on pre-trained convolutional neural network architectures. Furthermore, we compare these retrained neural network to the neural network trained with randomly initialized weights. We will demonstrate that the required training time for the transfer learning method is significantly shorter. Moreover, classifying time-frequency images based on Choi-Williams distribution achieves for both weight initialization methods an accuracy of over 99%.

I. INTRODUCTION

One of the functions of an electronic support system is to analyze surrounding radar emitters which may pose a threat. To avoid detection, some radar systems have been designed to use low probability of intercept (LPI) waveforms [1]. In order to separate radar emitters, electronic support performs deinterleaving, which tries to sort unknown radar pulses. This method can be improved, if apriori knowledge about the received pulses is available. Thus classification of LPI signals is essential to separate threats and to perform signal deinterleaving. Due to the nature of LPI signals, finding a highly reliable efficient solution to the classification problem of LPI waveforms is non-trivial. Approaching this problem is the focus of this paper.

The key to creating efficient LPI signals is to use pulse-compression based on intrapulse modulation. Pulse compression simultaneously yields a reduction of the radar's peak effective radiated power and a large time-bandwidth product [1]. Different pulse compression techniques induce different LPI waveforms. In this paper twelve LPI waveforms are considered (see subsection II-C for details).

In the literature several LPI waveform recognition techniques are analyzed [1]. In particular, the Choi-Williams distribution (CWD) is widely used [2]–[6]. Hence, utilizing CWD to convert the raw data into a 2D time-frequency image is appropriate. In order to classify the images the idea of convolutional neural networks (CNNs) is applied. Recent results in visual recognition competitions demonstrated the efficiency of CNNs. The well-known ImageNet Large Scale Visual Recognition Competition proves the quick develop-

ment and great success of CNNs [8]. Due to computational limitations of computers it is reasonable to transfer pre-trained CNNs and adapt them for a different classification task [15]. This method is known as transfer learning. It works for CNNs pre-trained on ImageNet [9], which is also used in this paper.

A model-based transfer learning was already analyzed in [4]. Model-based transfer learning does not retrain the pre-trained weights. In [4] it was shown that ResNet-152 with support vector machine as a classifier reaches an overall recognition rate of 97.8%. In [2] and [5] the LPI signals are classified on a CNN, which is trained in both articles without transfer learning. In [2] the CNN achieves an overall performance of 93.58%, when the SNR is -6 dB. In [5] classification is optimized with Tree-based Pipeline Optimization Tool and an overall recognition rate of 94.42% is reached, when the SNR is -4 dB. Note, that the parameter space for sample generations in all mentioned articles are different from each other, preventing a direct comparison.

In this paper, we initialize the weights from the pre-trained CNN architecture on ImageNet. We then retrain the entire CNN, which differs from the model-based transfer learning approach in [4]. Retraining the entire CNN architecture is reasonable, since the images from ImageNet and the time-frequency images clearly differ from each other and a huge amount of time-frequency images are generated. For comparison we also train the CNNs with randomly initialized weights instead of using the pre-trained weights from ImageNet. Sophisticated CNN architectures have been carefully studied and are easily accessible, thus we also investigate several CNN architectures, such as VGG16, ResNet50, InceptionResNetV2, DenseNet and MobileNetV2. We use two fully connected dense layers as a classifier in addition to the convolutional base. As expected, the transfer learning method is less time-consuming than the method with randomly initialized weights. Moreover, our classification method reaches an averaged accuracy of over 99%, when the SNR ranges from -6 dB to 2 dB, for all considered CNN architectures and for both weight initialization methods.

This paper is organized as follows. In Section II an overview of the simulation model is provided. This includes a detailed explanation of sample creation and preprocessing methods. The idea of CNNs with transfer learning and implementation details for training the CNNs is discussed in Section III. Section IV presents the simulation results for

the time-frequency images based on CWD. In addition, a conclusion and an outlook are drawn in the Section V.

II. LPI TIME-FREQUENCY IMAGES GENERATION

A. Framework

The received modulated LPI waveform is of the form

$$s(t) = x(t) + n(t), \tag{1}$$

with $x(t)$ being some modulated waveform and $n(t)$ being additive white Gaussian noise. The preprocessing commences with applying the CWD to transform the signal $s(t)$ to a 2D time-frequency image. In addition, the 2D time-frequency images are resized with area interpolation to match the input size of the neural network.

We consider nine SNR levels for each of the twelve LPI waveform classes, ranging from -6 dB to 2 dB. For each SNR level 700 signals per LPI class were randomly generated. We train on 70% percent of the generated data, leaving 15% for validation and 15% for testing. After preprocessing, the architecture of a convolutional base is loaded. In this paper we propose two weight initialization methods. The classification results of the two different methods will be compared in Section IV. One method uses pre-trained ImageNet weights, whereas the other method initializes random weights (see Section III for more details). On top of the convolutional base a dropout layer and two dense layers are used (see Fig. 1).

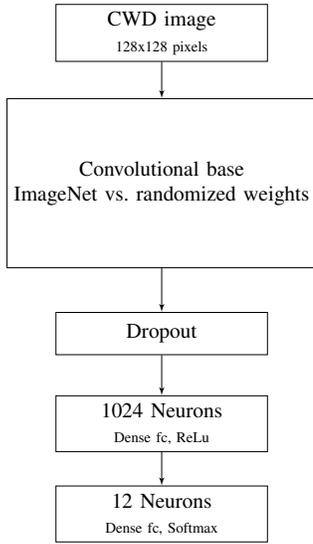


Fig. 1. This Figure displays the structure of the CNN.

B. LPI radar waveforms

In this subsection we briefly discuss the mathematical model of LPI radar signals. The transmitted signal can be written as

$$x(t) = Ae^{j\theta(t)}, \tag{2}$$

$\theta(t)$ is the instantaneous phase modulation and A is the amplitude [1]. In this paper we fix $A = 1$. In regard of equation (1) and by discretizing, we obtain

$$s(kT) = Ae^{j\theta(kT)} + n(kT), \tag{3}$$

where $k \in \mathbb{N}$ and T is the sampling interval. Note, that the instantaneous phase θ can be written as the sum of the instantaneous frequency f_k and the phase offset ϕ_k :

$$\theta(kT) = 2\pi f_k kT + \phi_k. \tag{4}$$

In this paper there are two different types of LPI waveforms, frequency modulated and phase modulated waveforms. For the frequency modulated signals f_k is not constant and ϕ_k is constant. Vice versa, for the phase modulated signals ϕ_k is not constant and f_k is constant.

C. Sample Creation

In order to train the CNN properly it is important to generate randomized data. Table I shows the twelve LPI waveforms which are considered for classification.

The sampling frequency for all waveforms is set to $f_s = 8000$ Hz. The carrier frequency is denoted by f_c . With m, c_{pp} we denote the number of code phases and the number of cycles per phase respectively. For the polytime codes T_1, \dots, T_4 we define k as the number of stepped frequency segments. In order to parametrize the deterministic part of the signal model given above f_c, m, c_{pp}, k are uniformly sampled from real-valued sets. The binary phase shifting keys (BPSKs) are based on Barker codes. The length of the Barker code B is uniformly chosen at random from $\{7, 11, 13\}$. The Costas arrays C are uniformly chosen from the set $C_A = \{\{3, 2, 6, 4, 5, 1\}, \{5, 4, 6, 2, 3, 1\}, \{2, 4, 8, 5, 10, 9, 7, 3, 6, 1\}\}$ at random, where the unit of C is given in kHz. Furthermore t_p denotes the duration in milliseconds of a sub-period for Costas waveforms. For the triangle linear frequency modulated (TLFM) waveforms we denote the bandwidth by Δf . The waveforms are generated with Matlab (R2017b). A detailed description of the waveforms can be found in [1].

D. CWD transformation

After waveforms $s(t)$ are generated we proceed in Matlab to transform $s(t)$ into a 2D time-frequency image by using the CWD transformation as it is defined in [1]. The CWD is a triple integral of the Cohen's class over the time and frequency space.

$$C_x(t, \omega, \phi) = \frac{1}{2\pi} \int_{\mathbb{R}^3} e^{i(\xi\mu - \tau\omega - \xi t)} \phi(\xi, \tau) A_x(\mu, \tau) d\mu d\tau d\xi, \tag{5}$$

where

$$A_x(\mu, \tau) = x(\mu + \tau/2)x^*(\mu - \tau/2) \tag{6}$$

with kernel function

$$\phi(\xi, \tau) = e^{-\xi^2 \tau^2 / \sigma}, \tag{7}$$

where $\sigma > 0$ denotes a scaling factor. In this paper we set $\sigma = 1$. The resulting image is resized to 128×128 pixels by using area interpolation in order to suit the input size of the

TABLE I
PARAMETERS FOR SAMPLE GENERATIONS

#	LPI signal	Parameter	Uniformly sampled from
0	BPSK	B	$\{7, 11, 13\}$
		f_c	$[1/5f_s, 1/7f_s]$
		c_{pp}	$\{1, 2, 3\}$
1	Costas	C	C_A
		t_p	$[1, 4]$
2	TLFM	f_c	$[1/16f_s, 1/6f_s]$
		Δf	$[1/20f_s, 1/16f_s]$
		c_{pp}	$\{3, 4, 5\}$
		m	$\{4, 5, 6, 7, 8\}$
3	Frank	f_c	$[1/7f_s, 1/5f_s]$
		c_{pp}	$\{3, 4, 5\}$
		m	$\{4, 5, 6, 7, 8\}$
4, 5	P1,P2	f_c	$[1/7f_s, 1/5f_s]$
		c_{pp}	$\{3, 4, 5\}$
		m	$\{4, 5, 6, 7, 8\}$
6, 7	P3, P4	f_c	$[1/7f_s, 1/5f_s]$
		c_{pp}	$\{3, 4, 5\}$
		m	$\{16\}$
8 – 11	T1-T4	f_c	$[1/7f_s, 1/5f_s]$
		k	$\{4, 5, 6\}$
		m	$\{2\}$

neural network (see Fig. 1). Samples of the resulting images of the twelve LPI waveforms are depicted in Fig. 2.

III. CNN AND TRANSFER LEARNING

CNNs are composed of a convolutional base and a classification part, which consists usually of fully connected dense layers followed by a softmax layer. The purpose of the convolutional base is to extract complex features from the input image. This is done by combinations of convolution, padding and pooling.

However, training CNNs on a large set of images, e.g. ImageNet, requires huge resources, in terms of GPUs and time. For example in [10] it is stated that training ImageNet on four NVIDIA Titan Black GPUs lasted two to three weeks for some CNN architecture.

To avoid time consuming training, this paper proposes the popular method of transfer learning. Since the images from ImageNet differ from the images here (Fig. 2) and since we generated a large data set of images, it is reasonable to retrain the pre-trained weights from ImageNet, but using the pre-trained weights as a starting point to save time.

In this paper we investigate different convolutional pre-trained bases, which are:

VGG16 [10], ResNet50 [11], InceptionResNetV2 [12], DenseNet [13] and MobileNetV2 [14].

The entire CNN structure is implemented in Python 3.6 with Keras. We stored the complete training data set and validation

set into the memory of the GPU to accelerate the process of training. We performed the training on a NVIDIA GeForce GTX 1080 Ti with 11 Gb memory.

TABLE II
PARAMETERS FOR TRAINING THE CNNs

Name	Parameter	Value
SGD	Learning rate	0.001
	Decay	10^{-6}
	Momentum	0.9
Early Stop	Monitoring	Validation loss
	Min. change	0.00001
	Patience	25 epochs
Other	Batch size	32
	Epochs	150
	Dropout	0.5
	Validation data size	15% of shuffled data
	Test data size	15% of shuffled data

Moreover, the initialization with random weights is based on the Xavier uniform initializer (see [16]). We remark that the hyperparameter space was explored (based on validation data) until reasonable parameters were found. The parameters from Table II worked sufficiently well for all considered CNN architectures. The results in Section IV are based on this implementation.

IV. SIMULATION RESULTS

We start this Section by first discussing the transfer learning results. Then we discuss the difference between the transfer learning approach to training the CNNs with the Xavier uniform initializer.

A. Classification with transfer learning

The result of the classification when using different CNN architectures and transfer learning can be found in Table III. T_A^{tr} , T_L^{tr} , t^{tr} and E^{tr} denote the accuracy of the test data, loss on the test data, duration in hours for the entire training session and best epoch with respect to the loss function on the validation set.

All considered CNN architectures have similar accuracy and loss values on the test data set, with exception of InceptionResNetV2 having a slightly higher loss. More precisely, the test loss function monotonically decreases with each epoch for all considered CNNs, except for InceptionResNetV2, until the values are in the order of 10^{-3} . Then the test loss function remains constant with small fluctuation (see Fig. 3). Since InceptionResNetV2 has the highest loss, we will disregard it from further classifications analysis. Fig. 4 depicts the t-SNE plot for VGG16. It can be seen that all classes can be well separated. Similar results are obtained for the other considered architectures. The confusion matrices of all architectures show that most missclassifications occur between Frank, P1, P3 and P4 waveforms (see Table IV), this is possibly explainable due to the similar shape of these

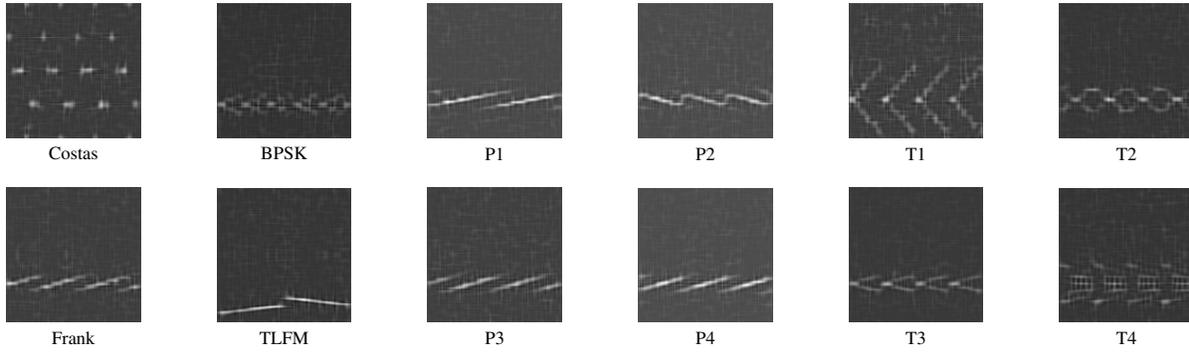


Fig. 2. LPI waveforms transformed by Choi-Williams distribution

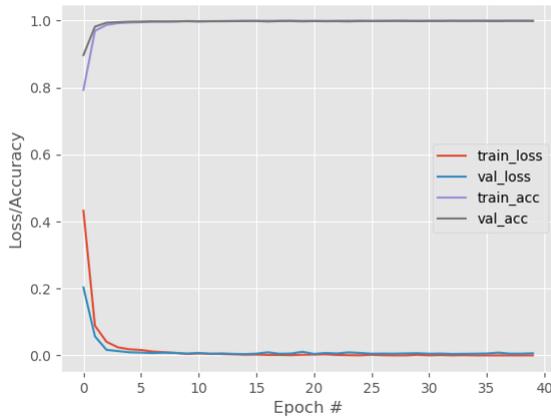


Fig. 3. This Figure displays the accuracy and loss values on the training and validation data set for VGG16 with the transfer learning method. The accuracy/loss plots for ResNet50, DenseNet and MobileNetV2 are very similar to this Figure.

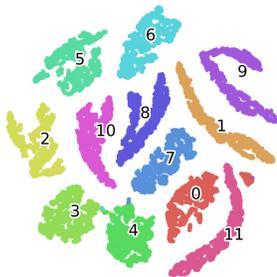


Fig. 4. This Figure displays the VGG16 t-SNE plot in the 13th epoch evaluated on the test set. The enumeration is according to Table I, e.g. 0 correspond to BPSK.

time-frequency images as can be seen in Fig. 2.

The classification performance of all investigated CNN architectures, except for InceptionResNetV2, are similar to each other with an accuracy above 99% and loss value around 10^{-3} . DenseNet has the best performance values.

TABLE III
RESULTS FOR TRAINING WITH TRANSFER LEARNING

Architecture	T_A^{tr}	$T_L^{tr}/10^{-3}$	t^{tr}	E^{tr}
VGG16	99.76	8.58	2.17	13
ResNet50	99.79	8.22	4.58	31
Inception-ResNetV2	99.70	16.45	11.50	31
DenseNet	99.89	4.64	9.40	22
MobileNetV2	99.83	9.39	5.61	46

The training time required is lowest for VGG16, thus it is reasonable to work with VGG16 for future work.

TABLE IV
NUMBER OF MISCLASSIFIED IMAGES

Architecture	Confusion between Frank, P1, P3, P4	Other confusions	Total
VGG16	23	4	27
ResNet50	17	6	23
Inception ResNetV2	28	5	33
DenseNet	11	2	13
MobileNetV2	17	2	19

B. Comparison to classification without transfer learning

The results of the comparison can be found in Table V, where we use following notations. We set $\Delta T_A = T_A^{tr} - T_A^0$, where the superscript in T_A^0 denotes the method without transfer learning, i.e. with randomly initialized weights. In the same fashion we define $\Delta T_L, \Delta t, \Delta E$ to be the difference of the loss on the test data, training time and best epoch from the transfer learning method to learning the CNNs with randomly initialized weights.

The performance, i.e. the loss values and accuracy, on the classification task without transfer learning are very similar to the performance with transfer learning. This means that the transfer learning approach does not yield significantly better CNN models for LPI classification. The loss on the

test data is not significantly different for both methods and for all considered CNN architectures (see Table V), except for InceptionResNetV2. For InceptionResNetV2 the method with randomly initialized weights yields much better results than the transfer learning method. Furthermore, the confusion matrices in both methods do not differ significantly for all considered CNN architectures. A difference of the two proposed methods can be examined from the time difference column, given by Δt in Table V. Since clearly $\Delta t < 0$ for all considered CNN architectures, the transfer learning method has a big advantage when time is considered to be a resource. Training the CNNs on randomly initialized weights vary the results. To show empirically that transfer learning has a time benefit, we run 65 training sessions on VGG16 with randomly initialized weights. In all test runs the method with transfer learning required less training time (examine Table VI for details). The performance is on average better by a very small margin, which can be seen by comparing Table VI to the first row of Table III. We empirically show that using pre-trained weights significantly shortens the time needed to train the classifier compared to using the Xavier uniform initializer.

TABLE V
DIFFERENCE FROM THE TRANSFER LEARNING METHOD TO LEARNING
THE CNNs WITH RANDOMLY INITIALIZED WEIGHTS

Architecture	ΔT_A	$\Delta T_L/10^{-3}$	Δt	ΔE
VGG16	$\approx -10^{-4}$	1.74	-1.13	-22
ResNet50	$\approx 10^{-5}$	-2.6	-1.83	-25
Inception-ResNetV2	$\approx -10^{-3}$	11.99	-8.03	-31
DenseNet	$\approx 10^{-4}$	-0.78	-15.4	-5
MobileNetV2	$\approx -10^{-4}$	3.89	-1.11	-8

TABLE VI
65 TRAINING SESSIONS ON VGG16 WITH RANDOMLY INITIALIZED
WEIGHTS

	Mean	Variance	Min	Max
t^0	3.36	0.48	2.23	5.37
$T_L^0/10^{-3}$	8.5	$3.91 \cdot 10^{-3}$	4.77	15.0
T_A^0	99.78	$2.96 \cdot 10^{-7}$	99.62	99.88

V. CONCLUSION

In this paper we investigated the classification task of LPI signals with several CNN architectures and two different weight initialization methods. We demonstrated that the transfer learning method yields much faster convergence compared to the method when the CNNs are trained with randomly initialized weights.

In terms of performance both methods do not differ significantly from each other. The CNNs VGG16, ResNet50, DenseNet and MobileNetV2 performed equally well on the classification task. In terms of required training time VGG16

is the fastest network to train. Hence our future research on classifying LPI waveforms with CNNs will be focused on VGG16 with transfer learning.

In electronic warfare processing large amount of data should be analyzed in real time. In our proposed classification method the CWD transformation is a computationally expensive operations which may hinder the use of the method in real-time. Thus an investigation into computational optimization of the CWD transform or its replacement without a loss of performance is necessary.

REFERENCES

- [1] Phillip E. Pace, "Detecting and classifying low probability of intercept radar," *Artech House*, Second Edition, 2009.
- [2] Seung-Hyun Kong, Minjun Kim, Linh Manh Hoang, Eunhui Kim "Automatic LPI Radar Waveform Recognition using CNN," *IEEE Access*, vol. 6, pp. 4207-4219, 2018.
- [3] Zhiyu Qu, Xiaojie Mao, Zhian Denjg, "Radar Signal Intra-pulse Modulation Recognition Based on Convolutional Neural Network," *IEEE Access*, vol. 6, pp. 43874-43884, 2018.
- [4] Qiang Guo, Xin Yu, Guoqing Ruan, "LPI Radar Waveform Recognition Based on Deep Convolutional Neural Network Transfer Learning," *Symmetry 2019*, vol. 11, page 540, 2019.
- [5] Jian Wan, Xin Yu, Qiang Guo, "LPI Radar Waveform Recognition Based on CNN and TPOT," *Symmetry 2019*, vol. 11, page 725, 2019.
- [6] Ming Zhang, Ming Diao, Lipeng Gao, Lutaio Liu, "Neural Networks for Radar Waveform Recognition," *Mechatronic Systems and Control (formerly Control and Intelligent Systems)*, vol. 46, 2018.
- [7] Rikiya Yamashita, Mizuho Nishio, Richard King Gian Do, Kaori Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, 2018
- [8] Md. Zahangir Alom, Tarek M. Taha, Christopher Yakopcic, Stefan Westberg, Mahmudul Hasan, Brian C. Van Esesn, Abdul A. S. Awwal, "The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches," *ArXiv e-prints, ArXiv:1803.01164*, 2018 [Online], Available: <https://arxiv.org/abs/1803.01164>
- [9] Minyoung Huh, Pulkit Agrawal, Alexei A. Efros, "What makes ImageNet good for transfer learning?," *ArXiv e-prints, ArXiv:1608.08614*, 2016 [Online], Available: <https://arxiv.org/abs/1608.08614>
- [10] Karen Simonyan, Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *ArXiv e-prints, ArXiv:1409.1556*, 2015 [Online], Available: <https://arxiv.org/abs/1409.1556>
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.
- [12] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, Alex Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 4278-4284, 2017.
- [13] Gao Huangm Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger, "Densely Connected Convolutional Networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2261-2269, 2017.
- [14] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4510-4520, 2018.
- [15] Sebastian Thrun, Lorien Pratt, "Learning to learn," *Kluwer Academic Publishers*, 1998.
- [16] Xavier Glorot, Yoshua Bengio, "Understanding the difficulty of training deep feedforward neural networks," *Journals of Machine Learning Research - Proceedings Track*, vol. 9, pp. 249-256, 2010.

Approximate LASSO Model Predictive Control for Resource Constrained Systems

Yun Wu, João F. C. Mota, Andrew M. Wallace
The School of Engineering and Physical Sciences
Heriot-Watt University
 Edinburgh, UK
 {y.wu, j.mota, a.m.wallace}@hw.ac.uk

Abstract—LASSO MPC is a popular method for solving optimal control problems within a receding horizon. It is, however, challenging to deploy LASSO MPC on resource constrained systems, such as embedded platforms, due to the intensive memory usage and computational cost as the horizon length is extended. By exploiting a reduced precision, approximation technique applied to Proximal Gradient Descent (PGD), we demonstrate an implementation on a resource constrained, reconfigurable device, such as a Field Programmable Gate Array (FPGA). Our experiments show equivalent performance to a high-precision optimisation solver, but with significant improvements to both logic cost and memory bandwidth, up to 60% and 80% reduction respectively, with up to 70% power savings.

Index Terms—MPC, LASSO, Proximal Gradient Descent, Approximate Computing, FPGA

I. INTRODUCTION

Model Predictive Control (MPC) is a popular technique to solve optimal control problems in discrete time. Its robustness, stability, and theoretical guarantees have made it widely adopted by industry [1]. Given a dynamic model of a system and an estimate of its current state, MPC computes the next states and necessary control inputs by minimizing a cost function that balances the achievement of a desired state in a predefined time-horizon with the energy required. As energy is often expressed in a quadratic form, MPC typically entails the application of (nonzero) input signals to the system at all time steps.

To reduce the number of times that inputs are applied, the work in [2] proposed *LASSO MPC* which, inspired by results on sparsity, regularizes the energy term with an ℓ_1 -norm penalty. This leads to sparse input signals. Despite many desirable features, (LASSO) MPC requires an iterative procedure to solve an optimization problem at each time step. This not only makes execution in real-time challenging (as the number of required iterations is unknown a priori), but also demands significant memory and computation, which can be limiting on resource-constrained systems, such as embedded hardware.

In such systems, one often has to trade-off accuracy for power savings by using approximate computing (AC) techniques [3]. One example is reduced precision (RP), in which data is represented with fewer bits than desired throughout the

entire computational stack. As arbitrary precision arithmetic is not supported in many modern processors, a flexible hardware architecture enabling RP, such as reconfigurable platform using an FPGA, is demanded.

Our goal. We aim to implement an efficient solution to LASSO MPC and deploy it on an FPGA with the goal of achieving real-time performance [4]. Our optimization algorithm of choice is proximal gradient descent (PGD) [5]. This is sufficiently tractable to allow us to analyze the effects of different RP strategies on accuracy, logic and memory resources, and power consumption.

Contributions. We summarize our contributions as follows:

- We apply PGD to LASSO MPC which, to the best of our knowledge, has never been done, and analyze its performance. The application of accelerated versions of PGD [5] should be immediate.
- We introduce an Approximate Core (AC) synthesis infrastructure.
- Using our infrastructure, we then conduct a detailed study of the effects of the AC strategy on our algorithm.

II. BACKGROUND

We briefly explain MPC, LASSO MPC, and then review different number representations and their use in RP computing.

State-space models. We consider state-space representations of linear time-invariant (LTI) discrete systems. At each time, t , such systems are completely described by a state vector $x[t] \in \mathbb{R}^n$, which is known to evolve as

$$x[t+1] = Ax[t] + Bu[t], \quad t = 0, 1, \dots, \quad (1)$$

where $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times m}$ are given matrices (assumed known), and $u[t] \in \mathbb{R}^m$ is the set of inputs at time t .

Model predictive control (MPC). Given a finite time-horizon $T \in \mathbb{N}$, a desired final state $x_f \in \mathbb{R}^n$, and an estimate of the current state $x_0 \in \mathbb{R}^n$, MPC attempts to compute a minimal energy state-trajectory such that the final state $x[T]$ is as close as possible to the desired one, x_f . This can be formulated as an optimization problem:

$$\begin{aligned} \min_{\bar{x}, \bar{u}} \quad & F(x[T]) + \sum_{t=0}^{T-1} \ell(x[t], u[t]) \\ \text{s.t.} \quad & x[t+1] = Ax[t] + Bu[t], \quad t = 0, \dots, T-1 \\ & x[0] = x_0, \end{aligned} \quad (2)$$

Work supported by EPSRC Grant number EP/S000631/1 and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

where $(\bar{x}, \bar{u}) := (\{x[t]\}_{t=0}^T, \{u[t]\}_{t=0}^{T-1}) \in \mathbb{R}^{n(T+1)} \times \mathbb{R}^{mT}$ is the optimization variable, $F : \mathbb{R}^n \rightarrow \mathbb{R}$ a function that penalizes deviations of the final state $x[T]$ from the desired one, and $\ell : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ a function that measures “energy” at each time instant. Notice that the first set of constraints is exactly (1), and the second set reflects the current state.

The functions F and ℓ are often quadratic forms. An example assuming $x_f = 0$ would be

$$F(x) = x^\top P x \quad (3a)$$

$$\ell(x, u) = x^\top Q x + u^\top R u, \quad (3b)$$

where $P, R \succ 0$ are positive definite matrices and $Q \succeq 0$ is positive semidefinite. With this choice, the variable \bar{x} in (2) can be eliminated. To see why, first write (1) as [6, Ch. 8]

$$\underbrace{\begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[T] \end{bmatrix}}_{=:\bar{x}} = \underbrace{\begin{bmatrix} I_n \\ A \\ \vdots \\ A^T \end{bmatrix}}_{=:\bar{A}} x_0 + \underbrace{\begin{bmatrix} 0 & \cdots & 0 \\ B & \cdots & 0 \\ \vdots & \ddots & \vdots \\ A^{T-1}B & \cdots & B \end{bmatrix}}_{=:\bar{B}} \underbrace{\begin{bmatrix} u[0] \\ u[1] \\ \vdots \\ u[T-1] \end{bmatrix}}_{=:\bar{u}}, \quad (4)$$

where I_n is the identity matrix in \mathbb{R}^n . Placing the resulting equation into the cost of (2) and manipulating, we obtain

$$\underset{\bar{u}}{\text{minimize}} \quad \frac{1}{2} \bar{u}^\top (\bar{B}^\top \bar{Q} \bar{B} + \bar{R}) \bar{u} + (\bar{B}^\top \bar{Q} \bar{A} x_0)^\top \bar{u}, \quad (5)$$

where $\bar{R} := I_T \otimes R$, and \bar{Q} is the diagonal concatenation of $I_T \otimes Q$ and P (\otimes denotes the Kronecker product). Problem (5) is unconstrained quadratic, and thus has a closed-form solution. In MPC [6], whenever (5) is solved, only the first input $u[0]$ is applied to the system, the resulting state is measured, and (5) is solved again using x_0 as the current state.

LASSO MPC. Although (5) has a closed-form solution, it typically yields dense (i.e., non-sparse) inputs, which can lead to over-actuated systems. To encourage sparse inputs, [2] proposed LASSO MPC which adds an ℓ_1 -norm penalty $\lambda \|\bar{u}\|_1$, with $\lambda > 0$, to (5) [or, equivalently, $\lambda \|u\|_1$ to (3b)]. Noticing that the objective of (5) can be written as $\frac{1}{2} \|H\bar{u} - y\|_2^2 - \frac{1}{2} \|y\|_2^2$, with $H := (\bar{B}^\top \bar{Q} \bar{B} + \bar{R})^{1/2}$ and $y := -H^{-1} \bar{B}^\top \bar{Q} \bar{A} x_0$, the resulting problem is

$$\underset{\bar{u}}{\text{minimize}} \quad \frac{1}{2} \|H\bar{u} - y\|_2^2 + \lambda \|\bar{u}\|_1, \quad (6)$$

which has format of LASSO [7]. However, the matrix H in (6) is square. So, instead of regularizing the problem as in classical LASSO, the ℓ_1 -norm term enforces sparse inputs at the cost of possibly not reaching (or delaying) the desired state. Notice also that because we eliminated the state variable \bar{x} from (2) to (5), the system dynamics are always (implicitly) satisfied.

Reduced precision (RP). Some platforms require representing arithmetic numbers with short binary codes, thereby reducing their accuracy. There are three main categories of representation: floating point [8], Q fixed point [9], and universal numbers (Unum) [10].

The IEEE 754 floating point arithmetic standard [11] represents a number by using three elements: a sign (1 bit), an exponent (n bits), and a mantissa (m bits). That is,

$$\text{sign} \times \text{mantissa} \times 2^{\text{exponent}}. \quad (7)$$

The Q fixed point format uses instead a fixed number of bits to represent the integer and the fractional parts of a number. Specifically, a number is represented by its sign (1 bit), its integer part (t bits), and its fractional part (k bits):

$$\text{sign} \times (2^{\text{integer}} + 2^{-\text{fraction}}). \quad (8)$$

Unum arose as an alternative to IEEE 754, and there are several versions. For example, Type III Unum, Posit, represents numbers by their sign (1 bits), regime (g bits), exponent (p bits), and fractional part (c bits):

$$\text{sign} \times (2^{2^p})^{\text{regime}} \times 2^{\text{exponent}} \times \left(1 + \frac{\text{fraction}}{2^c}\right). \quad (9)$$

Each of these representations has a different dynamic range. And different dynamic ranges affect not only the performance of the algorithm by limiting the type of operations that can be performed, but also their embedded implementation.

III. ALGORITHM AND IMPLEMENTATION

We now explain the algorithm we use to solve LASSO MPC and its implementation on resource-constrained systems.

A. Proximal Gradient Descent

We consider the LASSO problem in (6) and apply the proximal gradient descent (PGD) algorithm with fixed step size [12]. PGD solves problems of the form

$$\underset{\bar{u}}{\text{minimize}} \quad f(\bar{u}) := g(\bar{u}) + h(\bar{u}), \quad (10)$$

where $g : \mathbb{R}^q \rightarrow \mathbb{R}$ is convex, differentiable, and its gradient is Lipschitz-continuous, i.e., there exists $L > 0$ such that $\|\nabla g(\bar{u}) - \nabla g(v)\|_2 \leq L \|\bar{u} - v\|_2$, for all \bar{u}, v . The function $h : \mathbb{R}^q \rightarrow \mathbb{R} \cup \{+\infty\}$ is assumed convex and closed. Given a stepsize $\alpha \leq 1/L$ and an initial point \bar{u}^0 , PGD solves (10) by iterating (on k)

$$\bar{u}^{k+1} = \text{prox}_{\alpha h}(\bar{u}^k - \alpha \nabla g(\bar{u}^k)), \quad (11)$$

where the proximal operator of a convex, closed function ϕ at a point u is defined as

$$\text{prox}_\phi(\bar{u}) := \arg \min_v \phi(v) + \frac{1}{2} \|v - \bar{u}\|_2^2. \quad (12)$$

It is well known that the iterates produced by PGD satisfy [12]

$$f(\bar{u}^k) - f(\bar{u}^*) \leq \frac{\alpha L \|\bar{u}^0 - \bar{u}^*\|_2^2}{2k}, \quad (13)$$

where \bar{u}^* is a solution of (10), i.e., PGD converges sublinearly.

Application to LASSO MPC. As problem (6) has the format of (10) with $g(\bar{u}) = (1/2) \|H\bar{u} - y\|_2^2$ and $h(\bar{u}) = \lambda \|\bar{u}\|_1$,

the application of PGD is immediate and yields the soft-thresholding algorithm. Specifically, (11) becomes

$$\begin{aligned} \bar{u}^{k+1} &= \mathcal{S}_\lambda(\bar{u}^k - \alpha H^\top (H\bar{u}^k - y)) \\ &= \mathcal{S}_\lambda\left((I_q - \alpha H^\top H)\bar{u}^k + \alpha H^\top y\right), \end{aligned} \quad (14)$$

where the soft-thresholding operator $\mathcal{S}_\lambda(\bar{u})$ applies to component i , for $i = 1, \dots, q$, the following nonlinearity:

$$\left[\mathcal{S}_\lambda(\bar{u})\right]_i = \begin{cases} \bar{u}_i - \lambda & , \bar{u}_i > \lambda \\ 0 & , -\lambda \leq \bar{u}_i \leq \lambda \\ \bar{u}_i + \lambda & , \bar{u}_i < -\lambda. \end{cases} \quad (15)$$

Parameters and precomputations. PGD (11) and the associated convergence result in (13) apply whenever the stepsize α satisfies $\alpha \leq 1/L$, where L is a Lipschitz constant of ∇g . To compute it, we can estimate the maximum eigenvalue of $H^\top H$, $\lambda_{\max}(H^\top H)$, e.g. by Lanczos’s method, or use the structure of H and the matrices that define it:

Lemma 1. Let $H = (\bar{B}^\top \bar{Q} \bar{B} + \bar{R})^{1/2}$, where \bar{B} , \bar{Q} , \bar{R} are defined in (4)-(5). Also, partition \bar{B} vertically into $\bar{B}_1 \in \mathbb{R}^{nT \times mT}$, which contains the first nT rows of \bar{B} , and into $\bar{B}_2 \in \mathbb{R}^{n \times mT}$, which contains the last n rows of \bar{B} . Then,

$$\begin{aligned} \lambda_{\max}(H^\top H) &\leq \lambda_{\max}(Q)\lambda_{\max}(\bar{B}_1 \bar{B}_1^\top) \\ &\quad + \lambda_{\max}(P)\lambda_{\max}(\bar{B}_2 \bar{B}_2^\top) + \lambda_{\max}(R). \end{aligned} \quad (16)$$

The proof uses the subadditivity of $\lambda_{\max}(\cdot)$, and the properties of the Kronecker product; it is omitted for brevity. The matrices P , Q , and R encode the objective of MPC [cf. (3)], and the matrices \bar{B}_1 and \bar{B}_2 encode the system dynamics through their dependency on A and B [cf. (1)]. Although the matrices \bar{B}_1 and \bar{B}_2 have a lot of structure, namely \bar{B}_1 is block Toeplitz and \bar{B}_2 is the controllability matrix with permuted columns, it is not immediate to obtain a bound on $\lambda_{\max}(\bar{B}_1 \bar{B}_1^\top)$ and $\lambda_{\max}(\bar{B}_2 \bar{B}_2^\top)$ as a function of A and B . These quantities can be accurately estimated via Lanczos’s method.

Once α is set to the inverse of right-hand side of (16), the matrix $I_q - \alpha H^\top H$ and the vector $H^\top y = \bar{B}^\top \bar{Q} \bar{A} x_0$ in (14) can be precomputed before the iterations of the algorithm. We stop the algorithm whenever a maximum number of iterations k_{\max} is reached or when $|f(\bar{u}^{k+1}) - f(\bar{u}^k)| < \epsilon$, for some defined ϵ .

B. Approximate Core Synthesis

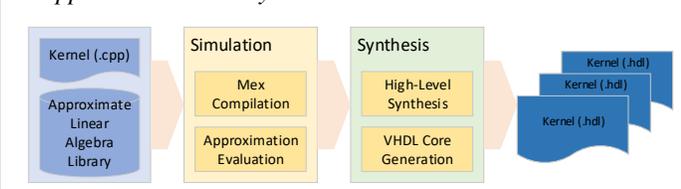


Fig. 1: Approximate kernel synthesis.

We now describe a proof-of-concept infrastructure that generates an approximate optimizer for (5) on a reconfigurable device. Fig. 1 shows the workflow of this infrastructure. The

optimizer kernel is a user-defined C++ function based on an approximate linear algebra library, which defines basic algebra operations such as addition, multiplication, inversion, and decomposition, for matrices and vectors. We have developed that algebraic library for arbitrary precision representations.

We solved LASSO MPC using different precisions using an integrated environment with the Matlab MEX API. By compiling the MEX files with the proposed kernel, we can evaluate the functionality and algorithmic performance of the algorithm. After checking the correctness of the kernel via functionality simulation, the kernel is synthesized using high-level synthesis tools, and the approximate core is generated in VHDL.

IV. EXPERIMENTS

We now describe experiments using LASSO MPC to control the attitude of a spacecraft [13] by ACADO [14]. Fig. 2 shows the state parameters for attitude control using reaction wheels. The state vector, corresponding to x in (1), is defined by $[roll, pitch, yaw, \omega_\omega, \omega_1, \omega_2, \omega_3]$ in Fig. 2. The control voltages for the reaction wheels that steer the spacecraft, $[\tau_1, \tau_2, \tau_3, \tau_\omega]$, are considered as the components of the input vector, u , in (1). In our experiments, the dynamic matrices A and B in (1) and the cost matrices P , Q , and R in (3) were set exactly as in [13].

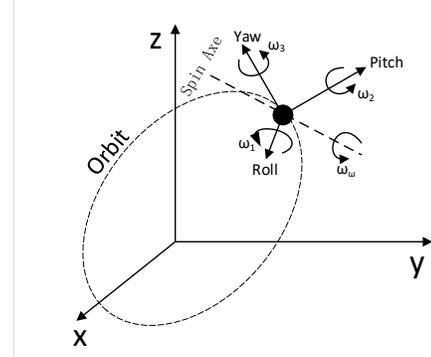


Fig. 2: Attitude Control [13]: seven states are considered here, Roll, Pitch, Yaw, $\omega_1, \omega_2, \omega_3, \omega_\omega$, where Roll, Pitch, Yaw describe the rotating angles of the body frame relative to the orbit frame, $\omega_1, \omega_2, \omega_3$ are the corresponding angular velocities, and ω_ω is the angular velocity along the spin axis. The wheels are controlled by the input voltages, $\tau_1, \tau_2, \tau_3, \tau_\omega$.

Experimental setup. We considered three different time horizons T : 1, 5, and 10. To assess our implementation of PGD, we compared its solution with the one returned by CVX [15], and used the sum of the absolute differences, $D = |u_{RP}^k - u_{cvx}|_1$ for k^{th} iteration, of the control input vector, $[\tau_1, \tau_2, \tau_3, \tau_\omega]$, as the performance metric. For example, $D_{Float-10}^k = |u_{Float-10}^k - u_{cvx}|_1$ is the solution between floating point using 10 bits and CVX at the k^{th} iteration, where $u_{Float-10}^k$ is the solution of (6) using a 10 bit floating point representation, and u_{cvx} is the solution returned by CVX.

A. RP Approximation Performance

We implemented the framework described in section III-B with different number representations, each with its own preci-

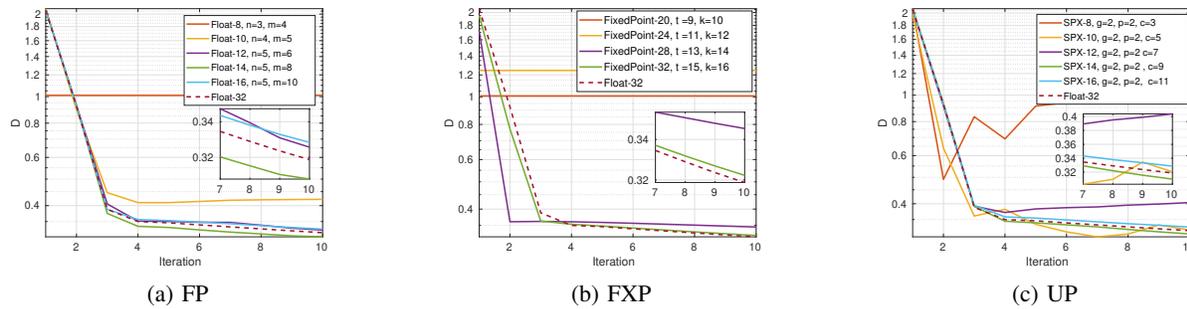


Fig. 3: Convergence performance of approximate PGD (14) between the baseline 32-bit floating point (FP) and reduced precision (RP) implementations using (a) Floating Point (FP), (b) Fixed Point (FXP), and (c) Unum Posit (UP).

sion. Fig. 3a shows the baseline, single precision floating point (32 bits) simulation using CVX as the Lasso MPC solver, compared against our reduced precision implementations described in paragraphs (a), (b), and (c) below. Due to limited space, we only consider $T = 10$ here. Similar phenomena happen for $T = 1$ and $T = 5$.

- (a) Floating Point (FP): In Fig. 3a RP floating point solutions are presented for PGD from 6-bit to 16-bit. As shown, performance with 8 to 10 bits is much worse compared to single precision float. However, from 12 bits the performance is considerable, which is similar to single precision floating. Hence, 12-bit floating point approximation is considered as a viable candidate for approximate FP implementation.
- (b) Fixed Point (FXP): In Figure 3b, RP fixed point is presented for PGD from 20-bit to 32-bit. As shown, proper convergence does not occur for 20 or 24 bit approximation, while similar performance is obtained using 32 bits. Hence, 32-bit fixed point is considered as a candidate for approximate FXP implementation.
- (c) Unum Posit (UP): In Fig. 3c, the RP unum posit is considered for PGD from 8 to 16 bits. Compared to the single precision float, similar performance is obtained from 14 bits which makes it the candidate of RP approximate PGD as well.

Fig. 4 shows the comparison of the the control state in an open-loop simulation with different RP techniques. As shown in Fig. 4b-4d, compared to the CVX solution using the baseline 32 point FP precision solution of Fig. 4a, the approximate solver shows very similar control process and performance to CVX over time. For example, in Fig. 4b, the PGD optimized input using 12 bits FP exhibits slightly larger variance on than CVX, while the PGD optimized output for Euler and Angular control takes about 1.5s to converge rather than about 1.1s with CVX. However, an approximation with over-reduced precision causes instability in the spacecraft attitude is shown in Fig. 4e.

B. Cost Evaluation

Choosing the closest MPC performance of each RP approximation to CVX across the different precisions shown in the last section, the corresponding costs are evaluated by implementing the PGD kernel on an Xilinx Ultrascale+ ZCU106 device as

shown in Table I. The single 32-bit FP precision implementation is considered again as the baseline against which other representations are compared. All costs are estimated from high level synthesis using the Xilinx Vivado design tool 2019.2.

Across different horizon lengths, the approximate fixed point implementation (FXP-24,28,32) consumes the least logic area, which is about 30% of the baseline (FP32) in each case. Accordingly, the power consumption is reduced to 25% when the horizon length is 1, and <50% when the horizon length is 5 or 10. As the problem size increases with horizon length, the number of bits for fixed point arithmetic grows from 24 to 32 to maintain a performance similar to the single precision implementation. This is due to the fact that the dynamic range of linear algebra, numeric computations is larger as the size increases.

The (FP-12, 16,14) implementation does not save as much logic area as fixed point, but as the horizon length increases, up to 40% reduction in the use of LUT is achieved. Accordingly, up to 20% power reduction is introduced, but only within a very limited time horizon, $T = 1$. However, the least number of bits are adopted compared to fixed point implementations, enabling significant saving in communication bandwidth to 14.06% of FP-32.

The approximate unum posit (UP-12,14) is based on high level synthesis of SoftPosit [16]. It consumes significantly larger logic area comparing to floating point and fixed point implementations, although similar savings on communication bandwidth are made due to the low number of representative bits.

Hence, if power and area are of most concern on a resource constrained system, the fixed point (FXP) lean PGD does provide significant savings compared to the full, single precision, FP-32 implementation. However, if communication is of the most concern in a resource constrained system, such as a mesh network with edge devices, the FP floating point variations show the largest bandwidth savings.

V. CONCLUSIONS

In this work, an approximate proximal gradient descent is applied to solve Lasso MPC with fixed step size. By adopting the reduced precision technique, a considerable optimization performance is achieved compared to high-precision solver

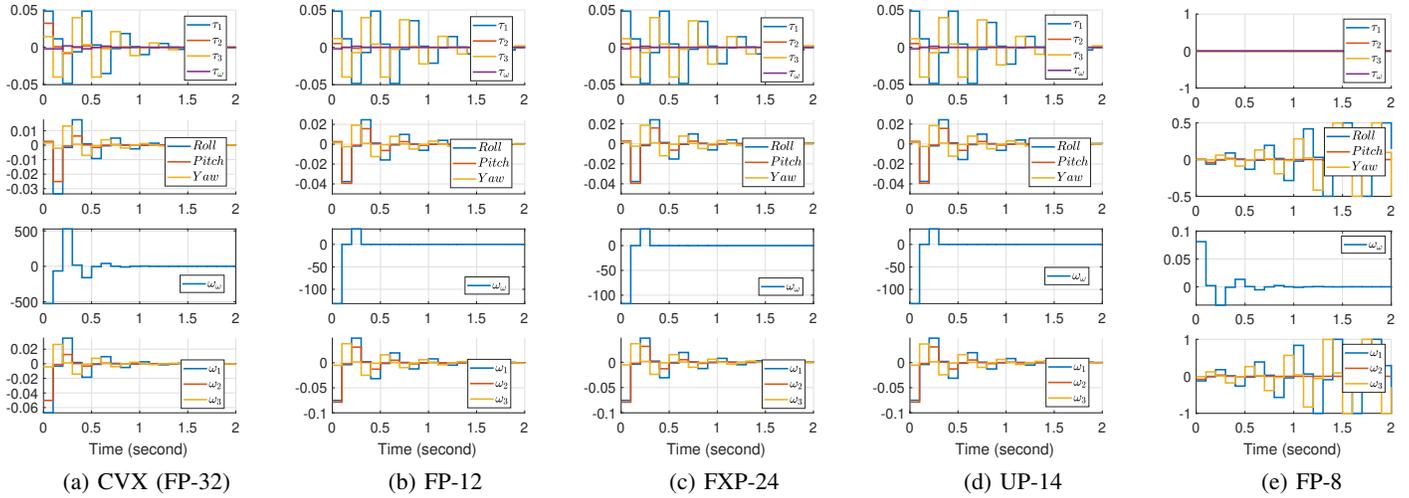


Fig. 4: LASSO MPC Simulation: $[\tau_1, \tau_2, \tau_3, \tau_\omega]$ is the control (input) vector in Volt while $[roll, pitch, yaw, \omega_\omega, \omega_1, \omega_2, \omega_3]$ is the state (output) vector. $roll, pitch, yaw$ are measured in Degrees and $\omega_\omega, \omega_1, \omega_2, \omega_3$ are measured in Rad/s. The simulation shows all input and output states changing over 2s with a 0.1s step interval.

TABLE I: RP Cost Comparison with Different Horizon Length (T)

Precision	T=1				T=5				T=10			
	FP-32	FP-12	FXP-24	UP-12	FP-32	FP-16	FXP-28	UP-14	FP-32	FP-14	FXP-32	UP-14
LUT ($\times 10^3$)	3.31	2.99	1.21	19.4	3.17	2.16	1.24	10.9	4.01	2.42	1.46	10.6
DSP48E1	30	0	11	12	20	6	10	4	20	6	16	4
BRAM	0	0	0	0	2	2	4	0	5	5	5	8
Clock (MHz)	482	465	443	393	434	401	403	382	370	384	379	382
T (M Inst/sec)	0.42	0.48	0.46	0.41	0.45	0.41	0.42	0.39	0.38	0.39	0.39	0.39
Power (mW)	273	199	68	248	219	220	110	152	250	254	113	148
Bandwidth	100%	14.06%	56.25%	14.06%	100%	25%	76.56%	19.14%	100%	19.14%	100%	19.14%

(CVX). An approximate core synthesis infrastructure is developed for fast prototyping the computational kernel of proximal gradient descent on reconfigurable device, FPGA. The results show up to 60% in logic cost reduction, 80% in memory bandwidth saving, and 70% in power reduction, which is very promising for deploying LASSO MPC on resource constrained system considering computing and communication cost. The future work includes further exploring the approximation effects against different MPC applications as well as different optimization algorithms.

REFERENCES

- [1] S. Qin and T. A. Badgwell, "A Survey of Industrial Model Predictive Control Technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733 – 764, 2003. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0967066102001867>
- [2] M. Gallieri and J. M. Maciejowski, "LASSO MPC: Smart Regulation of Over-Actuated Systems," in *2012 American Control Conference (ACC)*, 2012, pp. 1217–1222.
- [3] A. Agrawal, J. Choi, K. Gopalakrishnan, S. Gupta, R. Nair, J. Oh, D. A. Prener, S. Shukla, V. Srinivasan, and Z. Sura, "Approximate computing: Challenges and opportunities," in *2016 IEEE International Conference on Rebooting Computing (ICRC)*, 2016, pp. 1–8.
- [4] K. Ling, B.-F. Wu, and J. Maciejowski, "Embedded Model Predictive Control (MPC) using a FPGA," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 17, 01 2008.
- [5] A. Beck and M. Teboulle, "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems," *SIAM J. Im. Sc.*, vol. 2, no. 1, pp. 183–202, 2009.
- [6] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control For Linear And Hybrid Systems*. Cambridge University Press, 2017.
- [7] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *J. Royal Statistical. Soc., Series B*, vol. 58, no. 1, pp. 267–288, 1996.
- [8] G. Flegar, F. Scheidegger, V. Novaković, G. Mariani, A. E. Tom' s, A. C. I. Malossi, and E. S. Quintana-Ortí, "FloatX: A C++ Library for Customized Floating-Point Arithmetic," *ACM Transactions on Mathematical Software (TOMS)*, vol. 45, no. 4, Dec. 2019. [Online]. Available: <https://doi.org/10.1145/3368086>
- [9] S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. USA: California Technical Publishing, 1997.
- [10] J. Gustafson and I. Yonemoto, "Beating Floating Point at Its Own Game: Posit Arithmetic," *Supercomputing Frontiers and Innovations*, vol. 4, pp. 71–86, 01 2017.
- [11] J.-M. Muller, N. Brunie, F. de Dinechin, C.-P. Jeannerod, M. Joldes, V. Lefvre, G. Melquiond, N. Revol, and S. Torres, *Handbook of Floating-Point Arithmetic*, 2nd ed. Birkhäuser Basel, 2018.
- [12] A. Beck and M. Teboulle, *Convex Optimization in Signal Processing and Communications*. Cambridge University Press, 2010, ch. Gradient-based algorithms with applications to signal-recovery problems.
- [13] Øyvind Hegrenæs, J. T. Gravdahl, and P. Tøndel, "Spacecraft Attitude Control using Explicit Model Predictive Control," *Automatica*, vol. 41, no. 12, pp. 2107 – 2114, 2005. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0005109805002657>
- [14] B. Houska, H. Ferreau, and M. Diehl, "ACADO Toolkit – An Open Source Framework for Automatic Control and Dynamic Optimization," *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, 2011.
- [15] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," <http://cvxr.com/cvx>, Mar. 2014.
- [16] S. H. Leong, NGA SoftPosit, 2017 (accessed July 3, 2019). [Online]. Available: <https://gitlab.com/cerlane/SoftPosit>

Extraction of Analytic Eigenvectors From a Parahermitian Matrix

Stephan Weiss¹, Ian K. Proudler¹, Fraser K. Coutts², and Julian Deeks³

¹ Department of Electronic & Electrical Engineering, University of Strathclyde, Glasgow G1 1XW, Scotland

² Institute for Digital Communications, University of Edinburgh, Edinburgh, Scotland

³ DSTL Portsdown, Fareham, Hampshire PO17 6AD, UK

{stephan.weiss,ian.proudler}@strath.ac.uk, fraser.coutts@ed.ac.uk

Abstract—The space-time covariance matrix derived from broadband multichannel data admits — unless the data emerges from a multiplexing operation — a parahermitian matrix eigenvalue decomposition with analytic eigenvalues and analytic eigenvectors. The extraction of analytic eigenvalues has been solved previously in the discrete Fourier transform (DFT) domain; this paper addresses the approximation of analytic eigenvectors in the DFT domain. This is a two-stage process — in the first instance, we identify eigenspaces in which analytic eigenvectors can reside. This stage resolves ambiguities at frequencies where eigenvalues have algebraic multiplicities greater than one. In a second stage, the phase ambiguity of eigenvectors is addressed by determining a maximally smooth phase response. Finally, a metric for the approximation error is derived, which allows us to increase the DFT length and iterate the two stages until a desired accuracy is reached.

I. INTRODUCTION

For broadband multichannel data collected by M sensors in a time series vector $\mathbf{x}[n] \in \mathbb{C}^M$ with discrete time index n , second order statistics are captured by the space-time covariance matrix $\mathbf{R}[\tau] = \mathcal{E}\{\mathbf{x}[n]\mathbf{x}^H[n-\tau]\}$, with $\mathcal{E}\{\cdot\}$ representing the expectation operator, $\{\cdot\}^H$ the Hermitian transposition, and τ a lag parameter. This space-time covariance matrix generalises the narrowband or instantaneous covariance matrix $\mathbf{R}[0]$ to the broadband case, and leads to simple formulations of broadband problems analogous to their narrowband equivalent. To solve narrowband problems, factorisations of the covariance matrix, particularly its eigenvalue decomposition (EVD), are key. Hence, in the broadband case a similar factorisation is desired to generalise the utility of the EVD from the narrowband case.

The diagonalisation of $\mathbf{R}[\tau]$ for all lags τ , or equivalently of its z -transform, the cross-power spectral density (CSD) $\mathbf{R}(z) = \sum_{\tau} \mathbf{R}[\tau]z^{-\tau}$ for all values of τ , has been investigated in [1]–[3]. Since $\mathbf{R}[\tau]$ comprises auto- and cross-correlation sequences, $\mathbf{R}(z)$ is parahermitian, such that $\mathbf{R}^P(z) = \mathbf{R}^H(1/z^*) = \mathbf{R}(z)$, where the parahermitian operator $\{\cdot\}^P$ imposes a time reversal and Hermitian transposition. If the data $\mathbf{x}[n]$ is generated by causal and stable systems [4], then $\mathbf{R}(z)$ is analytic in z . If there is no temporal multiplexing

of the data involved, then such an analytic $\mathbf{R}(z)$ admits a parahermitian matrix EVD (PhEVD) [2], [3],

$$\mathbf{R}(z) = \mathbf{Q}(z)\mathbf{\Lambda}(z)\mathbf{Q}^P(z), \quad (1)$$

with an analytic and unitary $\mathbf{Q}(z)$ and analytic, diagonal, and parahermitian $\mathbf{\Lambda}(z)$. The diagonal elements of $\mathbf{\Lambda}(z)$ are referred to the analytic eigenvalues of $\mathbf{R}(z)$, and the columns of $\mathbf{Q}(z)$ their corresponding analytic eigenvectors.

In the past, a number of algorithms have targeted decompositions similar to (1). In the McWhirter or polynomial matrix EVD (PEVD) [1] with $\mathbf{R}(z) \approx \mathbf{U}(z)\mathbf{\Gamma}(z)\mathbf{U}^P(z)$, a paraunitary matrix $\mathbf{U}(z)$ achieves an approximate diagonalisation with spectrally majorised eigenvalues, i.e. the latter appear strictly in descending order at every frequency. Algorithms to accomplish this, such as the second order sequential best rotation (SBR2, [1]) or the sequential matrix diagonalisation (SMD, [4]) algorithms encourage or can even be shown to converge towards spectral majorisation [5]. If diagonal elements in $\mathbf{\Lambda}(z)$ intersect on the unit circle, then $\mathbf{\Gamma}(z)$ will converge towards a non-differential and hence non-differentiable permutation of $\mathbf{\Lambda}(z)$. Similarly, $\mathbf{U}(z)$ will converge towards a permuted, and hence discontinuous version of $\mathbf{Q}(z)$. Therefore, the factors $\mathbf{\Gamma}(z)$ and $\mathbf{U}(z)$ typically require significantly higher order polynomials for similar approximation error as achievable with analytic $\mathbf{\Lambda}(z)$ and $\mathbf{Q}(z)$.

In order to obtain lower-order approximations of the analytic factors in the PhEVD in (1), a first attempt has been reported in [6], where for a fixed DFT size, the eigenvectors were used to re-establish the coherence across bins that is otherwise lost when calculating EVDs independently in different frequency bins. To base this coherence on the eigenvectors can lead to robustness problems: eigenvectors are typically more strongly perturbed than eigenvalues [7]; in particular, at algebraic multiplicities $C > 1$ of eigenvalues, the associated eigenvectors can form an arbitrary C -dimensional basis and hence will not necessarily yield a smooth and therefore analytic continuation of eigenvectors across these algebraic multiplicities.

Therefore, we previously have based the association of eigenvalues across frequency bins on the better conditioned eigenvalues instead of the eigenvectors [8], [9]. We use the result of this association in this paper to also extract an approximation for the analytic eigenvectors. This raises two challenges. First, continuous 1-d subspaces have to be woven

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/S000631/1 and the MOD University Defence Research Collaboration in Signal Processing.

through C -d subspaces at algebraic multiplicities. Secondly, the arbitrary phase function of eigenvectors must be selected as maximally smooth in order to approximate the analytic eigenvectors with a low approximation error. This second step has e.g. been addressed in [10], which uses a gradient technique driven by a smoothness metric that is to be minimised. Thus, in Sec. II, we highlight the general approach, address smooth 1-d subspaces in Sec. III and propose phase smoothing similar to [10] in Sec. IV. Finally, Sec. V assess the overall approximation error, and Sec. VI provides some results.

II. ANALYTIC PARAHERMITIAN MATRIX EIGENVALUE DECOMPOSITION AND GENERAL APPROACH

A. Properties of the PhEVD

We assume that the space-time covariance matrix $\mathbf{R}[\tau] \circ \bullet \mathbf{R}(z)$ is not derived from multiplexed data, such that the PhEVD of $\mathbf{R}(z) = \mathbf{Q}(z)\mathbf{\Lambda}(z)\mathbf{Q}^P(z)$ exists with analytic factors $\mathbf{\Lambda}(z)$ and $\mathbf{Q}(z)$. From [2], [3] we know that for these factors, $\mathbf{\Lambda}(z)$ is unique up to a reordering. We further assume that there are no identical eigenvalues, i.e. that with $\mathbf{\Lambda}(z) = \text{diag}\{\lambda_1(z), \dots, \lambda_M(z)\}$, there are no functions such that $\lambda_m(z) = \lambda_\mu(z)$, $m \neq \mu$ holds for all values of z within the region of convergence. Due to the uniqueness theorem of analytic functions, $\lambda_m(z)$ and $\lambda_\mu(z)$ for $m \neq \mu$ can only share a finite number of crossing points, i.e. when evaluated on the unit circle, the eigenvalues $\lambda_m(e^{j\Omega})$, $m = 1, \dots, M$ will only possess algebraic multiplicities at a finite number of values $\Omega \in [0, 2\pi)$.

The matrix of eigenvectors, $\mathbf{Q}(z) = [\mathbf{q}_1(z), \dots, \mathbf{q}_m(z)]$, is not uniquely defined, and each eigenvector $\mathbf{q}_m(z)$, $m = 1, \dots, M$, can be modified by an arbitrary allpass function as a generalisation of the phase ambiguity of eigenvectors for the standard EVD of constant Hermitian matrices [12].

B. Evaluation on the Unit Circle

By working in the DFT domain, we evaluate $\mathbf{R}(z)$ in K equispaced bins along the unit circle, such that ideally

$$\mathbf{R}(e^{j\Omega_k}) = \mathbf{Q}(e^{j\Omega_k})\mathbf{\Lambda}(e^{j\Omega_k})\mathbf{Q}^H(e^{j\Omega_k}) \quad (2)$$

with $\Omega_k = \frac{2\pi}{K}k$, $k = 0, \dots, (K-1)$. However, by working in isolated frequency bins, the coherence is lost, and in the k th bin we obtain

$$\mathbf{R}(e^{j\Omega_k}) = \mathbf{Q}_k\mathbf{\Lambda}_k\mathbf{Q}_k^H, \quad (3)$$

where the r.h.s. factors are without direct equivalence to the terms on the r.h.s. of (2).

In the k th bin the eigenvalues may appear randomly reordered. As a fixture, we assume the $\mathbf{\Lambda}_k$ is majorised. Therefore

$$\mathbf{\Lambda}(e^{j\Omega_k}) = \mathbf{P}_k\mathbf{V}_k\mathbf{\Lambda}_k\mathbf{V}_k^H\mathbf{P}_k^T, \quad (4)$$

where \mathbf{P}_k is a permutation matrix that reorders the otherwise spectrally majorised eigenvalues in each bin, such that they

correspond to the analytic ones. The unitary \mathbf{V}_k is a block diagonal matrix

$$\mathbf{V}_k = \text{diag}\{\mathbf{V}_{k,1}, \dots, \mathbf{V}_{k,L_k}\}, \quad (5)$$

where L_k is the number of distinct eigenvalues in the k th bin, and $\mathbf{V}_{k,\ell} \in \mathbb{C}^{Q_{k,\ell} \times Q_{k,\ell}}$, $\ell = 1, \dots, L_k$, is unitary with a dimension equivalent to the algebraic multiplicity $Q_{k,\ell}$ of the corresponding eigenvalue. Note that $\sum_{\ell=1}^{L_k} Q_{k,\ell} = M$. In case of $L_k = M$ distinct eigenvalues, \mathbf{V}_k is a diagonal matrix imposing phase shifts only.

Therefore, for the matrix of eigenvectors, we have

$$\mathbf{Q}(e^{j\Omega_k}) = \mathbf{Q}_k\mathbf{V}_k^H\mathbf{P}_k^T. \quad (6)$$

From the extraction of analytic eigenvalues in [8], [9], we know the permutation matrices \mathbf{P}_k , and the dimensions of the unitary subblocks $\mathbf{V}_{k,\ell}$, but not their values.

C. Problem Statement and Approach

A number of problems remain in extracting analytic eigenvectors:

P1 (subspace alignment): In case of algebraic multiplicities, where at least some of the subblocks $\mathbf{V}_{k,\ell}$ exceed dimension one, we need to weave continuous one dimensional subspaces across these multiplicities.

P2 (phase alignment): The phase shift in individual frequency bins need to be aligned to provide an overall analytic function — and therefore achieve minimum support in the time domain.

P3 (approximation error): Regarding the number of frequency bins, from [8], [9] we know the value for K that guarantees an extraction of eigenvalues that approximates the analytic functions with a predefined accuracy. For the approximation of eigenvectors, a different, and potentially higher number of sample points, can be required. We therefore need a criterion that can determine whether the number of sample points (and therefore the approximation order) for the eigenvectors is sufficiently high.

In contrast to the only previous algorithm for the extraction of smooth eigenvectors in [6], which addresses problem *P2* for a fixed and predetermined order K , below we tackle problems *P1* to *P3* in Secs. III to V, respectively.

III. EXTRACTION OF SMOOTH 1-D EIGENSPACES

A. Rationale

In case of a C -fold algebraic multiplicity of eigenvalues, with $C > 1$, the corresponding eigenvectors can form an arbitrary basis within a C -dimensional subspace. However, to estimate analytic eigenvectors, we first need to weave smooth 1-d eigenspaces through such a manifold. Based on the assumption of non-identical eigenvalues and supported by the uniqueness theorem of analytic functions, we know that this can only occur at a finite and isolated number of frequency points.

Let Ω_{k_0} be a frequency bin where the eigenvalues $\lambda_m(e^{j\Omega_{k_0}}) = \dots = \lambda_{m+C-1}(e^{j\Omega_{k_0}})$ share a C -fold algebraic multiplicity. Since we know the analytic eigenvalues to a predefined accuracy, we can find frequency points $\Omega_{k_0} - \Delta$

and $\Omega_{k_0} + \Delta$ where $\Delta \ll 2\pi/K$ and the C eigenvalues are sufficiently distinct. Let \mathbf{Q}_{k_0-} and \mathbf{Q}_{k_0+} be the (appropriately ordered, using \mathbf{P}_{k_0-} and \mathbf{P}_{k_0+}) matrices of eigenvectors for $\mathbf{R}(e^{j(\Omega_{k_0}-\Delta)})$ and $\mathbf{R}(e^{j(\Omega_{k_0}+\Delta)})$. We first align phases across \mathbf{Q}_{k_0-} and \mathbf{Q}_{k_0+} , before interpolating through the appropriate C -dimensional subspace of \mathbf{Q}_{k_0} .

B. Phase-Alignment Across Algebraic Multiplicities

Let \mathbf{q}_{m,k_0-} and \mathbf{q}_{m,k_0+} be the m th eigenvectors in the columns of \mathbf{Q}_{k_0-} and \mathbf{Q}_{k_0+} . W.l.o.g., we retain \mathbf{Q}_{k_0-} as it is, but change the m th eigenvector at $\Omega_{k_0+\Delta}$ by a phase term ϑ_{m,k_0} that satisfies the optimisation problem

$$\vartheta_{\mu,k_0,\text{opt}} = \arg \min_{\vartheta_{\mu,k_0}} \|\mathbf{q}_{\mu,k_0-} - e^{j\vartheta_{\mu,k_0}} \mathbf{q}_{\mu,k_0+}\|_2^2, \quad \mu = m, \dots, m + C - 1. \quad (7)$$

By differentiating w.r.t. ϑ_{μ,k_0} and setting the gradient to zero, we obtain

$$\vartheta_{\mu,k_0,\text{opt}} = \arctan \frac{\text{Im}\{\mathbf{q}_{\mu,k_0-}^H \mathbf{q}_{\mu,k_0+}\}}{\text{Re}\{\mathbf{q}_{\mu,k_0-}^H \mathbf{q}_{\mu,k_0+}\}}. \quad (8)$$

With these phase shift angles $\vartheta_{\mu,k_0,\text{opt}}$, $\mu = m, \dots, m + C - 1$, the eigenvectors at $\Omega_{k_0} - \Delta$ and $\Omega_{k_0} + \Delta$ that are associated with the C -fold algebraic multiplicity in bin k_0 become aligned as closely as possible in the least squares sense.

C. Smooth Eigenspace Selection in Algebraic Multiplicities

Within bin k_0 , there is an ambiguity of eigenvectors w.r.t. an arbitrary unitary matrix $\mathbf{V}_{k_0} \in \mathbb{C}^{C \times C}$, such that with eigenvectors

$$\mathbf{U}_{k_0} = [\mathbf{q}_{m,k_0}, \mathbf{q}_{m+1,k_0}, \dots, \mathbf{q}_{m+C-1,k_0}], \quad (9)$$

the columns of $\mathbf{U}'_{k_0} = \mathbf{U}_{k_0} \mathbf{V}_{k_0}^H$ also represent valid eigenvectors. We want to solve the constrained problem

$$\begin{aligned} \min_{\mathbf{V}_{k_0}} & \|\mathbf{V}_{k_0} \mathbf{U}_{k_0}^H \mathbf{U}_{k_0-} - \mathbf{I}_C\|_F^2 + \|\mathbf{V}_{k_0} \mathbf{U}_{k_0}^H \mathbf{U}_{k_0+} \mathbf{\Theta}_{k_0} - \mathbf{I}_C\|_F^2 \\ \text{s.t.} & \quad \mathbf{V}_{k_0}^H \mathbf{V}_{k_0} = \mathbf{I}, \end{aligned} \quad (10)$$

where $\mathbf{U}_{k_0-} \in \mathbb{C}^{M \times C}$ and $\mathbf{U}_{k_0+} \in \mathbb{C}^{M \times C}$ contain the corresponding eigenvectors at the angular normalised frequencies $\Omega_{k_0} - \Delta$ and $\Omega_{k_0} + \Delta$, respectively. The diagonal matrix $\mathbf{\Theta}_{k_0} = \text{diag}\{e^{j\vartheta_{m,k_0}}, \dots, e^{j\vartheta_{m+C-1,k_0}}\}$ performs the phase alignment described in Sec. III-B. The latter is necessary such that a single unitary matrix \mathbf{V}_{k_0} can be used to compare bin k_0 to both the preceding and subsequent bins in (10).

To solve the constrained problem (10), [13] offers a number of approaches. This requires an expression for the cost, here denoted as ξ_{k_0} , which can be written as

$$\begin{aligned} \xi_{k_0} &= \|\mathbf{V}_{k_0} \mathbf{U}_{k_0}^H \mathbf{U}_{k_0-} - \mathbf{I}_C\|_F^2 + \|\mathbf{V}_{k_0} \mathbf{U}_{k_0}^H \mathbf{U}_{k_0+} \mathbf{\Theta}_{k_0} - \mathbf{I}_C\|_F^2 \\ &= \text{tr}\{(\mathbf{V}_{k_0} \mathbf{U}_{k_0}^H \mathbf{U}_{k_0-} - \mathbf{I}_C)(\mathbf{V}_{k_0} \mathbf{U}_{k_0}^H \mathbf{U}_{k_0-} - \mathbf{I}_C)^H\} \\ &\quad + \text{tr}\{(\mathbf{V}_{k_0} \mathbf{U}_{k_0}^H \mathbf{U}_{k_0+} \mathbf{\Theta}_{k_0} - \mathbf{I}_C) \cdot \\ &\quad \cdot (\mathbf{V}_{k_0} \mathbf{U}_{k_0}^H \mathbf{U}_{k_0+} \mathbf{\Theta}_{k_0} - \mathbf{I}_C)^H\}. \end{aligned} \quad (11)$$

Using matrix-valued differentiation [14] and Wirtinger calculus [15], we obtain as derivative $\nabla \xi_{k_0} = \frac{\partial \xi_{k_0}}{\partial \mathbf{V}_{k_0}}$

$$\nabla \xi_{k_0} = \mathbf{U}_{k_0}^H (\mathbf{U}_{k_0-} \mathbf{U}_{k_0-}^H - \mathbf{U}_{k_0+} \mathbf{U}_{k_0+}^H) \mathbf{U}_{k_0} \mathbf{V}_{k_0}^H - \mathbf{U}_{k_0}^H (\mathbf{U}_{k_0-} + \mathbf{U}_{k_0+} \mathbf{\Theta}_{k_0}). \quad (12)$$

Starting from an initial guess $\mathbf{V}_{k_0}^{(0)}$, this enables an iterative gradient search akin to [13], where in the i th iteration

$$\mathbf{V}_{k_0}^{(i-1/2)} = \mathbf{V}_{k_0}^{(i-1)} - \mu \nabla \xi_{k_0}(\mathbf{V}_{k_0}^{(i-1)}) \quad (13)$$

is no longer necessarily a unitary matrix. Therefore, (13) alternates with a projection

$$\mathbf{V}_{k_0}^{(i)} = \Pi(\mathbf{V}_{k_0}^{(i-1/2)}) \quad (14)$$

onto the Stiefel manifold of unitary matrices in $\mathbb{C}^{C \times C}$. The operation $\mathbf{B} = \Pi(\mathbf{A})$ that finds the unitary matrix closest to $\mathbf{A} \in \mathbb{C}^{C \times C}$ in the least squares sense is based on the SVD $\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^H$; in fact we have $\Pi(\mathbf{A}) = \mathbf{U}\mathbf{V}^H$ [12].

Thus, to solve the subspace alignment problem across algebraic multiplicities we first obtain $\mathbf{\Theta}_{k_0}$ via (8), and thereafter iterate (13) and (14) based on (12).

IV. PHASE ALIGNMENT WITHIN 1-D EIGENSPACES

A. Cost Function

If $\mathbf{q}_{m,k}$ is the k th bin of the m eigenvector, with $m = 1 \dots M$ and $\ell = 0 \dots (K-1)$, then although it now resides in a smooth eigenspace, we can still define an arbitrary phase shift $e^{j\varphi_{m,k}}$ for every such vector, such that $\mathbf{q}_{m,k} e^{j\varphi_{m,k}}$ also is a valid m th eigenvector in the k th frequency bin. These phase shifts must be selected such that each component of $\mathbf{q}_{m,k}$ can be interpreted as a sample point of a smooth function that is interpolated across the K bins. For each of the M components, $\mu = 1 \dots M$, these sample points can be organised into a vector $\mathbf{u}_{m,\mu} \in \mathbb{C}^K$,

$$\mathbf{u}_{m,\mu} = \mathbf{Q}_{m,\mu} \mathbf{a}_m \quad (15)$$

where

$$\mathbf{Q}_{m,\mu} = \text{diag}\{q_{m,\mu}[0] \dots q_{m,\mu}[k] \dots q_{m,\mu}[K-1]\} \quad (16)$$

$$\mathbf{a}_m^T = [e^{j\varphi_{m,0}}, \dots, e^{j\varphi_{m,k}}, \dots, e^{j\varphi_{m,K-1}}] \quad (17)$$

and $q_{m,\mu}[k]$ is the μ th component of $\mathbf{q}_{m,k}$. With these quantities, the smoothness of the interpolation of the m th eigenvector — and thus the smoothness of the alignment in phase in the different bins — can be assessed by a weighted inner product which measures the power in a p th derivative of the interpolation via a weighting matrix \mathbf{C}_p defined in [11], [16]

$$\chi_m = \sum_{\mu=1}^M \mathbf{u}_{m,\mu}^H \mathbf{C}_p \mathbf{u}_{m,\mu} \quad (18)$$

$$= \mathbf{a}_m^H \sum_{\mu=1}^M \mathbf{Q}_{m,\mu}^H \mathbf{C}_p \mathbf{Q}_{m,\mu} \mathbf{a}_m = \mathbf{a}_m^H \mathbf{D}_{p,m} \mathbf{a}_m. \quad (19)$$

The substitution $\mathbf{D}_{p,m} = \sum_{\mu=1}^M \mathbf{Q}_{m,\mu}^H \mathbf{C}_p \mathbf{Q}_{m,\mu}$ permits to express this smoothness metric directly in terms of the vector of phase shifts \mathbf{a}_m for the m th eigenvector.

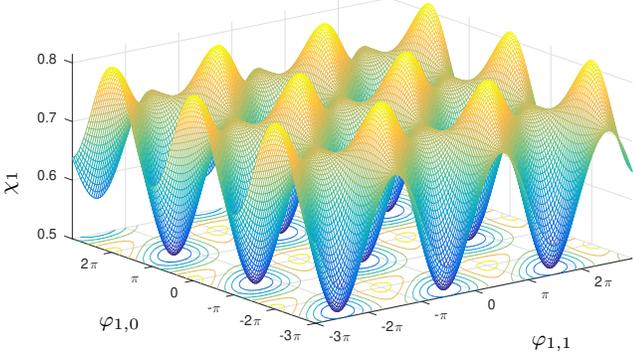


Fig. 1. Cost function value χ_1 in dependency of $\varphi_{1,0}$ and $\varphi_{1,1}$.

B. Gradient-Based Iterative Optimisation

The minimisation of χ_m is accomplished by appropriately tuning the phase in every frequency bin, assembled into a vector $\boldsymbol{\varphi}_m = [\varphi_{m,0}, \dots, \varphi_{m,K-1}]^T \in \mathbb{R}^K$,

$$\boldsymbol{\varphi}_{m,\text{opt}} = \arg \min_{\boldsymbol{\varphi}_m} \mathbf{a}_m^H \mathbf{D}_{p,m} \mathbf{a}_m = \arg \min_{\boldsymbol{\varphi}_m} \chi_m. \quad (20)$$

An example of the cost function χ_1 for $M = 2$ and

$$[\mathbf{q}_1(z) \ \mathbf{q}_2(z)] = \frac{1}{2} \begin{bmatrix} 1 + z^{-1} & 1 - z^{-1} \\ 1 - z^{-1} & 1 + z^{-1} \end{bmatrix} \quad (21)$$

is shown in Fig. 1 in dependency on phase shifts in the two frequency bins for $K = 2$. Since this cost function appears difficult to solve in closed form, we attempt an iterative gradient approach, such that $\boldsymbol{\varphi}_m$ depends on an iteration index n , and

$$\boldsymbol{\varphi}_m[n+1] = \boldsymbol{\varphi}_m[n] - \mu \frac{\partial \chi_m}{\partial \boldsymbol{\varphi}_m}. \quad (22)$$

Since $a_m[k] = e^{j\varphi_{m,k}}$, we can define

$$\frac{\partial}{\partial \boldsymbol{\varphi}_m} = \frac{\partial \mathbf{a}_m^H}{\partial \boldsymbol{\varphi}_m} \frac{\partial}{\partial \mathbf{a}_m^*} + \frac{\partial \mathbf{a}_m^T}{\partial \boldsymbol{\varphi}_m} \frac{\partial}{\partial \mathbf{a}_m} \quad (23)$$

$$= -j\mathbf{\Phi}_m^* \frac{\partial}{\partial \mathbf{a}_m^*} + j\mathbf{\Phi}_m \frac{\partial}{\partial \mathbf{a}_m}, \quad (24)$$

with

$$\mathbf{\Phi}_m = \text{diag}\{e^{j\varphi_{m,0}}, \dots, e^{j\varphi_{m,L-1}}\}. \quad (25)$$

Note that $\mathbf{\Phi}_m \mathbf{1} = \mathbf{a}_m$. Therefore

$$\frac{\partial \chi_m}{\partial \boldsymbol{\varphi}_m} = -j\mathbf{\Phi}_m^* \mathbf{D}_{p,m} \mathbf{a}_m + j\mathbf{\Phi}_m \mathbf{D}_{p,m}^T \mathbf{a}_m^* \quad (26)$$

$$= 2\text{Im}\{\mathbf{\Phi}_m^* \mathbf{D}_{p,m} \mathbf{\Phi}_m \mathbf{1}\}. \quad (27)$$

This defines the iterative update scheme.

C. Fast Newton Approach

The convergence of the iterative approach in (22) can be enhanced using a fast Newton approach [17] based on

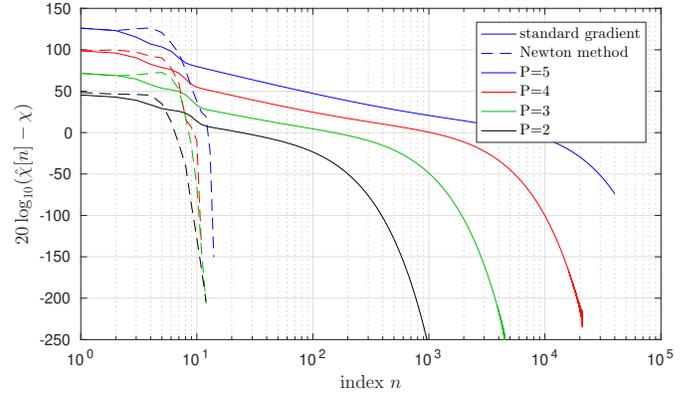


Fig. 2. Convergence curves for phase adaptation with different power of derivatives considered.

the Hessian matrix $\mathbf{H} = 2\text{Re}\{\mathbf{\Phi}_m^* \mathbf{D} \mathbf{\Phi}_m\}$ derived similarly to [10]. The resulting Newton update \mathbf{H}^{-1} , such that

$$\boldsymbol{\varphi}_m[n+1] = \boldsymbol{\varphi}_m[n] - \mu \mathbf{H}^{-1} \frac{\partial \chi_m}{\partial \boldsymbol{\varphi}_m}. \quad (28)$$

Example. For a $K = 64$ -point DFT of $\mathbf{q}_1[n] \circ \bullet \mathbf{q}_1(z)$ in (21), the phase is altered randomly. Thereafter, the above iterative updating schemes are used to calculate the phase. The results against the optimum attainable smoothness metric of $\chi = \frac{1}{2}$ for an undisturbed phase are shown in Fig. 2. The iterative gradient algorithm in (22) converges slowly, particularly if the power in the derivatives are taken at higher-order, such as $p = 5$, with faster adaptation for lower p . In contrast, the fast Newton update significantly reduces the number of iterations, reaching machine precision after around 12 iterations as seen in Fig. 2.

While the update (28) can converge faster when the term \mathbf{H}^{-1} is included, it may be attracted to critical points [13]. From Fig. 1 we know that the cost function can include saddle points and local minima. This can be overcome by a modulation applied to $\mathbf{\Phi}_m$ after initial convergence, and re-checking whether an enhanced solution is attained. For lack of space, this is not further elaborated here.

V. APPROXIMATION ORDER AND APPROXIMATION ERROR

Any approximation error will make itself felt by a lack of paraunitarity. While the eigenvectors $\mathbf{q}_m(e^{j\Omega})$, $m = 1 \dots M$, are orthogonal at the sample points $\Omega = \Omega_k = 2\pi k/K$, $k = 0 \dots (K-1)$, and therefore $\mathbf{Q}_k = [\mathbf{q}_1(e^{j\Omega_k}), \dots, \mathbf{q}_M(e^{j\Omega_k})]$ unitary, the continuous trigonometric or Dirichlet interpolations $\hat{\mathbf{Q}}(e^{j\Omega})$ through \mathbf{Q}_k , $k = 0, \dots, (K-1)$ may not satisfy unitarity. The interpolation $\hat{\mathbf{Q}}(e^{j\Omega})$ will only be unitary for all $\Omega \in \mathbb{R}$ if the arbitrary phase at every frequency value does not affect orthogonality. We can measure therefore

$$\epsilon_K = \frac{1}{2\pi} \int_{-\pi}^{\pi} \|\hat{\mathbf{Q}}(e^{j\Omega}) \hat{\mathbf{Q}}^H(e^{j\Omega}) - \mathbf{I}\|_F^2 d\Omega, \quad (29)$$

where ϵ_K is the error linked to the approximation order K .

To evaluate ϵ_K , we expand (29), to obtain the trigonometric interpolation $\hat{q}_\mu(e^{j\Omega})$ of $q_m(e^{j\Omega_k})$,

$$\epsilon_K = \frac{1}{2\pi} \int_{-\pi}^{\pi} \sum_{m=1}^M \left(\sum_{\mu=1}^M |\hat{q}_m^H(e^{j\Omega}) \hat{q}_\mu(e^{j\Omega})|^2 - 2\hat{q}_m^H(e^{j\Omega}) \hat{q}_m(e^{j\Omega}) + 1 \right) d\Omega. \quad (30)$$

For the second term in (30) we have based on Parseval

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} \sum_{m=1}^M \hat{q}_m^H(e^{j\Omega}) \hat{q}_m(e^{j\Omega}) d\Omega = \frac{1}{K} \sum_{k=0}^{K-1} \|\mathbf{Q}_k\|_F^2 = M.$$

For the first term in (30),

$$\begin{aligned} \frac{1}{2\pi} \int_{-\pi}^{\pi} \sum_{m=1}^M \left(\sum_{\mu=1}^M |\hat{q}_m^H(e^{j\Omega}) \hat{q}_\mu(e^{j\Omega})|^2 \right) d\Omega &= \\ &= \frac{1}{2K} \sum_{\kappa=0}^{2K-1} \|\hat{\mathbf{O}}_\kappa^H \hat{\mathbf{O}}_\kappa\|_F^2, \end{aligned} \quad (31)$$

where $\hat{\mathbf{O}}_\kappa$ is a twofold oversampled version of \mathbf{Q}_k , with a trigonometric interpolation such that for $\kappa \in \mathbb{Z}$

$$\hat{\mathbf{O}}_{2\kappa} = \mathbf{Q}_\kappa \quad \text{and} \quad \hat{\mathbf{O}}_{2\kappa+1} = \sum_{k=0}^{K-1} \alpha_k \mathbf{Q}_{\kappa+k+1}.$$

Working with sample points on the unit circle implies a cyclic wrap-around of the coefficients \mathbf{Q}_k for $k \geq K$. It can be shown that the interpolation filter possesses the coefficients of a Dirichlet kernel for even K [18].

Since the \mathbf{Q}_k are unitary, $\|\mathbf{Q}_k\|_F^2 = M$. Therefore, (31) can be split into two sums, one for odd and one for even values of κ , and the cost ϵ_K simplifies to

$$\epsilon_K = \frac{1}{K} \sum_{\kappa=0}^{K-1} \left\| \sum_{k=0}^K \alpha_k \mathbf{Q}_{\kappa+k+1} \right\|_F^2 - M. \quad (32)$$

Thus, the error in paraunitarity can be precisely measured from a simple two-fold trigonometric interpolation of the sample points, and if required, K can be doubled with a repeat of $P1$ and $P2$ until a desired accuracy has been reached.

VI. SIMULATION AND RESULTS

As an example to simulate the proposed approach, we build $\mathbf{R}(z)$ from the example in [2] with $\mathbf{\Lambda}(z) = \text{diag}\{z+3+z^{-1}, jz+3-jz^{-1}\}$. With $q_1(z)$ and $q_2(z)$ from (21), we perturb a sampled version of $\mathbf{Q}(e^{j\Omega})$ for $K=8$ via

$$\mathbf{Q}_k = \mathbf{Q}(e^{j\Omega_k}) \text{diag}\{e^{j\psi_{1,k}}, e^{j\psi_{2,k}}\} \mathbf{V}_k \quad (33)$$

with random, uniformly distributed angles $\psi_{m,k}$, $m=1,2$, $k=0, \dots, (K-1)$. Initially, $\mathbf{V}_k = \mathbf{I} \forall k$. For an ensemble of 1000 randomisations of $\{\psi_{m,k}\}$, we recover $\hat{\mathbf{Q}}(z)$ with an average $\epsilon_K = 2.54e-16 \pm 2.67e-15$ using the proposed approach or equivalently the methods in [6], [10].

If additionally a random unitary subspace perturbation \mathbf{V}_k occurs at the algebraic multiplicities of $\Omega_1 = \frac{\pi}{4}$ and $\Omega_5 = \frac{\pi}{5}$, then [6] fails to recover eigenvalues and eigenvectors independent of how large K is selected. In the proposed approach,

the eigenvalues can be recovered using [8], [9], and with $K=8$ and $\Delta = 2^{-8} \frac{2\pi}{K}$ in $P1$, the proposed method finds the correct eigenvectors with $\epsilon_K = 5.78e-13 \pm 1.83e-11$. Because the ground truth has finite order, no iteration via $P3$ is required.

VII. CONCLUSION

This paper has targetted a method for extracting approximations of analytic eigenvectors from analytic parahermitian matrices. Operating in the DFT domain, this requires (i) to re-establish coherence and find the correct association across frequency bins. This is a byproduct of the previous extraction of analytic eigenvalues in [8], [9], which is robust w.r.t. algebraic multiplicities of eigenvalues greater than one. Thus, the proposed approach concentrates on (ii) subspace and (iii) phase alignment of eigenvectors across bins. Particularly (ii) adds robustness over existing approaches in [6], [10]. Additionally, a simple error metric in the DFT domain allows us to iterate the approach by increasing the DFT length until a sufficient accuracy has been reached.

REFERENCES

- [1] J.G. McWhirter, P. D. Baxter, T. Cooper, S. Redif, and J. Foster, "An EVD Algorithm for Para-Hermitian Polynomial Matrices," *IEEE Trans. SP*, **55**(5):2158–2169, May 2007.
- [2] S. Weiss, J. Pestana, and I.K. Proudler, "On the existence and uniqueness of the eigenvalue decomposition of a parahermitian matrix," *IEEE Trans. SP*, **66**(10):2659–2672, May 2018.
- [3] S. Weiss, J. Pestana, I.K. Proudler, and F.K. Coutts, "Corrections to 'on the existence and uniqueness of the eigenvalue decomposition of a parahermitian matrix'," *IEEE Trans. SP*, **66**(23):6325–6327, Dec. 2018.
- [4] S. Redif, S. Weiss, and J.G. McWhirter, "Sequential matrix diagonalization algorithms for polynomial EVD of parahermitian matrices," *IEEE Trans. SP*, **63**(1):81–89, Jan. 2015.
- [5] J. G. McWhirter and Z. Wang, "A novel insight to the SBR2 algorithm for diagonalising para-hermitian matrices," in *11th IMA Maths in Signal Proc.*, Birmingham, UK, Dec. 2016.
- [6] M. Tohidian, H. Amindavar, and A. M. Reza, "A DFT-based approximate eigenvalue and singular value decomposition of polynomial matrices," *EURASIP J. Advances in Signal Proc.*, **2013**(1):1–16, 2013.
- [7] C. Delaosa, J. Pestana, N.J. Goddard, S. Somasundaram, and S. Weiss, "Sample space-time covariance matrix estimation," in *IEEE ICASSP*, pp. 8033–8037, Brighton, UK, May 2019.
- [8] S. Weiss, I.K. Proudler, F.K. Coutts, and J. Pestana, "Iterative approximation of analytic eigenvalues of a parahermitian matrix EVD," in *IEEE ICASSP*, Brighton, UK, May 2019.
- [9] S. Weiss, I.K. Proudler, and F.K. Coutts, "Eigenvalue decomposition of a parahermitian matrix: extraction of analytic eigenvalues," *IEEE Trans. SP*, submitted, 2020.
- [10] F.K. Coutts, K. Thompson, J. Pestana, I.K. Proudler, and S. Weiss, "Enforcing eigenvector smoothness for a compact DFT-based polynomial eigenvalue decomposition," in *IEEE SAM*, Sheffield, UK, July 2018.
- [11] S. Weiss, J. Selva, and M. Macleod, "Measuring smoothness of trigonometric interpolation through incomplete sample points," in *EUSIPCO*, Amsterdam, The Netherlands, Jan. 2021.
- [12] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, Maryland: John Hopkins University Press, 1996.
- [13] J. H. Manton, "Optimization algorithms exploiting unitary constraints," *IEEE Trans. SP*, **50**(3):635–650, Mar. 2002.
- [14] A. Hjørungnes and D. Gesbert, "Complex-valued matrix differentiation: Techniques and key results," *IEEE TSP*, **55**(6):2740–2746, June 2007.
- [15] W. Wirtinger, "Zur formalen Theorie der Funktionen von mehr komplexen Veränderlichen," *Mathematische Annalen*, **97**:357–376, 1926.
- [16] S. Weiss and M. D. Macleod, "Maximally smooth dirichlet interpolation from complete and incomplete sample points on the unit circle," in *IEEE ICASSP*, Brighton, UK, May 2019.
- [17] S. Haykin, *Adaptive Filter Theory*, 2nd ed, Prentice Hall, 1991.
- [18] A. Zygmund, *Trigonometric Series*, 2nd ed, Cambridge Uni. Press, 1959.

Sensing and Automation in the Future Maritime Environment

Daniel D. Sternlicht
 Naval Surface Warfare Center, Panama City Division
 110 Vernon Avenue
 Panama City, FL, USA
 daniel.sternlicht@navy.mil

Abstract— In this emerging era of great power competition, the goal of outpacing potential adversaries in the development of military technology takes on a new urgency. Evolving capabilities in sensing and automation are driven by a trade space that includes range and lethality versus close engagement and survivability; finders versus hidiers; centralized command/control versus asset independence/dispersion; and planning and judgement versus reaction and autonomy. This paper explores this trade space first by describing sensing and automation innovations demonstrated during the 1990/1991 Gulf War and shortly thereafter, followed by discussion on current and emerging game-changing technologies. Capabilities projected for near and far term advantage include: weapons systems ensuring long-range lethality; unmanned cooperative networks of offboard systems; artificial intelligence and machine learning; and exploitation of advanced materials and quantum technologies. These will play a vital role in realizing a networked force of manned and unmanned systems with the ability to sense, comprehend, communicate, predict, plan, and take appropriate action in the future maritime environment.

Keywords—Sensors, Surveillance, Reconnaissance, Navigation, Communications, Automation, Autonomy, Data Analytics, Artificial Intelligence, Machine Learning, Advanced Materials, Quantum Technologies, Gulf War.

I. INTRODUCTION

In this emerging era of great power competition, the goal of outpacing potential adversaries in the development of military technology takes on a new urgency. As emphasized by the late Shawn Brimley in his strategic analysis “While We Can,” recent history teaches that each lead in technological capability is fleeting at best. The U.S. experienced roughly a decade of advantage during the early nuclear era, and arguably, several decades during the evolution of precision-guided weapons – what defense analysts refer to as the first and second offset strategies. Today’s trends are not conducive to the preservation of the decisive, undisputed edge historically experienced, as the rapid proliferation of technology in the global marketplace will continue to compress the time during which any new military technology provides advantage. It is not unreasonable to assume that the emergence of any new disruptive military technology will be matched within a decade [1,2].

Evolving capabilities in sensing and automation are driven by a trade space that includes range and lethality versus close

engagement and survivability; finders versus hidiers; centralized command/control versus asset independence/dispersion; and planning and judgement versus reaction and autonomy. This paper explores this trade space by describing sensing and automation innovations demonstrated during the 1990/1991 Gulf War – and how they impacted and evolved for subsequent conflicts (Section II). This follows with a description of current and emerging game-changing technologies forecast to impact the future maritime battlespace (Section III). The final section summarizes the significance of these developments toward a third offset strategy for maritime superiority.

II. EARLY 21ST CENTURY BATTLESPACE

Following on the heels of the Soviet Block’s collapse, the 1990/1991 Gulf War can be considered the first engagement of the 21st Century Battlespace. While one-sided, involving a U.S.-led international coalition crossing swords with a technologically less sophisticated regional power, it was the largest-scale U.S. war since Vietnam, and provided lessons learned in weapons, sensing and automation that would resonate across the next 30 years.

Perhaps the most memorable event was the lightning advance of U.S. and Coalition armored divisions across the Arabian Desert during Operation Desert Storm. This was, of course, the terminal phase of a protracted, often times “hot,” conflict, with joint Army, Air Force, Navy and Marine Corps forces cooperating in land and sea operations. From a sensing and automation perspective, there are five areas that stand out in their impact on the war: Smart Weapons; Electromagnetic Warfare; Multi-Sensor Fusion; Mine Warfare; and Unmanned Systems. Table 1 collates some of these notable capabilities and systems (and shortfalls), as well as their implications for future conflicts, which include: the 1990s Balkan Wars; the 2001 Afghanistan War/Operation Enduring Freedom; the 2003 Iraq War; and the Libyan and Syrian Civil Wars of the 2010s [3-11].

A. Smart Weapons

Though “smart weapons” have been around since the mid-20th century, with the advent of radio-guided bombs and sonar-homing torpedoes in WWII, a new generation of precision-guided weapons played a significant role in the Gulf War and the conflicts that would follow. These include laser-

guided bombs that detect illumination from an aimed laser to guide a munition to that target. Tomahawk Land Attack Cruise Missiles (TLAM), launched from naval vessels, could autonomously navigate long distances for precision strike, via a low-altitude, radar-avoiding flight path. This was made possible through onboard navigation systems combining altimeters for Terrain Contour Mapping (TERCOM) and imaging sensors with Digital Scene Mapping Area Correlation (DSMAC) capabilities. Throughout the conflict, the threat from Iraqi missiles were significantly reduced through the use of High-speed Anti-Radiation Missiles (HARM) that homed in on a missile battery's radar transmitter. The utility of these smart weapons exceeded expectations and, following the Gulf War, larger inventories were procured. The incorporation of, and advances in, inertial navigation, Global Position System (GPS) receivers, mapping/scene correlators, and sensors for terminal homing, would make these weapons more abundant and more lethal in future conflicts.

B. Electromagnetic Warfare

Electronic warfare was not new; however, control of the electromagnetic spectrum during the Gulf War played a vital role in ensuring Coalition command and control of the battlespace/communications, and superiority in gathering intelligence and conducting surveillance/reconnaissance – a composite capability typically abbreviated C3ISR. From the onset of the Coalition buildup (Operation Desert Shield) following the August 1990 Iraqi invasion of Kuwait, to the commencement of Desert Storm hostilities in January 1991, satellites provided high-fidelity terrain maps that were used by TLAMs for their eventual precision strike operations. Target detection radars, synthetic aperture (active and inverse) radars, and infrared imagers were used on a variety of platforms so that Coalition assets could detect and target Iraqi weapons systems, before they themselves could be detected. Electromagnetic jamming was used by Coalition aircraft to blind Iraqi radars. Airspace management was greatly facilitated by Air Force/Navy Airborne Warning and Control Systems (AWACS) and Identify Friend or Foe (IFF) interrogators, while ground target battle management was largely orchestrated by the Army/Air Force Joint Surveillance and Target Attack Radar Systems (JSTARS). Lessons learned after the war would lead to expansion of satellite remote sensing and communications capabilities, improvements in all-weather sensors, and universal adoption of IFF capabilities across the services.

C. Multi-Sensor Fusion

The utility of fusing multiple sensors and sensor types to gain tactical understanding of the battlespace, and for calculating targeting solutions, was manifest by the large variety of sensing modalities used during the Gulf War. Putting this into practice would be combat pilots and air traffic operators, as well as ship-tracking data fusion centers at Central Command (CENTCOM) and stateside. During the war the Aegis Combat System (ACS), integrated on Ticonderoga-class guided-missile cruisers and representing the most sophisticated afloat system of its type, protected Coalition naval forces and conducted strikes using computer-controlled anti-air warfare and missile launch systems. The ACS would expand to additional platforms, and continue to evolve with

integration of TLAMs, advanced electronic support and counter measures, helicopter-borne sensors, and sophisticated Anti-Submarine Warfare (ASW) sonars. The Gulf War demonstrated the need for timely assimilation of multiple sources of data and intelligence. This, and improvements in faster/distributed computing, would lead to developments of network-centric warfare capabilities for threat assessment, strike planning, fire support, and battle damage assessment.

D. Mine Warfare

During the Gulf War Coalition forces controlled the sea lines of communication through air, surface and undersea superiority. This was not, however, the case for the seaward approaches to the Kuwait shoreline. Iraqi minefields laid in the Northern Persian Gulf – some by naval forces, others by clandestine means – tied up American and British mine countermeasures (MCM) vessels employing shipborne/airborne mine-hunting and mine-sweeping gear. This compromised allied abilities to bring expeditionary forces ashore, hindered naval gunfire support, and resulted in mine-strikes on two U.S. warships. Eleven nations would participate in a year of postwar cleanup that cleared over 1200 sea mines. The Gulf War was a wake up call that led to emphasis on mine-laying reconnaissance (“left of splash”) and advances in faster, unmanned, and platform interoperable (i.e. “organic”) MCM gear. Minefield reconnaissance would be effectively used to interdict clandestine mine laying operations during the 2003 Iraq War. This conflict would also see the first involvement of unmanned undersea vehicles (UUV) in a combat environment, as they were used for mine and harbor clearance following the Battle of Umm Qasr.

E. Unmanned Systems

The use of military drones was not new – radio-controlled planes were flown for special missions in WWII, and “drone boats” were used for riverine reconnaissance during the Vietnam War. The Gulf War, however, provided an *in situ* laboratory for developing newfound capabilities in unmanned systems: using advances in electronics, computer hardware/software, and small vehicle technology. This would foster continued improvements in machine sensing, perception, and autonomy, leading to expanded use of these systems in future conflicts and development of new concepts for unmanned warfare. During the Gulf War unmanned aerial vehicles (UAV) were used for targeting, battle damage assessment, and as decoys. In the time since: multiple classes of UAVs have been developed for surveillance, reconnaissance, targeting, and as weapons platforms; multiple classes of UUVs have been fielded and are under development for MCM, intelligence preparation of the operational environment, and for long-endurance surveillance/reconnaissance; and multiple classes of Unmanned Surface Vehicles (USV) have been fielded and are under development for MCM, patrol duties, surveillance, and for logistics support.

F. Information Control

Regardless of operational scale, the 1990/1991 Gulf War and follow-on wars were conflicts marked by an imbalance of military capability. Studying these engagements provides solid clues regarding the future of warfare – but also false leads,

especially for large-scale conflicts involving high-capacity, technologically sophisticated powers on each side. From a sensing and automation perspective, what is sure is that robust capabilities in command and control, communications, navigation, intelligence, surveillance, reconnaissance, targeting, and ultimately the control of information, will be key to holding a force together and for controlling the battlespace.

III. GAMECHANGERS: NEAR & FAR

Today’s trends are not conducive to the preservation of that decisive, undisputed edge experienced during the Gulf War and early 21st Century conflicts, as the rapid proliferation of technology in the global marketplace compresses the time during which any new military technology provides advantage. For example, with widespread use of GPS and the emergence of competing global navigation satellite systems, less affluent nations can build smart weapons – and (though more

expensive) alternative methods of guidance are being incorporated for when satellite networks are compromised.

In “The Operational Environment, 2035-2050: The Emerging Character of Warfare,” the U.S. Army Training and Doctrine Command (TRADOC) categorizes the development of future weapons systems, many with relevance to the undersea domain, for example, range and lethality vs. close engagement and survivability; finders vs. hiders; centralized command/control vs. asset independence/dispersion; planning and judgement vs reaction and autonomy [12,13]. These areas and their implications for future technology development are explored in the sections that follow [14-26].

A. Range-Lethality and Finders vs. Hidiers

Active sensors, with the advantage that comes with illuminating targets but the drawback of revealing the emitter’s position, will continue to have utility, depending on the degree of battlespace control and adversarial sophistication.

TABLE I. GULF WAR (1990/1991) ADVANCES IN SENSING & AUTOMATION, AND IMPLICATIONS FOR FUTURE CONFLICTS

Gulf War (1990/1991) Standouts	Implications for Iraq War (2003) & Other Conflicts
<p>Smart Weapons</p> <ul style="list-style-type: none"> Laser Guided Bombs <ul style="list-style-type: none"> Guided Bomb Unit: GBU-12 Ship-launched Tomahawk Land Attack Cruise Missiles (TLAM) <ul style="list-style-type: none"> Terrain Contour Mapping (TERCOM) Digital Scene Mapping Area Correlation (DSMAC) High-Speed Anti-Radiation Missiles (HARM) <ul style="list-style-type: none"> Radar detection and homing 	<p>Smart Weapons</p> <ul style="list-style-type: none"> Insufficient inventory during Gulf War led to future emphasis on laser-guided bombs and additional targeting/navigation modalities; Joint Direct Attack Munition (JDAM) Addition of GPS- and DSMAC- aided inertial navigation for greater accuracy
<p>Electromagnetic Warfare</p> <p>Intelligence, Surveillance, Reconnaissance, Targeting</p> <ul style="list-style-type: none"> Satellite mapping and communications Target detection and synthetic aperture radar (SAR) <ul style="list-style-type: none"> LAMPS (Light Airborne Multi-Purpose System) on Navy MH-60 Helicopter Forward Look Infrared (FLIR) for night-time and heat sensing <ul style="list-style-type: none"> Navy A-6 Intruder aircraft LANTIRN (Low Altitude Navigation and Targeting Infrared Night) on Air Force F-16 Jet Fighter <p>Radar Jamming</p> <ul style="list-style-type: none"> Navy EA-6B Prowler aircraft electronic countermeasures <p>Airborne/Ground Warning and Traffic Control</p> <ul style="list-style-type: none"> Identification Friend or Foe (IFF) Interrogators Navy E-2/Air Force E-3 Airborne Warning & Control Systems (AWACS) with aircraft tracking radar, passive sensing, and IFF Amy/Air Force E-8 aircraft Joint Surveillance and Target Attack Radar System (JSTARS) with SAR for ground target battle management 	<p>Electromagnetic Warfare</p> <ul style="list-style-type: none"> Gulf War difficulty in targeting mobile SCUD missile launchers spurred Improvements in all-weather sensors Expansion of satellite remote sensing and communications capabilities Gulf War disparity in IFF capabilities across services led to universal adoption
<p>Multi-Sensor Fusion</p> <ul style="list-style-type: none"> Navy Ticonderoga-class Aegis Guided Missile Cruisers – computer-controlled Anti-Air Warfare and missile launch systems Ship tracking data fusion centers at Central Command (CENTCOM) and within Continental US 	<p>Multi-Sensor Fusion</p> <ul style="list-style-type: none"> Need for real-time integration of Command, Control, Communications, Computers, Intelligence, Surveillance, Reconnaissance (C4ISR) led to development of Network-centric warfare capabilities for strike planning, fire support, and BDA Aegis Combat System evolves with integration of TLAM; SLQ-32 Radar with Electronic Support and Counter Measures (ESM/ECM); LAMPS with radar and Anti-submarine Warfare (ASW) sonobuoys; and ASW SQQ-89 bow sonar/towed array
<p>Naval Mine Countermeasures (MCM) Systems</p> <ul style="list-style-type: none"> U.S. Ocean Class (MSO) Mine Sweepers; MCM-class ships with SQQ-32 mine hunting sonar; MH-53 Helicopter Mine Sweepers; U.K. Hunt-class MCM vessels 	<p>Naval Mine Countermeasures (MCM) Systems</p> <ul style="list-style-type: none"> Insufficient mine-laying reconnaissance and slow pace of mine hunting/sweeping limited Gulf War expeditionary operations and naval fire support (Post-war clearance of over 1200 mines by international team), and spurred development of new MCM technologies with emphasis on unmanned systems The 2003 Iraq War saw first use of Unmanned Undersea Vehicles (UUV) in a combat environment, as they were used for mine and harbor clearance following the Battle of Umm Qasr
<p>Unmanned Systems</p> <ul style="list-style-type: none"> Unmanned Aerial Vehicles (UAV) used for targeting, battle damage assessment (BDA), and as decoys 	<p>Unmanned Systems</p> <p>Post-Gulf War:</p> <ul style="list-style-type: none"> Multiple classes of UAVs developed for surveillance, targeting, and weapons platforms Multiple classes of UUVs fielded and under development for MCM, light-weight rapid deployment, and long-endurance surveillance/reconnaissance Multiple classes of Unmanned Surface Vehicles (USV) fielded and under development for MCM, patrol, surveillance, and logistics

An exception to the range-lethality paradigm is the ability to employ survivable (hardened, concealable) or expendable assets that operate forward and in contested environments without requiring full control of the physical space or sensing/communications spectrum. With respect to finders vs. hidiers, in future land and air engagements, concealment will be very difficult, although that advantage will likely be maintained for the mid-term in the undersea domain – given the cost of operating in harsh conditions, opacity to electromagnetic radiation, and challenges to acoustic reconnaissance/surveillance that include high clutter and propagation ambiguity. Eventually, for offboard platforms to remain concealed, passive-only sensing, combined with artificial intelligence-based autonomy and information processing, will be key enablers.

Persistent Littoral Undersea Surveillance

An example of advances in distributed autonomy is the U.S. Office of Naval Research (ONR) initiative of the 2000s: Persistent Littoral Undersea Surveillance network (PLUSNet), as described in the monograph “Naval Innovation for the 21st Century.” This initiative developed and demonstrated the concept of using inexpensive undersea unmanned systems that were quiet, perceptive (with sensors), and had long endurance. Autonomous Ocean Gliders were the robotic systems of choice, as they can transit for months using minimal power (gliding forward through the water column by regulating changes in buoyancy) – using the reserve for: instruments that measure ocean quantities influencing sonar propagation; passive sonars (vector sensors) that measure both target-radiated signal strength and direction; and Iridium satellite-based communication links. This initiative, along with ONR demonstrations of unmanned undersea mine-hunting systems of the decade prior, would influence emerging approaches to naval undersea warfare worldwide. A legacy of this research is the standing up of a Glider Operations Center at the U.S. Navy Oceanographic Office.

B. Cooperative Offboard Networks

Cooperative networks of offboard systems will be essential to future battlespace operations – and this is where the final two TRADOC categories come into play. The trade space between maintaining control with full communications, and accepting the risk of acting without (or with limited) communications, will evolve in response to inherent challenges of operating in specific domains. In coming decades, networks of unmanned vehicles will be introduced in the ground, air, and surface domains. These will likely see their effectiveness and efficiency (i.e., ability to produce intended effects at an acceptable expenditure of resources) maximized when they can be used *en masse*, employing low-cost expendable units that are self-organized into *swarms*.

Centralized control of these networks in the challenging maritime environment will rely heavily on improvements in communications range, resilience, and bandwidth; while game-changing modes of independent operation will be enabled by advances in navigation, perception, and autonomy. Until high-

bandwidth communications are realized, this likely will evolve more slowly in the undersea domain. Pioneering systems, e.g. legacies of PLUSnet, will alternate between connected/centralized and disconnected/decentralized operational modes – a developmental stage that can be termed *cooperative autonomy*, as a prelude to true swarming capability.

As indicated in the previous subsection, feedback from cooperative systems (e.g. gliders) will facilitate composition of a common undersea picture that includes surveillance (detection, tracking, and targeting) and environmental characterization. Accurate characterization of the environment is key to optimizing sensor capabilities, and to setting bounds on how well these systems are expected to perform. The revolution in artificial intelligence that will facilitate operation of undersea networks, is already affecting sensing capabilities – with machine learning techniques (ML) being combined with physics-based models and expanding data sets to characterize sound speed structure and propagation (Figure 1), as well as the development of automated target recognition for sonar and other sensors.

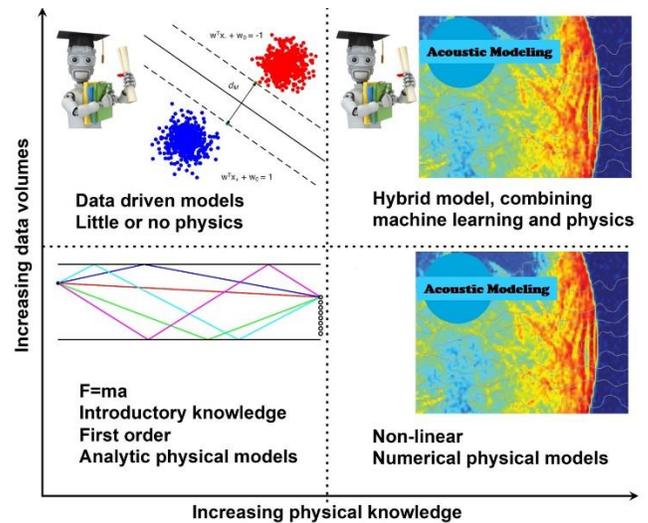


Fig. 1. Acoustic insight can be improved by leveraging the strengths of both physical and ML-based, data-driven models. Analytic physical models (lower left) give basic insights about physical systems. More sophisticated models, reliant on computational methods (lower right), can model more complex phenomena. Whereas physical models are reliant on rules, which are updated by physical evidence (data), ML is purely data-driven (upper left). By augmenting ML methods with physical models to obtain hybrid models (upper right), a synergy of the strengths of physical intuition and data-driven insights can be obtained. (From Bianco et al. [18], with permission.)

Ultimately, the control and effective use of these offboard systems will require the processing of large, incomplete, and rapidly changing stores of information, as well as the ability to automate key decision processes.

C. Advanced Materials & Quantum Technologies

Developments in new materials and quantum technologies are predicted to have game-changing impacts on the ability of manned and robotic systems to sense and interact with the

environment – for situational awareness, threat tracking and targeting, navigation, and communication. *Nanomaterials* are man-made substances crafted at the nanometer, molecular scale (e.g. a DNA strand is roughly two *nm* across), allowing them to support unique quantum and/or surface properties affecting optical, magnetic, and electrical behavior. These are some of the building blocks for quantum devices, as well as for the broad category of *metamaterials* (from the Greek work “Meta” meaning “beyond”) that exhibit properties not typically found in nature, and which are beginning to impact sensing, computing, and communications electronics worldwide.

The oft-hailed disruptive/revolutionary potential of *quantum technology* takes advantage of the paradoxical principles of quantum mechanics – such as computing bits (or “qubits” in quantum parlance) being in multiple states at once (i.e. *quantum superposition*) leading to exponential increases in processing efficiency over conventional computers; and particles (e.g. photons) sharing a single quantum state (i.e. *quantum entanglement*) such that a change in the state of one particle influences a change in another – even if they are separated by vast distance.

TABLE II. EMERGING CAPABILITIES FROM ADVANCED MATERIALS AND QUANTUM TECHNOLOGIES

Computing
Exponential increase of computing power for: AI/ML, Modeling & Simulation, Code breaking Nanoscale patterning of electronic circuits
Sensing and Communication
Piezoelectric and Piezoresistive Materials Quantum Illumination Radar for Low-SNR Detection Paint-on / Transparent / Flexible Antennas Optoelectronics Extremely Accurate Quantum Magnetometers Quantum Gravimeter Chemical and Biological Sensors Temperature and Humidity Sensors Communication Encryption via Quantum Key Distribution (QKD)
Positioning, Navigation, Timing (PNT)
High-Precision Quantum Atomic Clocks Quantum Compass Inertial Measurement Units
Stealth
Optical camouflage and acoustic absorbing material Refractive index engineering

Table 2 lists a selection of emerging capabilities from advanced materials and quantum technologies that are relevant to maritime sensing/automation; where, for example: Quantum illumination radars can be used to match quantum-entangled photons scattered from low cross-section targets to reference photons held at the receiver – more effectively distinguishing target backscatter from high levels of background noise;

Quantum magnetometers, gravimeters, and inertial measurement units can be combined to provide accurate estimates of vehicle position and motion-state (independent from external navigation signals); and light/sound refracting materials show promise of camouflaging objects constructed with them. A key feature for the majority of devices listed, is that they are high-precision, fine-resolution, and economic in size, weight and power. While some of these applications are more realized than others, their continued development over the next decade will produce new capabilities in sensing and automation that significantly impact the maritime battlespace.

IV. SUMMARY

This paper analyzed new sensing and automation capabilities demonstrated during the 1990/1991 Gulf War, and which evolved to influence subsequent military engagements. This includes developments in Smart Weapons, Electromagnetic Warfare, Multi-Sensor Fusion, Mine Warfare, and Unmanned Systems.

Competition to assure (or deny) maritime access in the 21st Century maritime battlespace will continue to accelerate as global and regional powers develop capabilities that include: long-range guided munitions and battle networks; advanced anti-ship and anti-air missiles; space, cyber and electronic countermeasures; and new undersea attack capabilities – capabilities that are facilitated by a global marketplace of rapid technology innovation.

The formulation of a “third offset strategy” for maritime superiority will be highly reliant on evolutionary and revolutionary progress in sensing and automation. Game-changing technologies projected for near and far term advantage include: weapons systems ensuring long-range lethality; unmanned cooperative networks of offboard systems; artificial intelligence and machine learning; and exploitation of advanced materials and quantum technologies. These will play a vital role in realizing a networked force of manned and unmanned systems with the ability to sense, comprehend, communicate, predict, plan, and take appropriate action in the future operational environment.

ACKNOWLEDGMENT

The author thanks Mr. Frank Downs, of the Naval Surface Warfare Center Panama City Division, for informing this paper with his expertise in advanced materials and quantum technologies.

REFERENCES

- [1] D. D. Sternlicht, “Maintaining Maritime Military Advantage,” Invited editorial for *Sea Technology*, vol. 60, no. 11, p. 7, 2019.
- [2] S. Brimley, “While We Can: Arresting the Erosion of America’s Military Edge,” Center for a New American Security, December 2015.
- [3] E. Cohen, “Gulf War Air Power Survey: Weapons, Tactics, and Training,” vol. IV, U.S. Air Force, U.S. Government Printing Office, 1993.
- [4] W. Rosenau, “Coalition Scud-Hunting in Iraq, 1991,” Chapter 3 of *Special Operations Forces and Elusive Enemy Ground Targets: Lessons from Vietnam and the Persian Gulf War*. Santa Monica, CA: Rand Corporation, 1994.

- [5] E. J. Marolda and R. J. Schneller. *Shield and Sword: The United States Navy and the Persian Gulf War*. Annapolis, MD: Naval Institute Press, 2001.
- [6] N. Friedman, "Naval Lessons of the Gulf War," in *Desert Shield Desert Storm: The 10th Anniversary of the Gulf War*. St. Petersburg, FL: Faircount, 2001.
- [7] M. E. O'Hanlon, "Operation Iraqi Freedom and the Future of the U.S. Military," Brookings Report, <https://www.brookings.edu/research/operation-iraqi-freedom-and-the-future-of-the-u-s-military/>, 2003.
- [8] N. Polmar, "U.S. Navy - The New Approach to Mine Warfare," *Proceedings of the U.S. Naval Institute*, vol. 147, no. 7, July 2015.
- [9] B. Knight, "Guide to Military Drones," Deutsche Welle (DW) Akademie, <https://www.dw.com/en/a-guide-to-military-drones/a-39441185>, June 2017.
- [10] Deputy Assistant Secretary of the Navy, "The Navy Unmanned Undersea Vehicle (UUV) Master Plan," November 9, 2004.
- [11] U.S. Department of Defense, "Unmanned Systems Integrated Roadmaps 2013-2038/2017-2042," 2013/2017.
- [12] U.S. Army Training and Doctrine Command, "The Operational Environment, 2035-2050: The Emerging Character of Warfare," 2017.
- [13] Department of the Navy, Research, Development, Test & Evaluation, "30 Year Research & Development Plan," 2017.
- [14] L. C. Z. Jobbagy, "The Efficiency Aspect of Military Effectiveness. *Militaire Spectator*," vol 178, no. 10, pp. 504-516, 2009.
- [15] S. J. Tangredi, *Anti-Access Warfare: Countering A2/AD Strategies*. Annapolis, MD: Naval Institute Press, 2013.
- [16] R. Buder, "Persistent Littoral Undersea Surveillance," Chapter 10 of *Naval Innovation for the 21st Century: The Office of Naval Research Since the End of the Cold War*. Annapolis, MD: Naval Institute Press, 2013.
- [17] B. Mensi, R. Rowe, S. Dees, D. Bryant, D. Jones and R. Carr, "Operational Glider Monitoring, Piloting, and Communications," *Proceedings of IEEE/OES Autonomous Underwater Vehicles (AUV)*, Oxford, MS, pp. 1-3, 2014.
- [18] M. J. Bianco, P. Gerstoft, J. Traer, E. Ozanish, M. A. Roch, S. Gannot, and C. Deledalle, "Machine Learning in Acoustics: Theory and Applications," *Journal of the Acoustical Society of America*, vol. 146, no. 5, November 2019.
- [19] O. D. Grace. *SONAR and Anti-Submarine Warfare*. Escondido, CA: Sage Publishing Co., 1999.
- [20] J. Stack, "Automation for Underwater Mine Recognition: Current Trends and Future Strategy," *SPIE: Detection and Sensing of Mines, Explosive Objects, and Obscured Targets*, vol. 8017, pp. 1-21, May 2011.
- [21] D. D. Sternlicht, M. P. Hayes, and R. E. Hansen. "Historical Development of Seabed Mapping Synthetic Aperture Sonar." *Proceedings of the Institute of Acoustics*, vol. 40, pt. 2, pp. 1-10, September 2018.
- [22] D. Vollath, *Nanomaterials: An Introduction to Synthesis, Properties and Applications*. Hoboken, NJ: John Wiley & Sons, 2013.
- [23] D. Sabba, *Metamaterials and their Applications*. Burlington, ON: Arcler Press, 2017.
- [24] C. L. Degen, F. Reinhard, P. Cappellero, "Quantum Sensing," *Reviews of Modern Physics*, vol. 89, no. 3, July-September 2017.
- [25] B. Balaji, "Quantum Radar: Snake Oil or Good Idea?," *International Carnahan Conference on Security Technology (ICCST)*, Montreal, QC, pp. 1-7, 2018.
- [26] M. Giles, "The US and China are in a Quantum Arms Race that will Transform Warfare," MIT Technology Review, <https://www.technologyreview.com/2019/01/03/137969/us-china-quantum-arms-race/>, January 3, 2019.

Author Index

A		L	
Altmann, Y.	31	Lane, R.O.	46
Apfeld, Sabine	21	Lay, Bunlong	71
Arakawa, Nobuya	61	Legros, Q.	31
Ascheid, Gerd	21	M	
B		McLaughlin, S.	31
Budik, Ondrej	66	McLaughlin, Stephen	26
Bukovsky, Ivo	66	Meignen, S.	31
Buller, Gerald S.	26	Mota, João F.C.	76
C		Mulgrew, Bernard	16
Charlish, Alexander	21,71	N	
Comblet, Fabrice	56	Noriyasu, Homma	66
Cornu, Cédric	56	P	
Coutts, Fraser K.	16,81	Pestana, Jennifer	1
D		Proudlar, Ian	6
Davies, Mike E.	6,31	Proudlar, Ian K.	1,11,81
Deeks, Julian	81	S	
Delaosa, Connor	1	Saito, Ryo	61
Dionelis, Nikolaos	41	Shukla, Vishwajeet	51
Dohnal, Gejza	66	Singour, Mayank	51
F		Stankovic, Vladimir	11
Foucault, Antoine	56	Steinbauer, Pavel	66
G		Sternlicht, Daniel D.	86
Gouk, Henry	36	Sun, Mengwei	6
H		T	
Halimi, Abderrahim	26	Thompson, John	16
Hamada, Shohei	61	Tsaftaris, Sotirios A.	41
Hopgood, James R.	6	W	
Hospedales, Timothy M.	36	Wallace, Andrew	26
I		Wallace, Andrew M.	76
Ichige, Koichi	61	Weiss, Stephan	1,11,81
Ichiji, Kei	66	Wu, Yun	76
K		Y	
Kashiwagi, Katsuhisa	61	Yaghoobi, Mehrdad	41
Khenchaf, Ali	56		