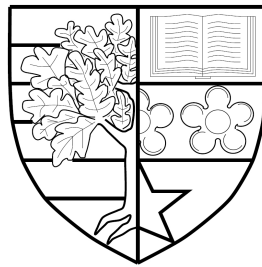


ANOMALY DETECTION & OBJECT CLASSIFICATION  
USING MULTI-SPECTRAL LIDAR AND SONAR

*by*  
PUNEET SINGH CHHABRA



Submitted for the degree of  
Doctor of Philosophy

INSTITUTE OF SENSORS, SIGNALS AND SYSTEMS  
SCHOOL OF ENGINEERING AND PHYSICAL SCIENCES  
HERIOT-WATT UNIVERSITY

February 2021

The copyright in this thesis is owned by the author. Any quotation from the report or use of any of the information contained in it must acknowledge this report as the source of the quotation or information.

## Research Thesis Submission

Please note this form should be bound into the submitted thesis.

Name:	Puneet Singh Chhabra		
School:	Engineering and Physical Sciences		
Version: <i>(i.e. First, Resubmission, Final)</i>	Final	Degree Sought:	PhD Electrical Engineering

### Declaration

In accordance with the appropriate regulations I hereby submit my thesis and I declare that:

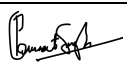
1. The thesis embodies the results of my own work and has been composed by myself
2. Where appropriate, I have made acknowledgement of the work of others
3. The thesis is the correct version for submission and is the same version as any electronic versions submitted\*.
4. My thesis for the award referred to, deposited in the Heriot-Watt University Library, should be made available for loan or photocopying and be available via the Institutional Repository, subject to such conditions as the Librarian may require
5. I understand that as a student of the University I am required to abide by the Regulations of the University and to conform to its discipline.
6. I confirm that the thesis has been verified against plagiarism via an approved plagiarism detection application e.g. Turnitin.

### ONLY for submissions including published works

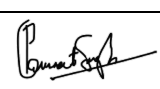
Please note you are only required to complete the Inclusion of Published Works Form (page 2) if your thesis contains published works)

7. Where the thesis contains published outputs under Regulation 6 (9.1.2) or Regulation 43 (9) these are accompanied by a critical review which accurately describes my contribution to the research and, for multi-author outputs, a signed declaration indicating the contribution of each author (complete)
8. Inclusion of published outputs under Regulation 6 (9.1.2) or Regulation 43 (9) shall not constitute plagiarism.

\* Please note that it is the responsibility of the candidate to ensure that the correct version of the thesis is submitted.

Signature of Candidate:		Date:	08/03/2021
-------------------------	---	-------	------------

### Submission

Submitted By ( <i>name in capitals</i> ):	Puneet Singh Chhabra
Signature of Individual Submitting:	
Date Submitted:	08-03-2021

### For Completion in the Student Service Centre (SSC)


Limited Access	Requested	Yes	No	Approved	Yes	No
<i>E-thesis Submitted (mandatory for final theses)</i>						
Received in the SSC by ( <i>name in capitals</i> ):				Date:		


## Inclusion of Published Works


Please note you are only required to complete the Inclusion of Published Works Form if your thesis contains published works under Regulation 6 (9.1.2)

### Declaration


This thesis contains one or more multi-author published works. In accordance with Regulation 6 (9.1.2) I hereby declare that the contributions of each author to these publications is as follows:


Citation details	P. S. Chhabra, Y. Pailhas, A. M. Wallace, J. R. Hopgood and Y. R. Petillot, "Target classification in SAS imagery using orthogonal basis selection," <i>OCEANS 2017 - Aberdeen</i> , Aberdeen, 2017, pp. 1-10, doi: 10.1109/OCEANSE.2017.8084647.
Author 1	
Author 2	
Signature:	
Date:	08/03/2021

Citation details	Puneet Singh Chhabra, Aurora Maccarone, Aongus McCarthy, Andrew Michael Wallace, Gerald Stuart Buller, "Analysis of Foliage Penetrating Photon Counting LiDAR Data for Underwater Mine Counter Measures" <i>OCEANS 2017 - Aberdeen</i> , Aberdeen, 2017,
Author 1	
Author 2	
Signature:	
Date:	08/03/2021

Citation details	P. S. Chhabra, A. M. Wallace, and J. Hopgood, "Improved image discrimination using fast non-linear orthogonal dictionary learning," in <i>European Signal Processing Conference, IEEE</i> , 2017.
Author 1	
Author 2	
Signature:	
Date:	08/03/2021

Please included additional citations as required.

Citation details	P. S. Chhabra, A. M. Wallace, and J. Hopgood, "Improved image discrimination using fast non-linear orthogonal dictionary learning," in European Signal Processing Conference, IEEE, 2017.
Author 1	
Author 2	
Signature:	
Date:	08/03/2021

Citation details	P. S. Chhabra, A. Maccarone, A. McCarthy, A. M. Wallace, and G. S. Buller, "Discriminating underwater lidar target signatures using sparse multi-spectral depth codes," in Sensor Signal Processing for Defence (SSPD) Conference, Edinburgh, 2016.
Author 1	
Author 2	
Signature:	
Date:	08/03/2021



# Abstract

In this thesis, we present the theory of high-dimensional signal approximation of multi-frequency signals. We also present both linear and non-linear compressive sensing (CS) algorithms that generate encoded representations of time-correlated single photon counting (TCSPC) light detection and ranging (LiDAR) data, side-scan sonar (SSS) and synthetic aperture sonar (SAS). The main contributions of this thesis are summarised as follows:

1. Research is carried out studying full-waveform (FW) LiDARs, in particular, the TCSPC data, capture, storage and processing.
2. FW-LiDARs are capable of capturing large quantities of photon-counting data in real-time. However, the real-time processing of the raw LiDAR waveforms hasn't been widely exploited. This thesis answers some of the fundamental questions:
  - can semantic information be extracted and encoded from raw multi-spectral FW-LiDAR signals?
  - can these encoded representations then be used for object segmentation and classification?
3. Research is carried out into signal approximation and compressive sensing techniques, its limitations and the application domains.
4. Research is also carried out in 3D point cloud processing, combining geometric features with material spectra (spectral-depth representation), for object segmentation and classification.
5. Extensive experiments have been carried out with publicly available datasets, e.g. the Washington RGB Image and Depth (RGB-D) dataset [108], YaleB face dataset<sup>1</sup> [110], real-world multi-frequency aerial laser scans (ALS)<sup>2</sup> and an underwater multi-frequency (16 wavelengths) TCSPC dataset collected using custom-build targets especially for this thesis.
6. The multi-spectral measurements were made underwater on targets with different shapes and materials. A novel spectral-depth representation is presented with strong discrimination characteristics on target signatures. Several custom-made and realistically scaled exemplars with known and unknown targets have been investigated using a multi-spectral single photon counting LiDAR system.
7. In this work, we also present a new approach to peak modelling and classification for waveform enabled LiDAR systems. Not all existing approaches perform peak modelling and classification simultaneously in real-time. This was tested on both simulated waveform enabled LiDAR data and real ALS data<sup>2</sup>.

This PhD also led to an industrial secondment at Carbomap, Edinburgh, where some of the waveform modelling algorithms were implemented in C++ and CUDA for Nvidia TX1 boards for real-time performance.

---

<sup>1</sup><http://vision.ucsd.edu/~leekc/ExtYaleDatabase/>

<sup>2</sup>This dataset was captured in collaboration with Carbomap Ltd. Edinburgh, UK. The data was collected during one of the trials in Austria using commercial-off-the-shelf (COTS) sensors.

# Acknowledgements

This thesis would not have been possible without the constant encouragement of my supervisors, Prof. Andrew Wallace (Heriot-Watt University, UK) and Dr. James Hopgood (University of Edinburgh, UK).

In particular, I would like to thank Andy for his patience and trust, for his insights, knowledge and guidance throughout my research years. My admiration and special thanks to my internal supervisor Dr. Yoann Altmann for his guidance and support throughout my PhD studies.

I would like to thank the entire University Defence Research Collaboration (UDRC) consortia and the Heriot-Watt university for their generous financial support throughout my PhD. I would also like to acknowledge my mentor Prof. Yvan Petillot (Heriot-Watt University, UK) for his support and and guidance. I would like to thank my friends, colleagues at the Heriot-Watt university and the university of Edinburgh, who have helped me during my PhD studies.

I would like to dedicate this thesis to my dearest wife Kritika. This thesis would not have been possible without her belief in me, even in the roughest of times. Last, but not the least, I will be forever indebted to my parents and my sister for their infinite support, trust and love.

Puneet S Chhabra

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background . . . . .	1
1.2	Motivation . . . . .	2
1.3	Contributions & Outline . . . . .	4
<b>2</b>	<b>Literature Review</b>	<b>7</b>
2.1	Sound Navigation and Ranging (Sonar) . . . . .	7
2.2	Light Detection and Ranging (LiDAR) . . . . .	9
2.2.1	Waveform Processing . . . . .	10
2.2.2	Seeing Through Foliage and Clutter . . . . .	16
2.3	3D Data Processing: Shape and Geometry . . . . .	22
2.4	Sparse Representation & Dictionary Learning . . . . .	29
2.4.1	Categorisation of Sparse Recovery Algorithms . . . . .	29
2.4.2	Dictionary Learning . . . . .	32
2.5	Anomaly Detection . . . . .	33
<b>3</b>	<b>Dictionary Learning &amp; Non-linear Data Approximation</b>	<b>38</b>
3.1	Simultaneous Approximation & Discrimination . . . . .	38
3.1.1	Sparse Discriminative Signatures (SDS) . . . . .	39
3.1.2	Orthogonal or Over-complete Dictionaries . . . . .	44
3.2	Kernel Dictionary Learning . . . . .	45
3.2.1	Kernelised Orthogonal Dictionary Learning . . . . .	46
3.2.1.1	Problem . . . . .	46
3.2.1.2	Proposed Approach . . . . .	47
3.3	Evaluation on benchmark datasets . . . . .	49
3.3.1	The AT&T (formerly ORL) Face Dataset . . . . .	52
3.3.2	Extended YaleB Face Dataset . . . . .	53
3.3.3	MNIST USPS Digit Dataset . . . . .	53
3.4	Summary . . . . .	54

<b>4</b>	<b>Anomaly Detection &amp; Classification from Aerial LiDAR Signals</b>	<b>55</b>
4.1	Mapping using Waveform LiDAR . . . . .	56
4.2	Anomaly Detection in LiDAR signals . . . . .	57
4.2.1	Contributions and Outline . . . . .	59
4.2.2	Problem Definition . . . . .	61
4.2.2.1	Layered Representation of a FW-LiDAR Signal . . . . .	61
4.2.2.2	The Concept of Normality . . . . .	62
4.2.2.3	Mapping Raw LiDAR Data into Sub-spaces . . . . .	63
4.2.3	Adopted Approach . . . . .	66
4.2.3.1	Step 1: Sparse Discriminative Signatures (SDS) . . . . .	67
4.2.3.2	Step 2: Layered Representation of SDS . . . . .	67
4.2.3.3	Step 3: Relationship Learning & Anomaly Detection . . . . .	68
4.3	Anomaly Detection Results . . . . .	70
4.3.1	Synthetic Data - Multi-Beam Modeller . . . . .	71
4.3.2	Layered Anomaly Score . . . . .	74
4.4	Summary . . . . .	76
<b>5</b>	<b>Peak Modelling &amp; Classification from Aerial LiDAR Signals</b>	<b>79</b>
5.1	Simultaneous Peak Extraction & Discrimination . . . . .	83
5.1.1	Pre-processing & Inflection Point Estimates . . . . .	83
5.1.2	Noise Estimation, Parameter Initialisation & Ranking . . . . .	84
5.1.3	Peak Estimation and Discrimination - A Sparse Solution . . . . .	84
5.1.3.1	Dictionary Generated using Mathematical Priors . . . . .	86
5.1.3.2	Orthonormal Dictionary Learning . . . . .	87
5.2	Multi-spectral Aerial LiDAR (Austria Trials) . . . . .	88
5.2.1	Mobile FW-LiDAR Backpack System (Edinburgh Trials) . . . . .	89
5.3	Point Cloud Classification Results . . . . .	91
5.3.1	Metrics used for Comparison . . . . .	91
5.3.2	Results on Simulated Data . . . . .	94
5.3.3	Results from Multi-Spectral Aerial LiDAR Scans . . . . .	99
5.3.4	Results on Terrestrial LiDAR Data (Edinburgh Trials) . . . . .	102
5.4	Summary . . . . .	103
<b>6</b>	<b>Object Classification from Single Photon Counting LiDAR Signals</b>	<b>104</b>
6.1	Introduction . . . . .	105
6.1.1	Data Acquisition: Setup & Targets . . . . .	105
6.2	LiDAR Data Analysis . . . . .	106

6.2.1	3D Shape Analysis: Point & Histogram Features . . . . .	106
6.2.2	Spin Images . . . . .	107
6.2.3	Curvature Representation (CR) . . . . .	109
6.3	TCSPC Object Discrimination Algorithm . . . . .	110
6.3.1	Stage 1 - Waveform Processing . . . . .	111
6.3.2	Stage 2 - Non-linear Dictionary Learning of SSR . . . . .	113
6.4	Results and Discussion . . . . .	114
6.4.1	Experiment 1 - Material Discrimination . . . . .	115
6.4.2	Experiment 2 - Mine Discrimination . . . . .	115
6.5	Summary . . . . .	118
<b>7</b>	<b>Object Classification from Sonar using Dictionary Selection</b>	<b>119</b>
7.1	Problem Formulation . . . . .	121
7.2	Proposed approach . . . . .	121
7.3	1-D Sonar Echoes - Experiments & Results . . . . .	130
7.4	Sonar Images - Experiments & Results . . . . .	133
7.4.1	Simulated Side-scan Imagery . . . . .	133
7.4.2	Real SAS Imagery . . . . .	141
7.5	Summary . . . . .	142
7.5.1	What is an optimal dictionary? . . . . .	142
7.5.2	Importance of SNR & Pixel Resolution . . . . .	144
7.5.3	Effect of occlusion on classification accuracy? . . . . .	144
<b>8</b>	<b>Conclusion</b>	<b>146</b>
<b>A</b>	<b>Proposition Proof</b>	<b>149</b>
	<b>Bibliography</b>	<b>151</b>

# List of Tables

3.1	Accuracies(%) on RGB-D Washington dataset. A set of pre-defined training and test datasets ([108]) were used for these experiments. . . . .	51
3.2	Accuracies(%) on AT&T and YaleB Datasets . . . . .	52
3.3	Accuracies(%) on USPS Digit Dataset . . . . .	53
4.1	LiDAR Simulator Configuration . . . . .	70
5.1	Comparison Criteria for Waveform Fitting Algorithms . . . . .	81
5.2	Statistics of the Peak Estimation and Discrimination (TDR - True dection rate (%), FDR - False detection rate (%), MHD - Modified Hausdorff distance, RMSE - Root mean square error). The 2ns pulse is used for the RJMCMC approach at 600, 1200, 2400 and 4800 iterations. . . . .	96
5.3	Average prediction accuracy on unseen data . . . . .	98
5.4	Average training time for MLP . . . . .	98
6.1	Experiment Key Parameters . . . . .	107
6.2	Curvature Representations using Eigenvalues . . . . .	110
6.3	Targets under investigation. These are designed using materials (Table 6.4)	114
6.4	List of all the materials used in our experiments. . . . .	116
6.5	Confusion Matrix - Material Discrimination . . . . .	116
6.6	Effect of <i>Depth Representation</i> (DR) on accuracy . . . . .	117
6.7	Confusion Matrix - Mine Discrimination . . . . .	117
7.1	State-of-the-art and benchmark rate of misclassification results on the Connectionist dataset. Our approach performs best in both set of experiments.	131
7.2	<i>Dataset01</i> confusion matrix, using Eigenfaces [132]. Image pixel resolution was set to 5cm and no noise was added. . . . .	135
7.3	<i>Dataset01</i> confusion matrix, using our approach. Image pixel resolution was set to 5cm and no noise was added. . . . .	135
7.4	<i>Dataset02</i> confusion matrix, using the proposed approach. Image pixel resolution was set to 5cm and no noise was added. . . . .	138
7.5	<i>Dataset02</i> confusion matrix using Eigenfaces [132]. Image pixel resolution was set to 5cm and no noise was added. . . . .	140

7.6 Confusion matrix, using our approach on the MUSCLE SAS image dataset.141

# List of Figures

1.1	Various real-world scenarios that benefit from FW-LiDAR systems. (Left) Forest health monitoring (Source - NeonScience [126]). (Right) Underwater mine countermeasures (Image still from War at Sea: Scotland's Story, Source - BBC [156]) . . . . .	2
1.2	An aerial scenario. This image illustrates how multi-spectral full-waveform time correlated single photon counting data is collected. . . . .	3
1.3	Two phase approach to a survey mission, e.g. large dense forest or terrain. Phase 1 - low-frequency scans coupled with real-time data processing identify anomalous regions. Phase 2 - High frequency scans coupled with slower post-processing algorithms for further shape and object type classification. . . . .	4
1.4	Thesis outline. . . . .	5
2.1	Left: Aerial image of City of Amiens. Right: Difference between last and first pulse digital elevation map (DEM). Source: Mallet et al. [118] . . . . .	10
2.2	Full-waveform signal recorded for a single pixel. The horizontal axis is the round-trip distance and the vertical its amplitude. Source: Wallace et al.[179] . . . . .	12
2.3	An advanced faint echo extraction method proposed by Stilla et al. Source: Stilla et al. [160] . . . . .	15
2.4	(Left) Pulses detected by standard peak method. Black: no pulse; White: pulse detected. (Right) A traffic light paradigm used to show faint echoes that are detected using this technique. Red: no pulse; Yellow: unknown; Green: weak echo. Source: Stilla et al. [160] . . . . .	15
2.5	Some examples of tree segmentation using the methods presented above. (Left) A region of interest (ROI) into a voxel structure before processing. (Middle) Two segmented trees (A and B). (Right) Examples of segmented trees. Source: Reitberger et al.[143] . . . . .	19
2.6	Sample results from [79] where authors have applied spatial filtering methods on point cloud data to separate targets from foliage and undergrowth. . . . .	20
2.7	(Left) Optical path in the Jigsaw Sensor (Right) The sensors optical head, with all the components integrated. Source: Marino et al.[113] . . . . .	21
2.8	(Left) Spin image and its local coordinate system. (Right)3D model and 2 sampled spin images. Source: Johnson et al. [97] . . . . .	25



2.9	Partial matching using a heat kernel descriptor. Source: Ovsjanikov et al. [129] . . . . .	26
2.10	Tensor Voting Framework as proposed by Medioni et. al [123]. . . . .	28
2.11	Compressive sensing process and a high-level categorisation of sparse recovery algorithms. . . . .	30
2.12	In this work we present a high-level categorisation of anomaly detection algorithms. . . . .	34
2.13	Treating the power-line detection in LiDAR data as an anomaly detection algorithm. Source [49] . . . . .	35
2.14	Detecting abnormal regions at the Chernobyl Nuclear site. Source [30]. . . . .	36
3.1	RGB-D Washington and YaleB Face dataset. . . . .	51
3.2	Exemplars of the learnt kernel dictionary. . . . .	54
4.1	Example FW-MSL signals used to detect anomalies. (Left) A typical FW-MSL return of a large foot-print with a range resolution of $\sim 5cm$ per bin. The backscattered returns shown here are recorded for 1020, 510, 550 and 590 nm wavelengths. This depth profile is a mixture of different spectra: vehicle, conifer, undergrowth and bare-ground. When observed from an altitude the tree top is the first peak, 600 <sup>th</sup> bin on the left. (Right) Stacked FW-MSL exemplars. . . . .	59
4.2	(A) A cross-section of a single conifer tree. (B) A <i>push-broom</i> style (small-foot print) dense scan pattern resulting in a cross-section slice of a tree. A <i>flash</i> style (large-foot print) scan pattern is illustrated in (C). (D) The forest/tree layered model. . . . .	61
4.3	Sliding window concept in order to extract over-lapping range-amplitude series. . . . .	64
4.4	Conceptual diagram of multi-wavelength <i>atom-layer</i> relationship matrix (coefficient matrix) which is learnt for several over-lapping sub-sequences (columns) on range-amplitude ladar signals. . . . .	68
4.5	Realistic LiDAR waveform generation workflow. (A) Natural terrain generation method using fractal based Brownian Motion Fields. (B) Layer 1 is a factorial Brownian filed (fBF) and layer 0 is a texture layer that is mapped to the elevation model. A closed-source tree modeller, Onyxtree is used to generate 3D tree models. (C) Tree 1 - Picea Engelmannii. (D) Tree 2 - Pinus Nigra. (E) Tree 3 - Pinus Parviflora. (F) The depth and intensity image from the rendered scene is then used to generate synthetic LiDAR waveforms. . . . .	72
4.6	Measured material spectral reflectance for natural (tree bark, needle and soil samples) along with data from the Aster spectral library for selected <i>man-made objects</i> : building material and galvanized steel. . . . .	73
4.7	(Left) Example FW-LiDAR signal. (Right) A ray-traced scene with three different tree species. . . . .	74

4.8	Layer wise reconstruction error computed using 4.2.3.7 is illustrated here. These results also label the species of the trees using a discriminative optimisation routine 3.1.1.9. Samples 34 and 69 record high reconstruction error as this is bare forest floor with some undergrowth. Layer 1 and 9 record the lowest reconstruction error as these layers record mostly background and system noise. . . . .	75
4.9	Anomaly detection on simulated scenes. . . . .	75
4.10	Material spectrum of an individual tree with a hidden anomaly. . . . .	76
4.11	A 3D point cloud of the test sample and FW-LiDAR signals belonging to 2 different conifer species. Anomalies are introduced in the form of man-made objects at different depths and layers. . . . .	78
5.1	The Proposed Approach. . . . .	82
5.2	Peak dictionary illustrating exemplar atoms generated using Eq. (5.1.3.2) for a peak . . . . .	86
5.3	The study areas. (Left) Site 1 - Schwengenthal, Austria. Markers in pink and blue represent a small group of co-registered scan points collected using the LMS-Q560 and LMS-Q780 sensors. The waveforms are illustrated in Figure 5.4. (Center) A zoomed out view of the scan area. (Right) Site 2 - Bertholstein, Austria. . . . .	89
5.4	Example waveforms from Site 1 - Schwengenthal collected using the LMS-Q1560 (blue) and LMS-Q780 (pink) sensors. . . . .	90
5.5	Nearest-neighbour correspondences between the LMS-Q1560 and LMS-Q780 point cloud data. . . . .	90
5.6	Full-waveform LiDAR backpack system for terrestrial scanning . . . . .	91
5.7	(Left column) Shape parameter ( $\rho$ ) histograms of different targets - vegetation (blue), forest floor (orange) and vehicle (yellow) for each tree specie. (Right column) The cumulative peak shape $\rho$ distribution for different targets within each scene. . . . .	97
5.8	(Top) Unsupervised classification results using the SPeED algorithm on the Riegl dataset collected in Austria. (A) Observed scene, (B) 1550nm, (C) 1064nm and (D) Combined result. (Bottom) Cumulative peak shape ( $\rho$ ) distributions for different targets. . . . .	100
5.9	Site 2 - Bertholstein, Austria 3D point cloud classification results. (Left) - Classification results rendered and colour coded: buildings (Red), high vegetation (dark green), lower vegetation (green). (Middle) - We compute geometry features (introduced in chapter 6) in individual 3D voxels (each voxel is colour coded). (Right) Each voxel is colour coded (white - high point density and black - low point density). . . . .	101
5.10	Terrestrial LiDAR data was collected at the King's buildings campus using the backpack system. . . . .	102

5.11	Geo-referenced point cloud data generated using the waveform processing algorithm, SPeED. The colours here represent: first returns (blue), first-of-many (green) and last returns (orange) . . . . .	102
6.1	a) A schematic of the experimental set-up; b) (left) an exemplar under investigation with different targets; c) full-waveform LiDAR signatures for a single pixel. . . . .	106
6.2	(a) The cylindrical co-ordinate system and its oriented-point basis. (b) Two different oriented-points and their neighbourhood points shown with a red ellipse on one of the exemplars used. (c) Spin maps for the two oriented-points. . . . .	107
6.3	Raster images of different geometric features computed for each 3D point, controlled using a local neighbourhood radius, $r$ . (a) Anisotropy $A_{\mathcal{E}}$ , b) Planarity $P_{\mathcal{E}}$ , c) Sphericity $S_{\mathcal{E}}$ and d) Linearity $L_{\mathcal{E}}$ Table 6.2. . . . .	109
6.4	The adopted approach to process multi-spectral TCSPC LiDAR data. Figure 5.1 illustrates the flowchart for the supervised algorithm, SPeED and algorithm 3 lists the pseudo-code for the K-SAD algorithm. In this chapter, as noted in the infogram above, the K-SAD algorithm is used as a semi-supervised classification algorithm. . . . .	110
6.5	Multiple peaks estimated (red curve) using the SPeED algorithm on the original SPC histogram (green). . . . .	111
6.6	Reflectance spectra extracted using our technique for different materials (plastic, metal, sand, concrete). . . . .	113
6.7	Results of underwater foliage penetration and point cloud classification. . . . .	114
6.8	Mine clusters using the spectral depth codes, $\mathbf{Q}$ , on the spectral axis, echo area; and the depth axis, linearity and sphericity, respectively. . . . .	117
7.1	The proposed system has been adapted to the MLO classification problem by encompassing a dictionary building routine specifically for underwater sonar signals. The prediction module is a generic classifier with class labels as its output. . . . .	122
7.2	a. Some of the dictionary atoms used in the experiments. The order of these atoms is optimised for every input signal. b. An illustration showing that the DCT approximation is more accurate using the same number of coefficients when compared with DFT. . . . .	125
7.3	Construction of the PDFB or contourlet transforms. . . . .	128
7.4	Spectral envelope of a mine and rock sonar return from the Connectionist Sonar Dataset. . . . .	130
7.5	Effect of pre-defined dictionary on classification. . . . .	132
7.6	Example side-scan images from <i>Dataset02</i> , created using a real-time side-scan simulator [132]. Images shown here have a $5cm$ pixel resolution and are on a plain featureless sea-bed. . . . .	134

7.7	Misclassification rate for <i>Dataset01</i> illustrated as a function of SNR: (a) at <i>3cm</i> , (b) at <i>7cm</i> and (c) at <i>18cm</i> pixel resolution. . . . .	136
7.8	The effect of pixel resolution on misclassification error for <i>Dataset01</i> using the proposed approach. (a) Misclassification error rate for individual targets illustrated as a function of pixel resolution. (b) Mean misclassification rate of different target at different pixel resolutions is illustrated as a function of Rayleigh noise. . . . .	137
7.9	Misclassification rate for <i>Dataset02</i> illustrated as a function of SNR at (a) at <i>3cm</i> , (b) at <i>7cm</i> and (c) at <i>18cm</i> pixel resolution. . . . .	139
7.10	Misclassification rate for different pixel resolutions of <i>Dataset02</i> illustrated as a function of Rayleigh noise. Even at <i>10cm</i> pixel resolution, the rate of misclassification stays below 22% for $-20\text{dB}$ SNR. . . . .	140
7.11	Real synthetic aperture sonar exemplar images of mine like objects from the MUSCLE SAS dataset. . . . .	140
7.12	Impact of different dictionary categories on rate of misclassification for different MLOs. . . . .	142
7.13	Effect of pre-defined dictionary on target classification. . . . .	143
7.14	(Top) Example images of 4 different targets. (Bottom) Targets under 5%, 16%, 25% and 50% occlusion. . . . .	144
7.15	Rate of misclassification for 7 different targets achieved using the proposed approach on <i>Dataset03</i> by varying the percentage of occluded pixels. . .	145
7.16	Rate of misclassification for 4 different targets achieved using the proposed approach on <i>Dataset01</i> by varying the percentage of occluded pixels. . .	145

# Chapter 1

## Introduction

### 1.1 | Background

Laser based range sensing is fairly modern. NASA first began work on LiDAR prototypes for eventual space exploration. These efforts were evident in the 1971 Apollo 15 Mission where astronauts used a laser mapping sensor to map the surface of moon [1]. The following decade scientists efforts were focussed on measuring atmospheric properties such as: ocean water, forest canopies and polar ice sheets [87, 170]. Before laser based mapping, high-resolution terrain mapping and profiling was achieved using photogrammetry and Radar [67, 41]. Low-resolution mapping was achieved using radar or space stereo imagery. Radar can penetrate cloud cover and efficiency over large areas made it very popular, but it has certain limitations. Radar struggles to penetrate forest canopies to map ground elevation and is prone to artefacts in urban areas and steep elevation.

The ground coverage of LiDAR based sensors are similar to photogrammetric sensors, e.g. aerial cameras. LiDAR's capability to provide geometry and high-speed scanning is where they differ from other active sensing technologies. Early 2000's saw a steep rise in the amount of data generated by such sensors and an improvement in the data processing techniques to handle and process such data. In comparison, state-of-the-art, full-waveform LiDAR (FW-LiDAR) systems are capable of capturing 250,000 or more pulses per second and capture multiple returns from individual pulses[179].



Figure 1.1: Various real-world scenarios that benefit from FW-LiDAR systems. (Left) Forest health monitoring (Source - NeonScience [126]). (Right) Underwater mine countermeasures (Image still from War at Sea: Scotland’s Story, Source - BBC [156])

Laser range finding can be achieved using either triangulation or time-of-flight, of which the latter is more suited for long range measurements. The principal methodologies for a time-of-flight system include phase shift in an AM signal, measurement of frequency shift in a FM modulated signal, or measurement of transmit-receive pulse separation [9]. A 3D image (or Point Cloud) can be constructed, either by recording the received radiation of a static laser beam that encompasses the target on a focal array of independent pixels, or the laser beam must be scanned across the scene. Some of the earliest commercial full-waveform systems came to the market in 2004 [91] and with the increasing availability of 3D scanners there has been a significant push towards faster and more intelligent ways to process such data.

## 1.2 | Motivation

A study by the UK ministry of defence (MoD) in 2010 identified several issues within the signal processing domain, especially in cluttered and challenging environments. These are jointly characterised these into themes and challenges [128]. Inference, in such complex environments, is data driven and data is implicit to the sensing modalities used for intelligence and counter-intelligence. In this thesis, we focus primarily on challenges 8 (Maximising the Information Capture from LiDAR Returns) and 12 (Signal Processing Algorithms and Techniques to Manage Noisy 3D Point Clouds)[128].

Not only future battlefields, as illustrated in Figure 1.1, but also scientists and engineers are making use of modern FW-LiDAR systems and assisting in solving other global

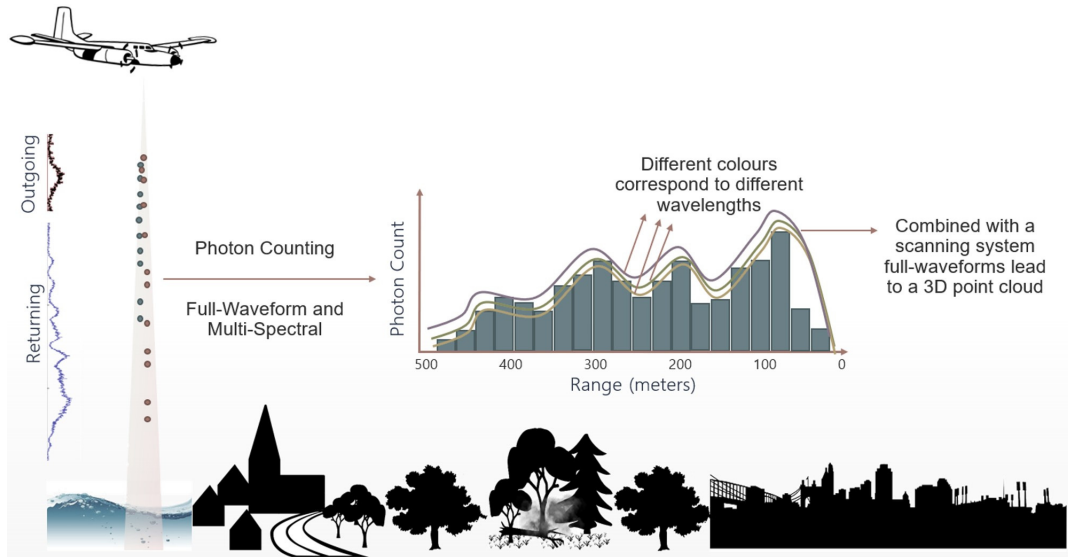


Figure 1.2: An aerial scenario. This image illustrates how multi-spectral full-waveform time correlated single photon counting data is collected.

problems. For example:

- Forest health monitoring, e.g. tree species and studying the undergrowth [125], classification [32, 142, 175],
- Cloud and aerosol characterisation [32, 142, 175], and
- Driverless Cars, e.g. driving in bad weather.

High frequency state-of-the-art sensors demand efficient data management and intelligent processing for effective situational awareness in complex and harsh environments, e.g. dense forests, driverless cars and conflict management. This data primarily encodes obscured information (targets). Signal processing is fundamental to decode such data, detect anomalies and targets and to the capabilities of other communication systems. Figure 1.2 illustrates an environment monitoring scenario. A FW-LiDAR, on-board an aerial platform, transmits tens of thousands of pulses every second. The on-board sensor can record and time stamp backscattered pulses down to a resolution of a photon. The underlying motivation behind this work is to categorise a survey mission (e.g. aerial mapping shown in Figure 1.2) into two phases: (i) Phase 1 - low-frequency scans (with a larger laser footprint) that identify key regions of interests, e.g. anomalous spectra in

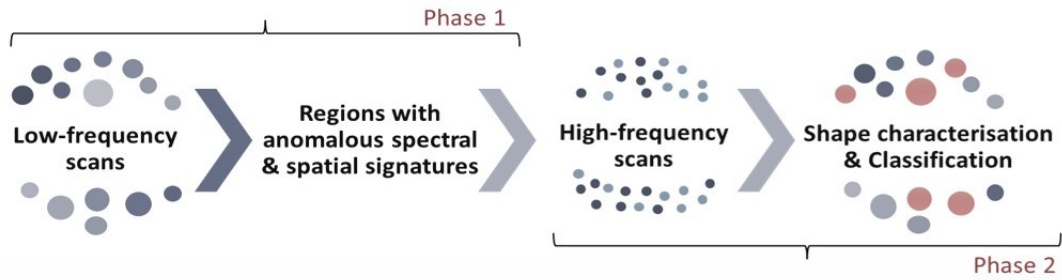


Figure 1.3: Two phase approach to a survey mission, e.g. large dense forest or terrain. Phase 1 - low-frequency scans coupled with real-time data processing identify anomalous regions. Phase 2 - High frequency scans coupled with slower post-processing algorithms for further shape and object type classification.

forest growth or spatial signatures; and (ii) Phase 2 - high-frequency scans (with a smaller laser footprint) on the anomalous regions for shape and material characterisation. Figure 1.3 presents this idea as an illustration.

Researchers, in the last two decades, have suggested robust signal modelling (statistical) techniques to process FW-LiDAR data. A detailed review is presented in Chapter 2. In this work, we analyse full-waveform multi-spectral TCSPC LiDAR data, collected using [114], 1D Sonar signals [74], synthetic aperture sonar and side scan sonar images [131, 132, 74] to improve aerial and bathymetric situational awareness. The underlying key idea of this thesis is:

*We show that intelligent encoding (Chapter 3) can be more practical and efficient in comparison to robust signal modelling techniques. The idea is to use compressed representation of time-series FW-LiDAR data (Chapter 3) and combine that with geometric (shape) and spectral properties (Chapter 6). We also show how to detect spectral anomalies (Chapter 4) and classify objects in LiDAR (6) and sonar data Chapter 7.*

### 1.3 | Contributions & Outline

Figure 1.4 illustrates an outline and organisation of this thesis. An introduction and background survey is presented in Chapter 1 and 2, respectively. Chapter 3 presents the



theory and the algorithms developed. The use of these algorithms in practice and on real-world LiDAR (aerial and underwater) and Sonar data is presented in Chapters 4, 5, 6 and 7.

- Chapter 2 describes Sonar and laser based distance measurement schemes along with topographic LiDAR principles. A detailed survey is presented on the full-waveform processing and modelling techniques. We also discuss sparse approximation and dictionary learning techniques and a high-level categorisation of sparse recovery algorithm is presented. We conclude Chapter 2 by presenting a survey on anomaly detection and dictionary learning techniques.
- Chapter 3 presents novel data approximation and dictionary learning techniques. An optimisation and discrimination algorithm, *Sparse discriminative signatures*, that can re-order and select a subset of dictionary atoms from a larger sub-space is presented in Section 3.1.1. With the K-SAD algorithm, Section 3.2, we show that if

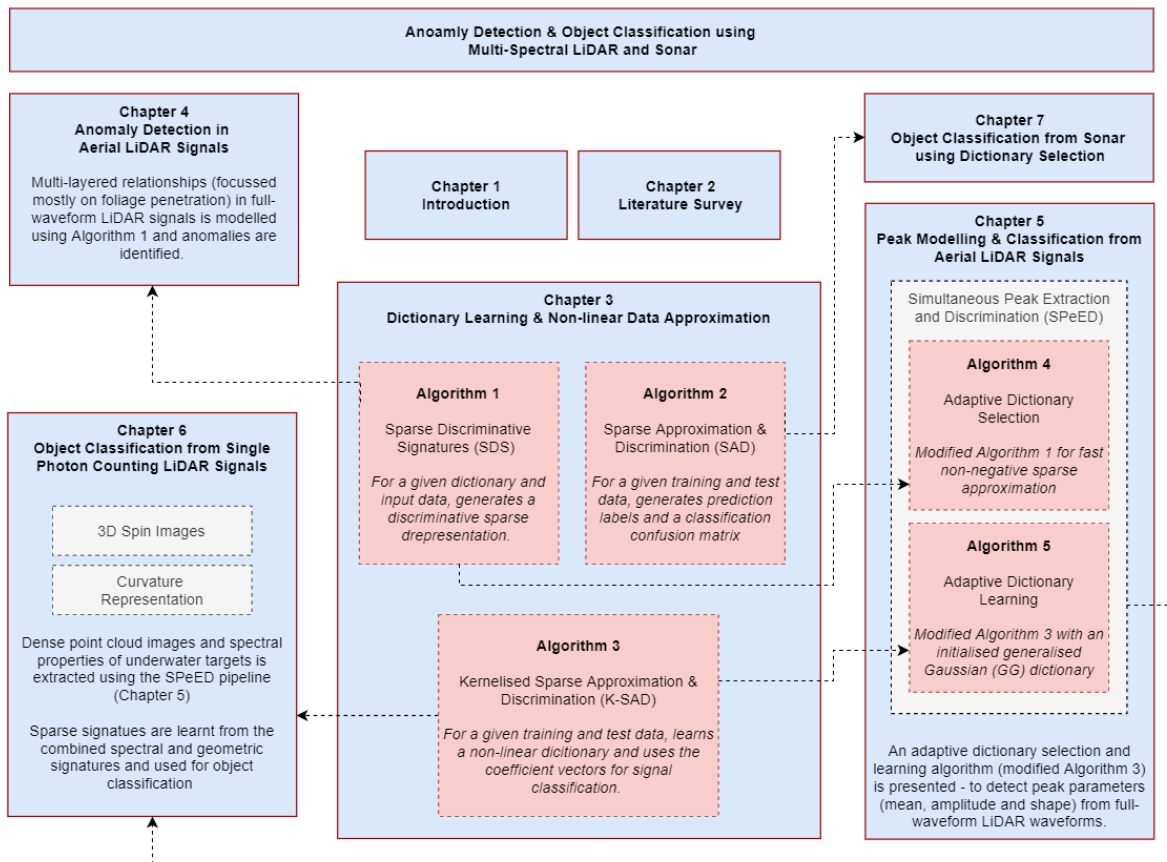


Figure 1.4: Thesis outline.

we learn a discriminative orthogonal dictionary instead, learning and classification run-time can be significantly reduced. These algorithms, with minor variations, have been applied to 1D signals (acoustic and full-waveform LiDAR signals), 2D images (side scan and synthetic aperture sonar) and 3D point cloud (LiDAR data).

- In chapter 4, we show how we make use of *sparse discriminative signatures* (Algorithm 1) to detect anomalies in multi-spectral full-waveform LiDAR waveforms.
- In chapter 5, we consider the problem of extracting multiple peaks and classifying them, simultaneously. We present an algorithm (Section 5.1), simultaneous peak extraction and discrimination (SPeED), for the analysis and discrimination of small foot-print airborne and terrestrial FW-LiDAR.
- Chapter 6 reports the analysis and discrimination of TCSPC LiDAR signals measured on custom underwater targets. In this chapter we use the novel non-linear orthogonal dictionary learning algorithm, proposed in Chapter 3 (Algorithm 3) to detect and classify underwater targets. Finally, we propose two shape representations that combine geometry and material spectra.
- Chapter 7 presents a signal approximation and discrimination approach that applies to both 1D and 2D datasets. A two-stage approach to underwater sonar signal discrimination is proposed. The work presented here is developed from an earlier approach [53] to optimal classification of different tree species using synthetic 1D photon count histograms. Here it has been extended to represent and classify multiple types of targets in both 1D sonar echoes and 2D sonar images.
- Chapter 8 provides an overall conclusion to the thesis.

# Chapter 2

## Literature Review

### 2.1 | Sound Navigation and Ranging (Sonar)

Data collected using modern sonar systems can be highly redundant, consisting of a highly correlated and densely sampled version of the physical world. Modelling and extracting a compressed representation of the underlying scene and signal conditioning that leads to that observation is highly beneficial for signal reconstruction and target recognition. Such algorithms should capture salient features based on the generating cause, e.g., the physical phenomena causing a spherical target to appear circular in a 2-D image plane. The feature space is generally of lower dimension than the data space, and the focus of this work is to identify such a representation, a *dictionary*, that supports Automatic Target Recognition (ATR) for underwater mine countermeasures (MCM).

Active sensing for MCM requires the transmission of a pulse; for traditional side-scan sonar (SSS) and synthetic aperture sonar (SAS), the pulse is a sound wave. In each case, the transmitted pulse reflects from the target or the environment, then is recorded and processed to form an image. Detecting and differentiating signals of interest from clutter underwater is extremely challenging and has been studied extensively [23, 154, 77, 63, 76, 132].

In [23], authors propose an approach to reduce false detections by employing a probabilis-

tic calibration scheme on top of support vector machines (SVM) and Gaussian Process classifiers. They test their approach on SAS imagery of two target types. Although they do not report state-of-the-art results, using uncertainty they seem to classify previously unseen target types with significant confidence. In order to compute detection statistics, Groen et al. [77, 76] propose a method based on target template matching. Detection statistics with respect to vehicle speed and range are computed. Fundamentally, template based approaches have degraded performance for unseen target classes.

In [76], the authors propose a model based matching approach to object classification. Similar to the methods used in radar, templates generated by a sonar model are preferred in their approach. The sonar models, for various objects with varying facets, were built on several assumptions, e.g. the acoustic paths between the target and sonar always follow a straight line, the target is assumed to be a 3D surface. Their detection was based on a correlation operator and a probabilistic model matching scheme. The experiments report reasonable results but recommend a maximum of 36 templates to be used when the grazing angle is known within a few degrees. The number of templates could rise when the grazing information is unknown.

A more sophisticated feature selection and classification scheme applied to underwater object detection was proposed by Sawas et al. [154]. The approach is highly influenced by a feature selection approach, *adaboost*, a method for combining complex classifiers in a *cascade* which assists in clutter removal. The overall algorithm complexity and detection time using the cascade approach is very low. Authors in [132] propose an approach for the classification of mine like objects (MLO) using Eigenfaces, motivated by the principle component analysis (PCA) technique. They report state-of-the-art results both on SSS and SAS imagery. Fakiris et al. [63] report a comprehensive study on MLO classification on SSS imagery with 70 man-made targets. They propose an automatic detection and classification scheme which is composed of three stages: texture based feature extraction, dimensionality reduction using ICA, and finally, an unsupervised clustering approach for classification.

## 2.2 | Light Detection and Ranging (LiDAR)

Airborne and ground based LiDAR, when pulsed, measure the round trip time of a laser pulse. In comparison, a continuous system measures the phase difference between the reference and measured signal which is then mapped to a 3D point cloud. In an urban scene, targets or objects are mostly distributed in space, the reflected signal in a continuous system is the superposition of echoes (backscatters) at different distances. In a discrete case with a fixed number of targets, the power of the received signal is the sum of the contributions from  $N$  targets. The standard equation is borrowed from the radar equation (2.2.0.1). For more details see [118],

$$P_r(t) = \frac{D^2}{4\pi\lambda^2} \int_0^A \frac{\eta_{sys} \eta_{atm}}{R^4} P_t \left( t - \frac{2R}{v_g} \right) \sigma(R) dR. \quad (2.2.0.1)$$

The combined power of the receive signal can be considered as the sum of contribution of  $N$  objects with their individual characteristics:

$$P_r(t) = \sum_{i=1}^N P_{r,i}(t) * \eta_{sys}(t) \eta_{atm}(t), \quad (2.2.0.2)$$

where  $P_r$  is the received power,  $D$  is the receiving sensor's aperture diameter,  $\lambda$  the wavelength,  $A$  the flying altitude,  $R$  the distance to the object,  $v_g$  the group velocity of the laser pulse,  $\eta_{atm}$  and  $\eta_{sys}$  are the atmospheric and system transmission factors respectively,  $\sigma(R) dR$  the effective differential cross-section [176] and  $P_{r,i}(t)$  is the echo from the  $i^{th}$  object with a spatial spread  $[R_i + \Delta R, R_i - \Delta R]$ .

The echo of the  $i^{th}$  object can be expressed as:

$$P_{r,i}(t) = \frac{D^2}{4\pi\lambda^2} \int_{R_i - \Delta R}^{R_i + \Delta R} \frac{1}{R^4} P_t \left( t - \frac{2R}{v_g} \right) \sigma_i(R) dR. \quad (2.2.0.3)$$

Laser scanners that provide one backscattered echo per laser pulse. We refer *diffraction cone* as the area covered (in 2D) by the laser beam on the ground. Several pulse scanning systems are capable of recording more than one pulse, e.g. first and last pulse. Thiel et

al. [165] have shown how to discriminate up to six individual returns from a single pulse.

### 2.2.1 | Waveform Processing

In the case of dense vegetation and forest scenes, the first echo often belongs to the canopy and the last belongs to ground, but this is not always true. In order to verify this we can make use of 3D visualisation tools that can plot the 3D point cloud. Full-waveform LiDAR systems such as [91] have the ability to record the complete waveform of the backscattered echo signal. Such systems sample at a frequency of around 1GHz and are capable of capturing the entire profile distribution of object hit by the laser pulse. Full-waveform LiDAR returns can be decomposed into a sum of echoes. These echoes, geolocated or arranged in a 3D grid, result in a 3D point cloud (Section 2.2.1). The resulting point cloud data can be used for interpretation, e.g. point classification, building reconstruction, segmentation etc. We begin this section presenting few signal processing methods that can detect faint echoes and characterise such echoes with additional information (e.g., surface and geometry)

With the advancements in the laser radar sensor industry research in sensing through foliage using such sensors is fairly new. In the past LiDAR sensors have been used for forest scene segmentation and tree classification [140]. Military vehicle detection [133, 127, 121, 10] using LiDAR sensors has been very popular. Recent works include MIT Jigsaw program [113, 171], algorithms and techniques by the Swedish Defence Research Agency

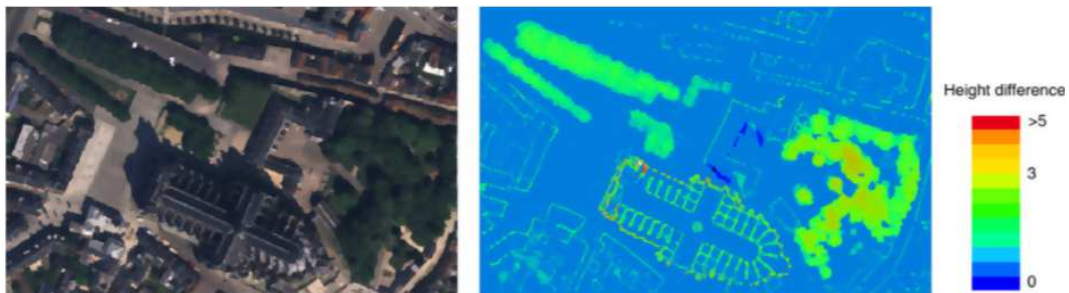


Figure 2.1: Left: Aerial image of City of Amiens. Right: Difference between last and first pulse digital elevation map (DEM). Source: Mallet et al. [118]

(FOI) [78, 159] and applications in Automatic Target Recognitions [11, 194, 162, 115] have surfaced recently.

**Signal Modelling** Modelling of full-waveform LiDAR signals has been an active area of research which involves the processing of raw data (1D signals) to extract range (depth) and material spectra. Parametric approaches are always chosen to model echoes within a waveform and mathematical models characterise the shape and reflectance properties of such echoes. Methods such as Non-linear least Square (NLS) [88, 140] or a maximum-likelihood method to model the waveform as a mixture model [137] can be used. For small foot-print LiDAR waveforms, Wagner et al. [176] has shown that we can model a full-waveform LiDAR signal as a sum of Gaussian pulses. Recently, Wallace et al. [179] used a Bayesian approach [85] to model LiDAR signals with a specific modelling function, e.g. a set of piece-wise exponential functions. A grammar of such functions is defined and an optimisation technique is suggested resulting in the detection of faint echoes and a dense 3D point cloud.

Mathematical models can be used to define a set of parameters representing a waveform. We would like to represent a backscattered waveform with such models for: echo (peak) estimation in a signal and characterise shape and reflectance characteristics of the object that falls within a laser footprint. This section is dedicated to describe methods for waveform modelling (Non-linear methods [98], Maximum Likelihood based methods ([139]) and Hernández-Marín et al. [85] state-of-the-art stochastic approach ). Please see Mallet et al. [118] for further details and comparisons. Below we present a brief discussion on the reasons behind waveform modelling and how it is relevant to this work. We begin this section by formally defining a waveform function.

Let us represent a waveform as a function  $y = f(x_i)$ . This function needs to be decomposed into a sum of  $N$  components:

$$y_i = \sum_{k=1}^N \psi_k(x_i) + b_i, \quad (2.2.1.1)$$

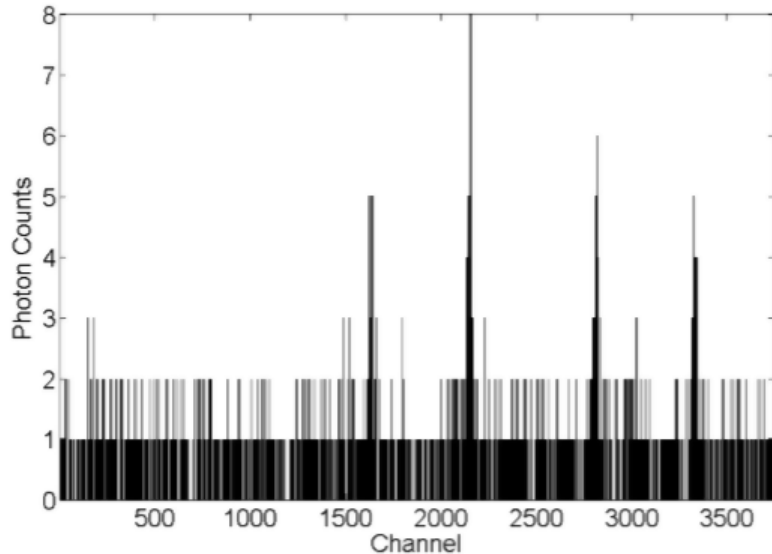


Figure 2.2: Full-waveform signal recorded for a single pixel. The horizontal axis is the round-trip distance and the vertical its amplitude. Source: Wallace et al.[179]

where  $\psi$  is the echo model with a set of parameters  $\theta (f = \sum_k \psi_k)$ ,  $f$  is the waveform model,  $\{x_i\}_{i=1,\dots,N}$  is a sequence of uniformly spaced points,  $y = \{y_i\}_{i=1,\dots,N}$  the sampled waveform, and  $b$  is the background noise.

Wagner et al. [176] proposed a method that considers a waveform as a mixture of Gaussian distributions. Received waveforms are a convolution between the transmitted pulse (mostly Gaussian with a calibrated width) and the surface scattering function within the laser footprint. When the laser footprint (laser pulses that travel longer distance diverge and create a large footprint) is large, the Gaussian assumption holds well.

In urban scenes laser peaks are distorted (subject to roof tops and vehicles and their materials). Chauve et al. [45] proposed a more generalised Gaussian function to improve signal fitting. Persson et al. [137] propose a pulse detection method based on an iterative Expectation-Maximisation (EM) approach. They assume the return waveform as a sum of Gaussians, but it is possible to extend it to other probability density functions. Hernández-Marín et al. [85] propose a method to fit terrestrial LiDAR waveforms with a set of four piecewise exponential function. This method creates a library (grammar or bag of words) of such functions that can be defined and adapted. Decomposing a backscattered signal into its individual components is an essential first stage in 3D laser



mapping. Figure 2.2 illustrates an example of a full waveform that is recorded for an individual pixel of a distributed target.

The above method [85] does not assume or fix the number of peaks in the returned waveform. However, it can detect even faint returns (mostly echoes of target closer to the ground). One major drawback with this method is that it is iterative and are not practical for real-time applications. Experiments in Chapter 4 show further comparisons and results. For urban scenarios, the method proposed by Kirchhof et al. [104] shows improvement in range estimates of opaque and semi-penetrable objects.

**Echo Extraction and Classification** There are two main motivations behind signal processing of backscattered waveforms.

1. Decomposing the waveform into a sum of components of echoes and characterising them in order to detect objects along the path of the laser beam. Maximising the rate of detection is most desirable in such a scheme in order to generate denser 3D point clouds and then foster signal processing capabilities for information extraction. Such a scheme is highly beneficial to remote sensing and archaeological applications (e.g., segmentation of forestry data and artefact reconstruction).
2. Spatio-temporal signal analysis is performed and features are extracted within a 3D waveform space. One of the main advantages with the later scheme is the capability to extract faint echoes in a full-waveform signal. In this section we first look at some of the methods that determine range and extract echoes for object discrimination.

**Surface Modelling** Jutzi et al. [98] proposed an approach that models the emitted pulse, the surface, and the backscattered pulse in order to extract surface features. They perform deconvolution to remove the properties of the emitted signal from the received signal in order to obtain surface responses. Further, temporal position, length and amplitude of the corresponding surface features are approximated by Gaussian functions using the Levenberg-Marquardt method. Their experiments show that surfaces with a distance

corresponding to less than 0.15m can be resolved.

The received pulse is computed using the LiDAR equation ((2.2.0.1)) and incorporates the physical principles into their models, e.g. *geometrical reflectance*, *material reflectance* and *multi-surface reflections* to describe the influence of the surface on the laser beam. In our work, we strongly believe that such an approach can be used to detect anomalies in the signal domain and also to reduce the search space within the point-cloud in order to perform high-level tasks, e.g. segmentation and recognition. The steps involved are listed below:

1. Detect pulse with a noise dependent threshold using the emitted pulse model
2. Remove *emitted pulse characteristics* from the received waveform by performing *deconvolution* in the frequency domain
3. Estimate the surface response using the Wiener filter
4. Waveform fitting using the LM method

#### *Faint Echoes*

Most of the methods discussed so far require echoes to be modelled with an analytical function. Stilla et al. [160] show an interesting waveform stacking technique that establishes neighbourhood relationships between consecutive waveforms and improves the signal to noise ratio. By means of mutual information accumulation they generate a hypothesis for planes with different slopes. Each signal is then assessed by a likelihood value with respect to the accepted hypothesis. Finally, they perform classification applying thresholds. Figure 2.3 illustrates the approach. Compared to standard algorithms this method predicts faint echoes as can be seen in Figure 2.4.

#### *Improving point density and range*

With full-waveform LiDAR systems we aim to make use of the entire waveform to generate a dense 3D point cloud. The accuracy in range/distance estimate and point density has an direct impact on the quality of man-made object segmentation or recognition in a 3D point cloud. In an urban scenario or poor weather conditions, e.g. fog, rain

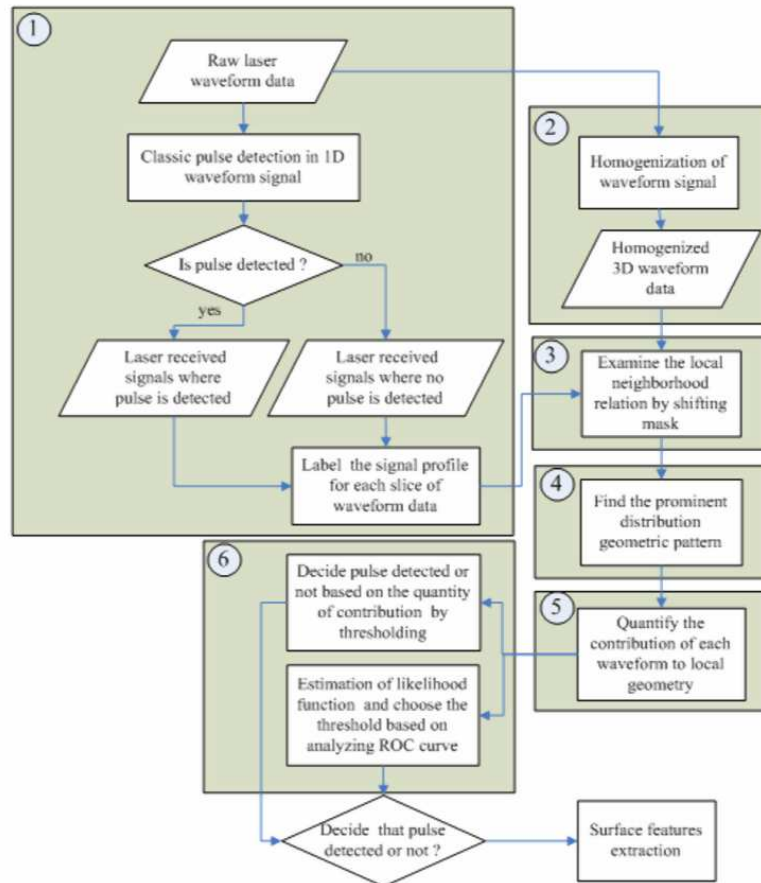


Figure 2.3: An advanced faint echo extraction method proposed by Stilla et al. Source: Stilla et al. [160]

or smoke, objects can be occluded, partially penetrable (e.g., glass windows and roof canopies) or even impenetrable. Robust distance estimate can improve object boundaries for object reconstruction and segmentation. Kirchhof et al. [104] propose a method to improve the point density and distance estimate which in turn improves the segmentation

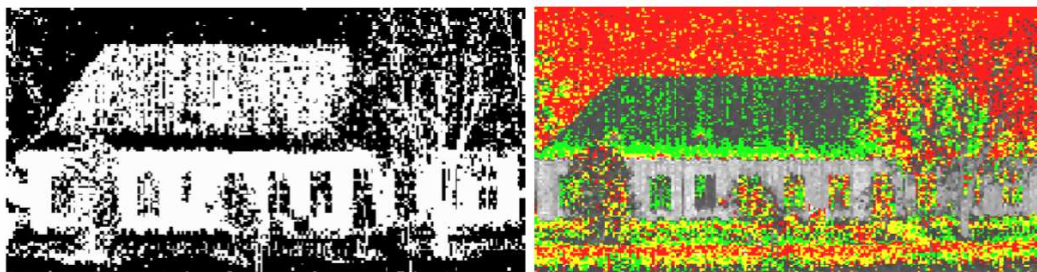


Figure 2.4: (Left) Pulses detected by standard peak method. Black: no pulse; White: pulse detected. (Right) A traffic light paradigm used to show faint echoes that are detected using this technique. Red: no pulse; Yellow: unknown; Green: weak echo. Source: Stilla et al. [160]

accuracy. They aim to fill gaps in partially occluded surfaces in an urban scenario with an assumption that the laser pulse hits a planar surface with a slope. Using a matched filter the surface response is modelled. Figure 2.3 illustrates their approach. First they estimate the interaction between the surface and laser pulse, *surface response*.

## 2.2.2 | Seeing Through Foliage and Clutter

When processing full-waveform LiDAR data to detect objects hidden behind foliage, the first echo may belong to canopy and the last return to the ground; but this is not always true. Hence, in order to confirm this, a 3D point cloud of the surveyed area is required. So far, we have seen literature that focuses on waveform modelling (transmitted and received) combined with surface modelling and methods that improve range/distance estimate for more accurate 3D point clouds and also echo classification in applications such as segmentation. Most of these methods do not consider the geometric properties of the surface except for [104] where the authors use the geometric properties to pre-segment occlusion and foliage.

Kirchhof et al. work is the closest we have found that aims to model and segment foliage at the 3D point level. Wagner et al. [176] have given a theoretical background into the interactions of a laser beam with the atmosphere, roofs, trees, power lines etc. Almost all the methods described so far neglect the mitigation of the laser beam when travelling through a tree volume.

Chehata et. al [47] present an approach highly influenced by the image processing community to segment urban scenes into different classes (vegetation, buildings, artificial surface and natural surface). Their features are a fusion of three optical components R,G and B, five multi-echo LiDAR components and four full-waveform components. The resulting feature vector  $f_v$  is given by:

$$f_v = [R \ G \ B; \Delta_z \ N_z \ R_z \ N_e; A \ \omega \ \sigma \ \alpha]^T \quad (2.2.2.1)$$

For the sake of completeness, we list the features in detail below:

- $R, G, B$ : 3 optical components, e.g., colors.
- $\Delta_z$ : height difference between the lidar point and the lowest point found in a large cylindrical volume whose radius can be set experimentally. This feature helps in discriminating ground and off-ground objects.
- $N_z$ : deviation angle of the local normal vector (a local plane needs to be computed) from the vertical direction. This is a strong feature that highlights the ground.
- $R_z$ : residuals of the local plane estimated in a small voxel (3D spatial boundaries, spherical or cylindrical). In general residuals are high for vegetation.
- $N$ : total number of echoes within the waveform for a LiDAR point. This feature will be high for vegetation and building facades.
- $N_c$ : normalized number of echoes obtained by dividing the echo number by the total number of echoes within the waveform of the current LiDAR point. Again, this feature highlights the vegetation.
- $A$ : echo amplitude. High amplitude values can be found on building roofs, on gravel, and on cars. Streets (asphalt or tar) have lower values. The lowest values may also correspond to vegetation due to higher attenuation.
- $\omega$ : echo width. Higher values correspond to vegetation since it spreads the lidar pulses. A narrow pulse is likely to correspond to ground and buildings. However, this is not the case with higher roof slope.
- $\sigma$ : echo cross-section, equals to  $A \times \omega$ . The values are high for buildings, medium for vegetation and low for artificial ground.
- $\sigma$ : echo shape describing how locally distorted the waveform is. Chauve et. al [45] shows that very low and high shape values correspond relatively to building roofs and vegetation.

Machine learning methodologies (support vector machines and random forests) have been applied to learn several representations using a set of training images and LiDAR signals to learn and classify data in to several classes ((vegetations, buildings, natural ground and artificial ground etc.). A similar approach using support vector machines (SVM) is proposed by [44]. Intuitively these methods are attractive and easy to implement but a detailed analysis on the influence of individual features is missing. Also, these approaches are not tested on objects hidden in dense forests and partially occluded.

With an efficient ground plane estimation, we can filter out remaining points into unnatural objects. We have identified methods [140, 143] that map, segment and classify various tree species. These methods segment single trees from the rest of the 3D data using dense full-waveform returns. They exploit a voxel based method to segment an entire tree and special tree saliencies (features) are computed to classify trees into two categories: deciduous and coniferous. Their approach is split in to three main steps:

1. *Waveform Modelling* Decompose a full-waveform signal into its individual components using *Sum of Gaussian* method similar to [176]. Extract 3D point clouds. Each point also bears additional information such as *Width*, twice the height standard deviation and *Intensity*, approximated as twice the standard deviation times the amplitude;
2. *Feature extraction* Features are made up of geometric  $x, y, z$  cues, internal geometry cues inspired by tree characterisation methods [125] and finally intensity information derived using the waveform decomposition step.
3. *Classification* This is a 2 step procedure that perform clustering of the tree species followed by a Bayesian classification.

Some of their segmentation results can be seen in Figure 2.5.

Results by Reitberger et al. [140] and others involved in tree segmentation and classification operate on a large dataset (forest areas spread over kilometers and sensing altitudes of 500 meters in altitude). When looking for targets under trees we will encounter dense

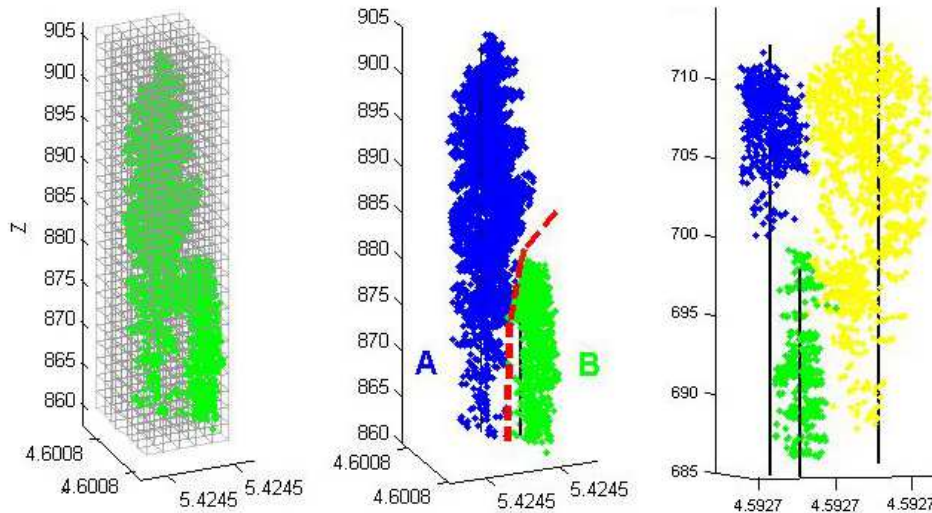


Figure 2.5: Some examples of tree segmentation using the methods presented above. (Left) A region of interest (ROI) into a voxel structure before processing. (Middle) Two segmented trees (A and B). (Right) Examples of segmented trees. Source: Reitberger et al.[143]

forest and targets which are relatively 10 to 15 times smaller and camouflaged. But such objects (e.g., vehicles) are mostly on the ground plane and extend to  $< 2 - 4$  meters above it. Recent work by the Swedish Defence Research Agency (FOI) [159, 78, 79] have proposed spatial filtering methods and algorithms that make use of object separation in 3D point cloud data in order to segment out objects for further recognition.

**Spatial Filtering** Grönwall et al. [79] propose a framework for object detection and classification. Their philosophy towards object (target) detection is a scheme that complements on two sub tasks: firstly, data reduction in order to identify everything except the target while preserving information that defines the target; secondly, point characteristics that define a particular class of target. Their approach breaks down non-target points in a scene into 3 logical categories: foreground clutter (outliers or points that do not belong to a homogeneous region), points above a certain elevation (height) and vegetation. They back this philosophy with a strong set (11) of assumptions. Although most of them are implicitly assumed in several cases (e.g., target being on the ground and ground surface being within the field-of-view) some of them are target specific (e.g., dimensions of the target to be known as prior and target being impenetrable). They

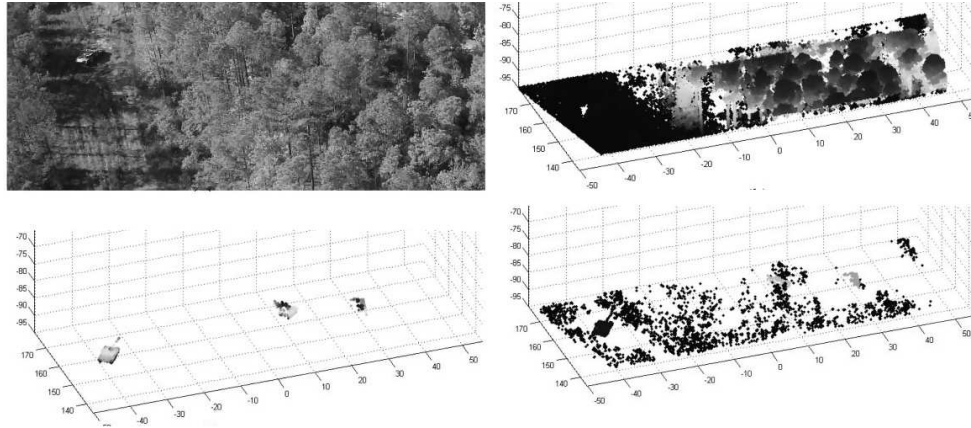


Figure 2.6: Sample results from [79] where authors have applied spatial filtering methods on point cloud data to separate targets from foliage and undergrowth.

make similar assumptions on the geometry of trees.

Combined with these assumptions and several ad-hoc features, e.g., elevation, tree trunks, clutter type A, clutter type B, planar surfaces, local convex regions and rectangular shadows, they test on real world LiDAR images and report their results. Their analysis shows that with the help of all 7 features they achieve a detection rate of 70% with a false alarm rate of  $< 0.015$ . Out of the 30 datasets, 70% were used for training and 30% for testing. Figure 2.6 illustrates a sample scene and the data used by Grönwall et. al [79]. The authors claim that they achieve better results from elevated sensors, e.g., drone or aerial, perspective compared to a ground-to-ground perspective. Their algorithm also suffers if the ground estimate is not accurate.

Their approach significantly reduce *clutter* (non-target points) is to have an efficient pre-processing step that cleans the data. Their list of 11 assumptions do not apply in an urban scenario. In a forest environment, the target shape can be more structured than the environment itself but this is not particularly true for an urban city scenario. Finally, this approach is evaluated only on geometric data. No spectral or polarisation data is considered.

**Jigsaw** The Jigsaw [113] is a program started in the year 2003 in collaboration with MIT, Sarnoff. Funded by The Defence Advanced Research Projects Agency (DARPA) and U.S. Army, the aim of the program was to support troop survivability by offering



robust situational awareness and accurate target identification. In a networked battlefield scenario, the Joint Commander Control is required to make a decision based on information that is made available. Using a light weight UAV equipped with LiDAR based sensors, a similar scenario can be imagined to support ground troops.

Jigsaw is a concept which is made up of a sequence of events which begins with a handover of a set of geo-locations (accurate with  $\pm 10$  meters), the military aircraft flies to the location with their in-house LiDAR sensor along with other set of sensors and collects 3D data. Their goal is for total latency in fifteen minutes, which includes data collection, processing and transmission from airborne platform to ground stations.

Their LiDAR sensor design is illustrated in Figure 2.7 along with the laser, detector array, Risley scanner and transmit-receive optics. This entire unit is manufactured to fit in a 30" diameter gimbal along with an Inertial Measurement Unit (IMU), a visible camera, and a long-wave infra-red (LWIR) camera. Some of the notable sensor parameters used are: Laser wavelength -  $532nm$ , laser pulse width -  $300ps$ , receiver aperture diameter -  $7.5cm$  and range resolution of  $40cm$ .

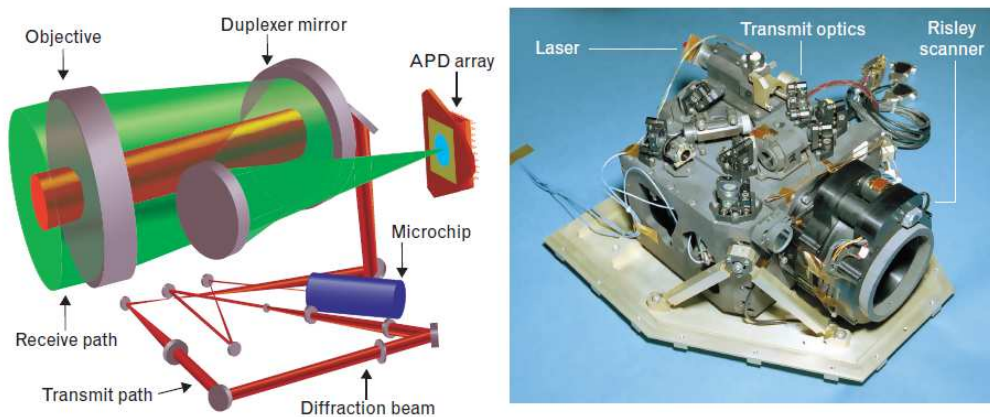


Figure 2.7: (Left) Optical path in the Jigsaw Sensor (Right) The sensors optical head, with all the components integrated. Source: Marino et al.[113]

;

## 2.3 | 3D Data Processing: Shape and Geometry

With the help of Sarnoff Corporation and Harris Corporation in the U.S. the Jigsaw team have created a suite of data processing algorithms. They convert raw data into a set of 3D  $(x, y, z)$  points and range histograms. Vaidyanathan et al. [172] use *Spin Images* [97] on a wide range of military vehicles and perform detection and recognition using machine learning. We present *Spin Images* in section 2.3 as part of geometric profiling of shape of objects.

**Object Detection and Classification** So far we have looked at methods and modalities that deal with 1D signals. We have looked at methods that extract objects (targets) from a 1-D backscatter waveforms. In this section, we present a review related geometric object modelling methods that surround *Partial Retrieval* of objects. Since data correspondence in a dense 3D point cloud is not fully solved and occlusion gives only a partial view of an object our focus has been towards *Partial Shape Retrieval* methods.

**Partial Shape Retrieval (Detection and Recognition)** We define partial retrieval as the goal to detect and recognise a moving or stationary object of interest. We refer “partial” here (in a 3D shape) meaning that it is composed of one or more semantic shapes and can be characterised by a sub-set of shapes belonging to ‘object’. No assumption is made that the set is complete and is a full-shape, it can be relative. A tank turret is part of the tank shape and the barrel a part of its shape. Its relevance is application dependent. If a search analogy is taken, then we are interested in searching for a *part* (‘sub-set’) in the *whole* object (‘set’). We can consider this as an ‘object-to-object’ search but with similarity measures that are based on partial matches.

Partial shape based retrieval methods need to address address difficulties such as: partial matches, partial correspondence and shape matching and partial similarity measures. We categorise partial retrieval methods based on local and global saliencies:

1. Point based descriptors

2. Segmentation based methods
3. View based methods
4. Statistical shape modelling

For the sake of brevity and scope of this document we will summarise (i) Point based descriptors and (ii) Segmentation based methods. Point based descriptors are most relevant for partial retrieval of objects in dense point clouds. We present some segmentation methods that have been successful to segment and recognise whole objects in computer aided design (CAD) like datasets.

Some of the most widely used point based features are *Surface Normal and Curvature* [6]. These do not discriminate underlying surfaces very well except when the scale chosen is very small and the data is dense. These are local features that characterise a point and the region around it. Previous work related to point features are: Moment Invariants [150], Integral Volume Descriptors (IVD) [70], Spherical Harmonic Invariants (SHI) [34] and a few shape correspondence methods. These are translation and rotation invariant but they do not discriminate the surface they are part of. Additionally, when the data is sparse; and when scale needs to be chosen in order to compute these features. These are points with a 'single-value' point-descriptors and may not be expressive enough and their values highly depend on the sensor and noise. Multi-point descriptors such as Curvature Maps [174] and Spin Images [97] were proposed originally for 3D meshes and have better local characterisation. But these methods do not deal with partial view and occlusion very well.

Methods such as Extended Gaussian Images (EGI) [90] describe the object on a unit sphere and tend to have problems handling arbitrary curved objects. Eigen-shapes [36] show impressive results, but most of their data is CAD models and is tested on synthetic data. Spherical harmonic invariants (SHI) and spin images work well in dense point-cloud datasets, but their performance seems to degrade for noisier and sparser datasets [21]. Moment Invariants, SHI and IVD are successfully used in point cloud registration, but they do not encode the 'surface-type' information which is highly desirable but are not

robust against occlusion.

**Point based Descriptors** This review shall focus on some of the below:

- How are surfaces defined and how to define parts?
- Is parametrising them the correct approach?
- Should objects be considered in their entirety or point based features suffice?
- Should segment-based object detection approach be used? [19]
- Single value features (e.g., curvature, normal) or multi-value features (curvature maps, spin images)?
- Should we handle part organisation?
- What similarity measures to use?

A point descriptor is usually a scalar or a vector that is a function computed around an individual point that captures certain information, e.g., Gaussian curvature. Here we discuss some of the point based descriptors recently proposed: Spin Images [97], Local Spherical Harmonics Descriptor [102], Heat Kernel Descriptor [58] which is based on the fundamental solution of the heat equation [161] and a variant of Spin Images (Point Feature Histogram [149] and View Feature Histogram [148]) that are popular with the robotics and machine learning community.

**Spin Images** Spin images is a technique proposed by Johnson et al. [97] that maps all the vertices in  $\mathbf{R}^3 \mapsto \mathbf{R}^2$  based on an oriented point basis  $(p, n)$ . In other words  $(x, y, z)$  are mapped in to coordinates  $(\alpha, \beta)$ . Figure 2.8 illustrates the basic idea. Here  $\alpha$  is the perpendicular distance to the normal line  $L$ , and  $\beta$  is the signed perpendicular distance to the tangent plane  $P$ .

Once the basis formed, the neighbours are projected onto a cylindrical coordinate system, and a 2D image is generated which represents a density histogram of points with respect

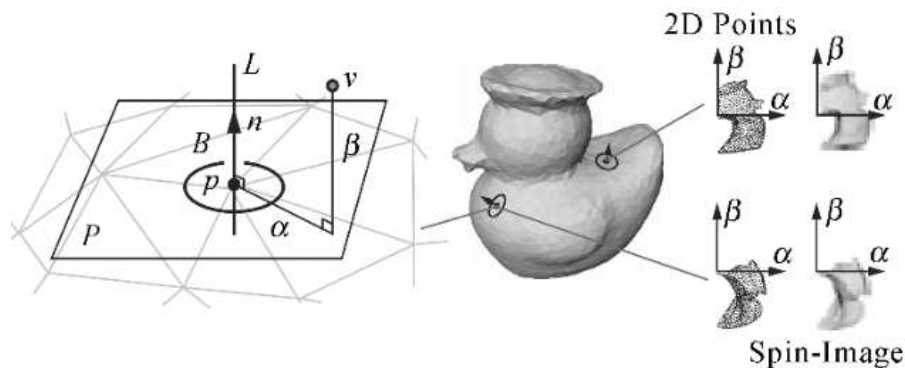


Figure 2.8: (Left) Spin image and its local coordinate system. (Right) 3D model and 2 sampled spin images. Source: Johnson et al. [97]

to the basis. By embedding a 3D shape as a manifold, pair-wise geodesic distances are preserved which makes the descriptor robust to isometric transformations. Recent work by Sipiran et al. [171] show that partial shapes can be detected using spin images, but as part of this work we have tested their approach on synthetic CAD style data with high point density.

**Local Spherical Harmonics Descriptors** Kazhdan et al [102] first proposed a Spherical harmonic descriptors (Global) for a 3D model. The descriptor is constructed on several concentric spheres centring at the mean point. The same concept can be applied locally to capture local information. Local spheres with different radii are centred around a point of interest and a binary identification function is defined in order to evaluate intersections and non-intersection relationship between each sphere and the underlying mesh. Using different frequency vectors, the spherical function is decomposed into harmonic functions. This can be used to analyse local surfaces [13]. The descriptor is robust to rotation and translation transformations but is susceptible to non-rigid deformations. When looking for rigid objects (e.g., vehicles and buildings) this approach can be robust.

**Diffusion Geometry & Heat Kernel Descriptor** . This work has received a lot of attention since published in 2010 by Ovsjanikov et al. [130]. The idea here is to find structure-preserving maps between shapes. Heat kernel descriptors are derived from the

heat diffusion equation by using the Laplace-Beltrami operator on manifold. Laplace-Beltrami operator, in its generality, is a Connection Laplacian, or a differential operator acting on several tensors of a manifold. This approach can extract non-rigid features on surfaces and any volumetric representation. The paper gives convincing results (Figure 2.9) in partial extraction and detection, e.g., human arms, horse legs and elephant trunks. Due to its intrinsic property, heat kernel is insensitive to isometric deformation and robust to fine variations. One of the drawbacks to this approach is that the scale needs to be normalised as it cannot handle different scales of a shape.

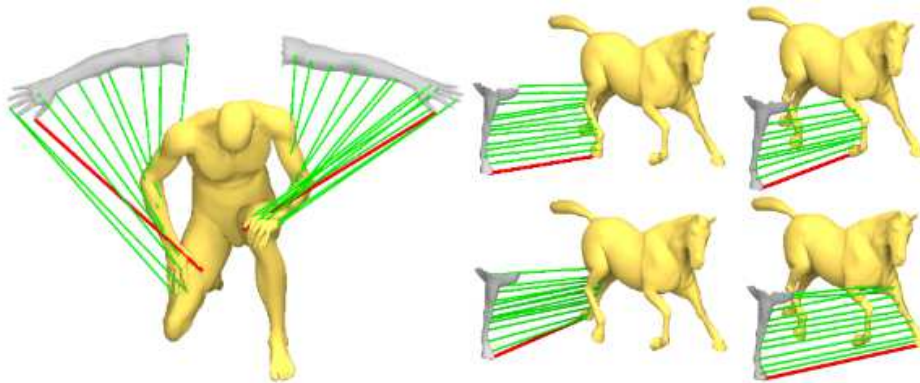


Figure 2.9: Partial matching using a heat kernel descriptor. Source: Ovsjanikov et al. [129]

**Other point based features** Partial similarity can also be computed at point level. We study this by looking at methods and approaches applied to indoor robotics applications. We discuss the *Point Feature Histogram* [149]. PFH is an informative pose-invariant local features which model underlying surface model properties. The dissimilarity between the descriptors should be high enough to categorise them into different surfaces or shapes. The computation procedure is similar to spin images [97] and is based on the combination of a geometric relation between a point  $p$  and its nearest  $k$  neighbours. A basis coordinate system is fixed which incorporates a 3D point  $\langle x, y, z \rangle$  and its surface normals  $\langle nx, ny, nz \rangle$ . The author also suggests the use of other properties such as curvatures and 2nd order moments.

**Generic shape representations** Csakany and Wallace propose an approach in [55] to segment and classify 3D objects by representing an object by generic shapes. This work addresses the problem of representation and classification of 3D-objects from depth data or meshed data. Firstly, they segment surface patches on the basis of curvature and quadratic surface fitting. Secondly, they are represented by a modified Gaussian image which includes shape based indexes. Finally, they perform learning and measure the similarities between feature sets of objects and generic classes. This framework is tested on a group of 3D objects built using CAD models and laser scanned depth data.

There are few drawbacks to this approach. Representing objects with a set of generic shapes (quadratic surface) is going to work on organised point clouds or dense CAD style datasets. Another implicit assumption in this work is made on the wholeness of the objects and this approach has not been tested under occlusion or even sparse point clouds. In the case of objects hidden behind dense foliage, we have to deal with problems such as: occlusion, missing points, shadows and ghost points.

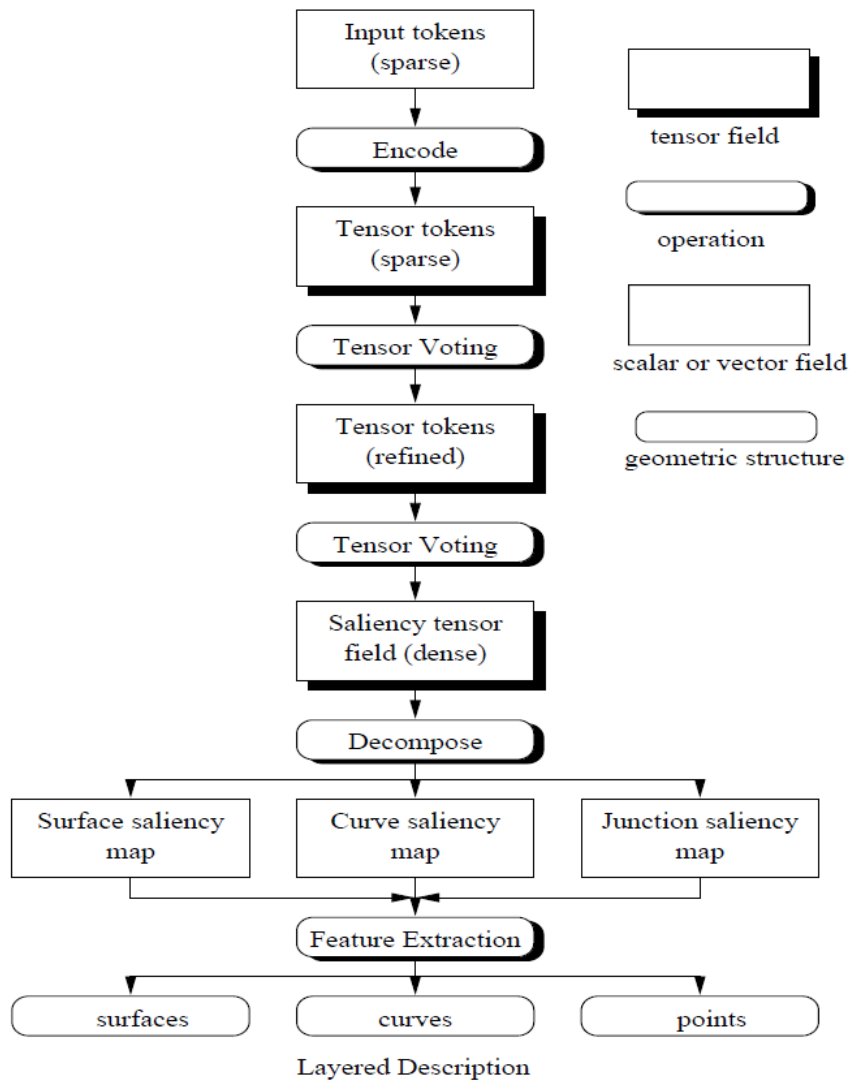


Figure 2.10: Tensor Voting Framework as proposed by Medioni et. al [123].

**Voting based Schemes** LiDAR point cloud data are often noisy and identifying salient and structured information is a challenging task. In order to estimate these salient structures a voting scheme methodology has to be robust to noise and handle data communication between points in a local neighbourhood. The goal is to extract geometric structures such as, regions, curves, surfaces, planes, and their interactions from organised or un-organised LiDAR data sets. Below we present two such voting schemes: The Hough Transform, originally proposed by R.O. Duda in [62] and The Tensor Voting framework by Gérard Medioni et. al in [123].



**Hough Transformation (HT)** HT is a method used in image processing and computer vision for detecting lines and circles in 2-dimensional data sets. So far it has not received a lot of attention for 3D. High computation costs have led to several variations for HT. Borrmann et. al [28] present several variations (Generalised 3D Hough Transform, Probabilistic Hough Transform) and its application to detect planes in 3D data. They evaluate new accumulators on synthetic 3D data sets.

## 2.4 | Sparse Representation & Dictionary Learning

Sensors, both natural (e.g. eyes or ears) and artificial (e.g. 2D or 3D imagers) are capable of recording huge amount of data every second. The underlying process that causes such observations are usually low-dimensional compared to the recorded datasets. In this chapter, we present techniques that can represent a signal of length  $P$  with only  $K \ll P$  nonzero coefficients. The extraction and identification of such relevant information (generating cause) within the classes of Sonar and LiDAR signals is the central topic of this thesis. However, the techniques presented here can be adopted and applied to other sensing modalities (e.g. 1D, 2D or 3D datasets). These techniques have been studied for over a decade and have been highly beneficial for signal reconstruction, compression [33], approximation [4] and discrimination [52, 116, 191].

### 2.4.1 | Categorisation of Sparse Recovery Algorithms

Compressive sensing theory asserts that a certain set of signals can be recovered accurately using fewer measurements than the Nyquist/Shannon sampling principle. Figure 2.11 illustrates the three main processes involved: sparse representation, measurements (encoding), and sparse recovery (decoding).

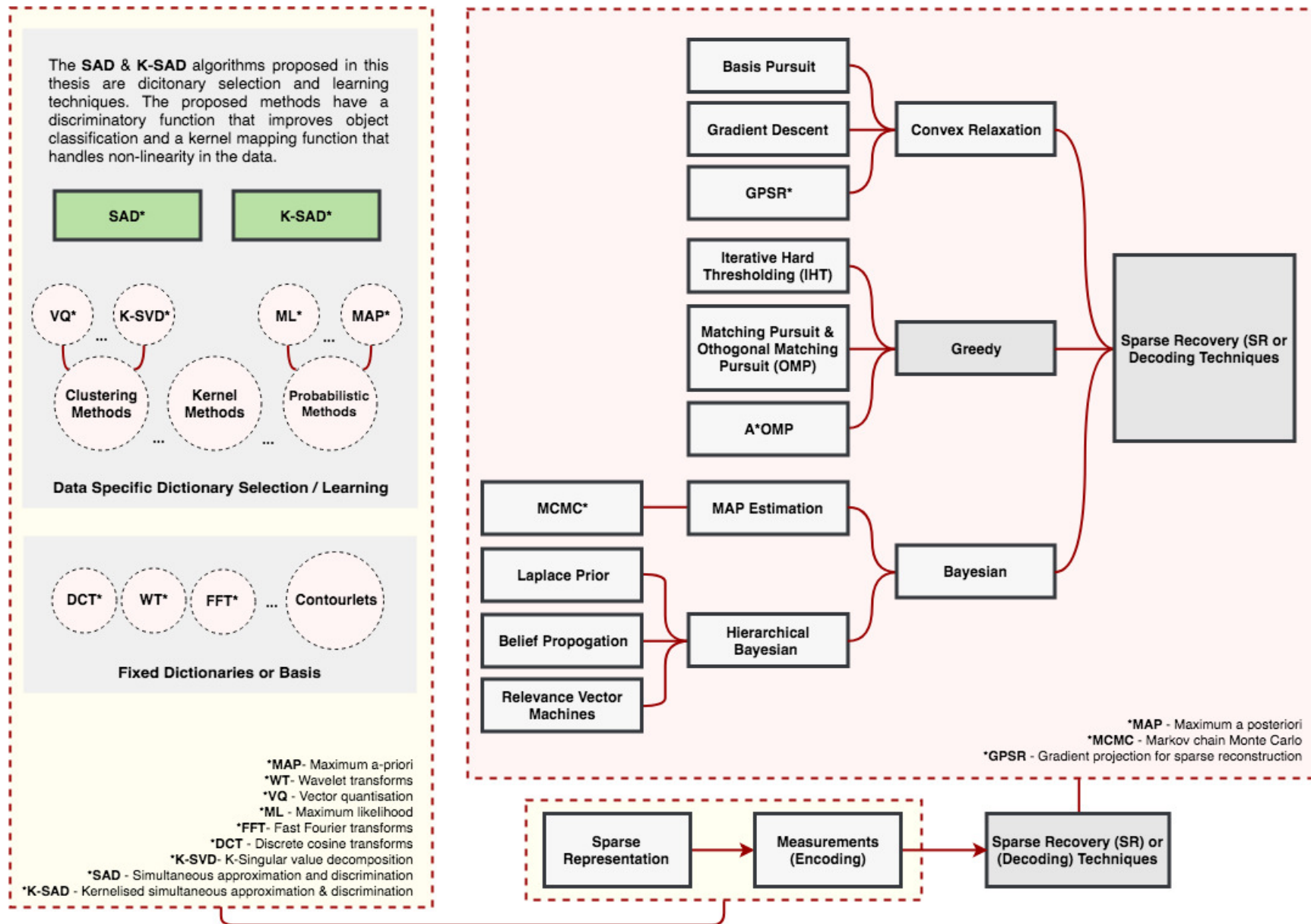


Figure 2.11: Compressive sensing process and a high-level categorisation of sparse recovery algorithms.

**Sparse Representation** - The signal representation model proposed here is based on the recent developments in approximation, dimensionality reduction and compressive sensing, which have an underlying concept, *sparsity*. Sparsity, also referred to as “compactness” means that an original signal can be represented using a contribution vector and a finite basis, i.e. the contribution vector is almost sparse [38]. Given an approximate dictionary, an orthogonal or bi-orthogonal basis, which can be fixed or learnt from the data the signals or images of interest can be represented sparsely. A detailed discussion on orthogonality can be found in [147]. A signal  $x \in \mathbb{R}^N$  is said to be  $K$ -sparse, if  $K$  elements of its entries are non-zero.

**Coefficients as features**- A transformation, through ad-hoc feature extraction combined with sparsity based methods can offer much better results. The feature extraction stage is a way of transforming or representing the raw data by using a reduced set of data points or parameters. Most signals are either naturally sparse, or can be made sparse in a particular basis using specific transforms, e.g., the Discrete Cosine Transforms (DCT) or directional and analytical Wavelets. Regardless of the choice of such bases, when learnt, the resulting sparse code, or the *coefficients* can be treated as a feature representation.

More recently, it has been shown that a sparse representation can also be learnt from the data itself [4, 188, 186]. A detailed review with historical and current state-of-the-art dictionary learning and design methodologies can be found here [134, 147]. Sparse representation has a significant impact on the discriminatory nature of any classification system. Dictionary learning (DL) for signal approximation and discrimination is equal to identifying, given a set of training samples, an appropriate set or dictionary such that any  $K$ -subset of it spans a  $K$ -dimensional subspace. In contrast to hand-crafted dictionaries, DL methods adapt an over-complete or orthogonal dictionary to an observation, hoping for better sparsity.

The discriminative nature of these sparse coefficients may not always be natural, but it can be discovered by carefully selecting and clustering the coefficients under certain constraints. Experiments in chapter 7 show that these methods on their own cannot accurately discriminate between different objects, e.g. underwater mine-like-objects (MLOs).

Similarly, in order to take into account multi-resolution, invariance to rotation and scale, using raw signals alone, produces unsatisfactory classification performance.

**Measurement (Encoding)** - The measurement process consists of taking only a few measurements  $y \in \mathbb{R}^P$  from the sparse signal  $x \in \mathbb{R}^N$ .  $y$  can be approximated with a simple multiplication of the sparse signal  $x$  by a dictionary matrix  $\mathbf{D} \in \mathbb{R}^{P \times N}$ , where  $K$  is the number of measurements, and  $N$  is the dimension of the sparse signal with  $K \ll N$ . Ideally, the reduction (compression) from  $\mathbb{R}^N$  to  $\mathbb{R}^K$ , must preserve the underlying generating cause (information) stored in the  $K$ -sparse signal necessary to recover the original signal from these measurements, i.e. a dictionary matrix.

**Sparse Recovery (Decoding)** - The sparse recovery process aims to recover the sparse signal  $x$  from a small set of measurements of  $y$  when  $x$  is projected on a measurement or dictionary matrix  $\mathbf{D}$ . The sparse recovery problem is an underdetermined system of linear equations and the uniqueness of the solution is guaranteed when the signal is sufficiently sparse and the measurement or dictionary matrix satisfies the restricted isometry property (RIP) [50]. Over the last decade, several sparse recovery algorithms have been proposed. These algorithms can be classified into three main categories: Greedy [166, 101, 60, 182, 25], Bayesian [95, 14, 17] and Convex & Relaxation [50, 69, 66].

## 2.4.2 | Dictionary Learning

Given a set of examples  $\mathbf{Y} = [y_1, \dots, y_N] \in \mathbb{R}^{P \times N}$ , the aim is to extract a sparse matrix,  $\mathbf{X} = [x_1, \dots, x_K] \in \mathbb{R}^{N \times K}$ , and learn a dictionary  $\mathbf{D} = [d_1, \dots, d_K] \in \mathbb{R}^{P \times K}$ , simultaneously. Traditional DL algorithms have two steps:

### 1. Sparse Coding:

$$\min_{x_i} \|y_i - \mathbf{D}x_i^T\|_2^2, \text{ s.t. } \|x_i\|_0 \leq T_0, \forall i = 1, \dots, N \quad (2.4.2.1)$$

## 2. Dictionary Update:

$$\min_{d_j} \left[ \|\mathbf{Y} - \sum_{j=1}^K d_j x_j^T\|_F^2 \right], |d_j| = 1, \forall j = 1, \dots, K \quad (2.4.2.2)$$

Equation (2.4.2.2) assumes that both  $\mathbf{Y}$  and  $\mathbf{D}$  are fixed except in column  $d_j$  and the coefficients that correspond to it, the  $j^{\text{th}}$  row in  $\mathbf{X}$ , denoted as  $x_j^T$ .  $\|\cdot\|_0$  is the sparsity measure. The unit length and orthogonal constraint,  $|d_j| = 1$ , makes the dictionary  $\mathbf{D}$  orthonormal.

**Over-complete vs Orthogonal Dictionaries** - Learning an over-complete dictionary using greedy methods ensures maximum sparsity, but the dictionary can be most coherent, i.e. highly redundant. Enforcing the incoherence condition on an over-complete dictionary [38] whilst solving (2.4.2.2) is a difficult task. In this work, we learn a structured orthogonal dictionary and use the derived coefficients as target signatures for classification purposes.

**Non-linearity in the data** - Most real-world signals or images have an intrinsic non-linear similarity measure and can be harder to discriminate. Recently, Kernel based DL [189, 96, 173] methods have been proposed as an effective way of capturing non-linearity in the input space and learn sparse encodings, simultaneously. However, these methods require the kernel or Gram matrix,  $\Phi \in \mathbb{R}^{N \times N}$ . For large-scale datasets, computing such a matrix is a computationally complex task, both in space and time. Work by Golts and Elad [72] incorporate an effective way of approximating a kernel matrix with an over-complete dictionary and Gangeh *et. al* [68] show that an kernelized orthogonal dictionary can be learnt.

## 2.5 | Anomaly Detection

Anomaly detection (AD) is the process of detecting data points that are inconsistent with the distribution of the majority of the data. AD has had a long history of research in machine learning, statistics and data mining. Anomalies are also known as abnormal

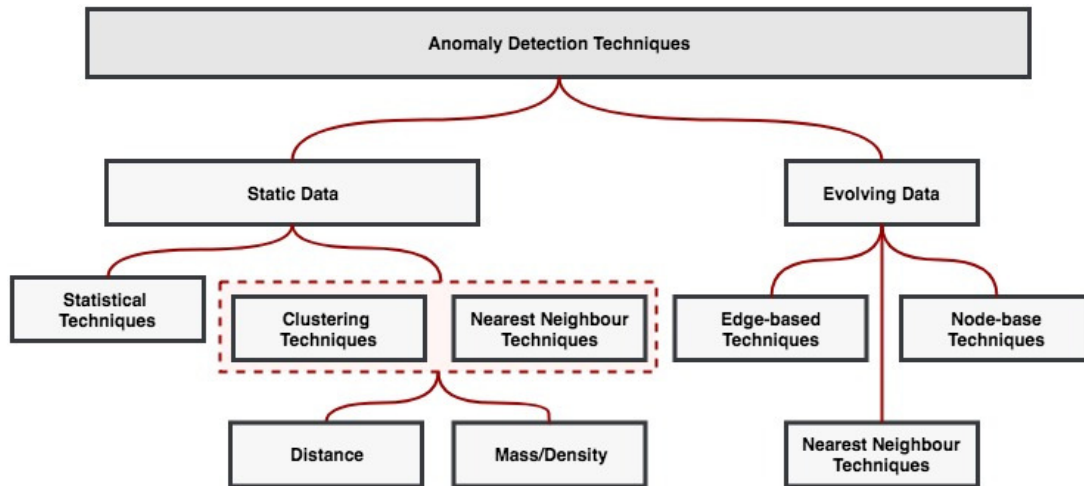
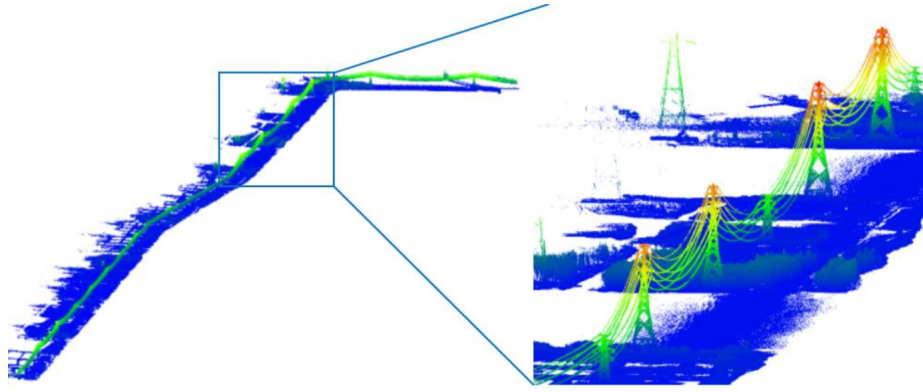


Figure 2.12: In this work we present a high-level categorisation of anomaly detection algorithms.

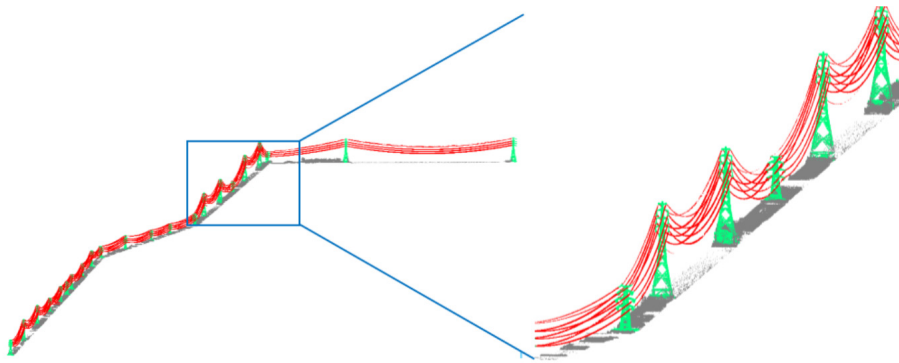
events, abnormalities, deviants or outliers. Detailed AD surveys on static data for various scenarios have been proposed [42, 3, 43, 106]. Static datasets are a known complete set of observations that are available and the anomalies are detected with regards to the entire dataset. Figure 2.12 illustrates a high-level categorisation of AD techniques for static and evolving data. In our survey, we have left out evolving datasets [73, 92, 151]. The unsupervised scenario, applying AD techniques to data streams can be divided into two main groups: statistical methods and clustering or nearest-neighbour methods.

The *statistical* based methods learn distributions that resemble *normal behaviour*. A test set is labelled abnormal if it does not fit into the current model (or has a lower probability score). Gaussian mixture model (GMM) [164, 71] and Bayesian approaches [122] apply a outlier probability score to data streams (signals or images).

The *clustering* or *nearest-neighbour* techniques label data points as anomalous if they fall in to clusters which have a lower mass (density) or small number of data points. The balanced iterative reducing and clustering using hierarchies (BIRCH) [192] is a popular technique introduced in 1996 and several variants [169] have been proposed and analysed since then. The nearest-neighbour technique scores outliers based on a distance metric, either Euclidean, Mahalanobis or in combination with cluster density [56]. Recently, a distance based outlier detection algorithm called LEAP [40] has been proposed. This



(a) Sample dataset from [49] where authors treat power-line detection problem as an anomaly detection problem.



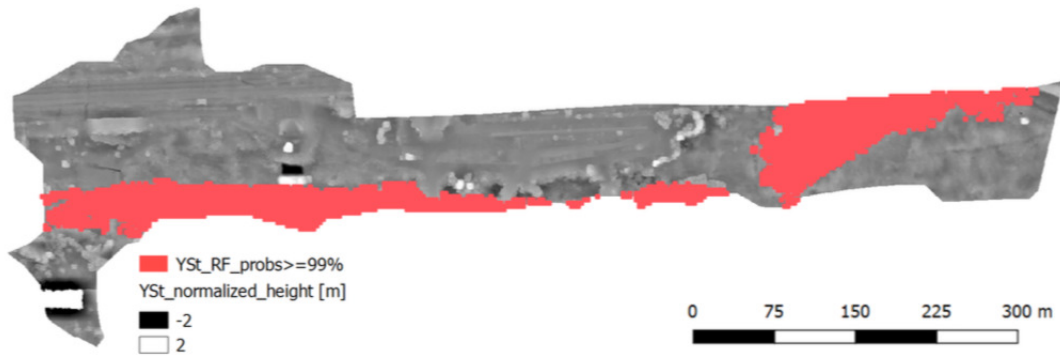
(b) Detected power-lines coloured in red.

Figure 2.13: Treating the power-line detection in LiDAR data as an anomaly detection algorithm. Source [49]

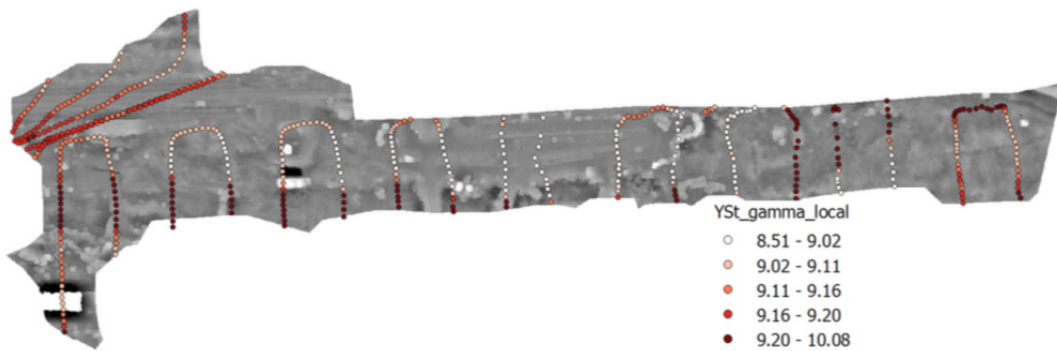
method scales well for large datasets by optimising its search space. In our work, we borrow the idea of sliding window on data streams, not to compute global or local distances but to learn correlations across tree layers (See chapter 4).

**Anomaly Detection (AD) in LiDAR Data** In this thesis we present an approach that learns the abundance of distinctive material spectra and their relationships across different tree layers within a range(depth)-amplitude data series. Within the context of FW-LiDAR, our aim is to detect a deviation from the learnt models resulting in an abnormal event.

Over the last decade, AD techniques has been applied to 3D point cloud data. More recently for: UAV right-of-ways (RoW) clearance management [49] and nuclear disaster monitoring [30], space LiDAR for environmental monitoring [57, 163], mining and tunnel



(a) Normalised height map and gamma spectrometer measurements results. Source [30].



(b) Normalised height map and classification results using Random Forests (RF).

Figure 2.14: Detecting abnormal regions at the Chernobyl Nuclear site. Source [30].

change detection [181].

In [49], authors proposed an anomaly detection algorithm to detect power lines in 3D point cloud data. Their approach models power lines as splines and after a series of filtering algorithms scores the *non-terrain* point-cloud data. The successful candidates are then segmented using a curve fitting algorithm. UAV clearance or RoW is then treated as an anomaly detection problem. Figure 2.13 illustrates example images with power-lines segmented. Recent work by Briechle et. al [30] have applied anomaly detection and classification techniques to map contaminated areas around Chernobyl nuclear site. After the 1986 explosion, radioactive fall-out and contaminated trees (the *Red Forest*) were buried in the Chernobyl's exclusion zone (ChEZ). Today, three decades later, the exact location of these sites and materials is needed. The authors use aerial LiDAR, multi-spectral camera and gamma-spectrometry data collected using an UAV. In the paper, authors use established approaches to feature engineering on 3D LiDAR data and detect anomalies in regions around the nuclear site. Figure 2.14b illustrates their anomaly



classification results using Random Forests (RF) in the study area 3: Yanov Station (YS)

3.3. The classification of class ‘contaminated’ is achieved by fusing results from the gamma spectrometer measurements and normalised height maps (generated using the 3D point cloud data).

# Chapter 3

## Dictionary Learning & Non-linear Data Approximation

### 3.1 | Simultaneous Approximation & Discrimination

We present an optimisation and discrimination algorithm that can re-order and select a subset of dictionary atoms from a larger subspace, *sparse discriminative signatures* (Section 3.1.1). The resulting over-complete dictionary and their contribution (coefficients) vectors are then used as features for object classification. The ordering and coefficients of the dictionary atoms differ for each target class so a supervised dictionary learning approach is formulated using training samples and a cross-validation test is conducted.

The goal here is to find an optimal dictionary, such that an input dataset,  $\mathbf{Y}$ , can be accurately reconstructed and the size of this subspace is as small as possible. One may take a simpler approach, randomly and uniformly selecting basis vectors to build the subspace. This strategy may be risky as it may not capture the variability in the training data for different target classes. A principled selection strategy that produce highly discriminative multi-class dictionary, which selects an optimal subset of  $\mathbf{D}$  is proposed. We call this approach *Sparse Discriminative Signatures* (SDS). We then show how SDS can be used, in conjunction with a K nearest-neighbour approach, for signal classification. We call this combined approach *simultaneous approximation and discrimination* (SAD).

### 3.1.1 | Sparse Discriminative Signatures (SDS)

For  $N$  training samples, with signal dimensionality  $P$ , let  $\mathbf{Y} \in \mathbb{R}^{P \times N}$  correspond to the signal matrix. The representation model in this work assumes that  $\mathbf{Y}$  can be approximated with a combination of vectors (*basis vectors*), from an over-complete dictionary, (a *dictionary is over-complete when the number of basis vectors is greater than the dimensions of the input*),  $\mathbf{D} \in \mathbb{R}^{P \times K}$ , either built using pre-selected functions or learnt from the data. A corresponding coefficient matrix  $\mathbf{X} \in \mathbb{R}^{N \times K}$  controls the contribution of each individual atom of  $\mathbf{D}$ . Thus, building on the model as shown in (3.1.1.1), in its simplest form, the representation model can be re-written:

$$\mathbf{Y} \approx \mathbf{D}\mathbf{X}^T \quad (3.1.1.1)$$

Using all the column vectors of  $\mathbf{D}$ , denoted by  $\{d_k\}_{k=1}^K$  and vectors of  $\mathbf{X}$ , denoted by  $\{x_k\}_{k=1}^K$ , a signal  $\mathbf{y}_n$  can be represented as a sparse linear combination of these columns. The representation of  $\mathbf{Y}$  may either be exact ((3.1.1.1)) or an approximation, such that,  $\|\mathbf{Y} - \mathbf{D}\mathbf{X}^T\|_2 \leq \epsilon$ . For a full-rank matrix  $\mathbf{D}$  and  $K \ll N$ , there are an infinite number of solutions to this representation ((3.1.1.1)).

When solving (3.1.1.1) using iterative methods, a stop criteria is needed, e.g., a sparsest approximation constraint, i.e.  $\min_{\mathbf{X}} \|\mathbf{X}\|_0$ , where  $\|\cdot\|_0$  is the  $l_0$  quasi-norm, which counts the non-zero entries of a vector, or when the error term  $\epsilon$  stops decreasing and goes below a threshold. An optimal  $\mathbf{D}_{opt}$  and  $\mathbf{X}_{opt}$  can be defined by minimising the following objective function:

$$\{\mathbf{D}_{opt}, \mathbf{X}_{opt}\} = \arg \min_{\mathbf{D}, \mathbf{X}} [\|\mathbf{Y} - \mathbf{D}\mathbf{X}^T\|_2^2 + \beta_1 \|\mathbf{X}\|_0], \quad (3.1.1.2)$$

where the parameter  $\beta_1 > 0$  is a regularization scalar that balances the trade-off between the reconstruction accuracy and sparsity.

Finding a solution to ((3.1.1.2)) is an NP-hard problem primarily due to its combinatorial optimisation, i.e., minimising for the best reconstruction with the least number of

basis functions that are individually convex, but not jointly convex. This can be solved iteratively. A suboptimal solution to this problem is found by replacing the  $l_0$  norm in ((3.1.1.2)) with the  $l_1$  norm [53] and rewriting ((3.1.1.2)) using components of  $\mathbf{X}$  as:

$$\arg \min_{\mathbf{X}, \mathbf{D}} \left[ \|\mathbf{Y} - \mathbf{D}\mathbf{X}^T\|_2^2 + \beta_1 \sum_{k=1}^K \|x_k\|_1 \right] \text{ subject to,} \\ \|x_k\| \leq 1, \forall k = 1, 2, \dots, K. \quad (3.1.1.3)$$

where  $\|\cdot\|_1$  is the  $l_1$  norm. Equation 3.1.1.3, in its current form is not suited for learning multi-class dictionaries. The minimisation problem is augmented with a discriminative function that provide higher degree of separation between the dictionary atoms. SDS (Algorithm 1) provides a solution to the modified objective function and re-ranks the dictionary atoms.

**Discriminative Coefficients** The sparse solution ((3.1.1.3)) can be updated with a discrimination function parametrised by  $\mathbf{D}_{opt}$  using a Fisher discriminant function which has been very popular in Linear Discriminant Analysis (LDA), a generalisation of Fisher's linear discriminant [155]. The discrimination power is maximised when the spatial distribution of similar classes (scatter matrix,  $S_w$ ) has samples that are closer to each other when compared to samples from different classes (scatter matrix,  $S_b$ ). This is used to re-rank dictionary atoms in order to achieve maximum discrimination.

For a set of samples or coefficients,  $\mathbf{X}$ , a given dictionary matrix,  $\mathbf{D}$ , we have  $c$  different classes or groups. Let  $N_i$  be the number of samples in class  $i$ . Each class group,  $\Omega_i$ , has a class mean, which we denote  $\bar{x}_i$

$$\bar{x}_i = \frac{1}{N_i} \sum_{j=1}^{N_i} x_{i,j} \quad (3.1.1.4)$$

where there are  $N_i$  data points (coefficients) in class  $\Omega_i$ .

The grand or overall mean of all coefficient samples can be written as:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^c N_i \bar{x}_i = \frac{1}{N} \sum_{i=1}^c \sum_{j=1}^{N_i} x_{i,j} \quad (3.1.1.5)$$

---

**Algorithm 1: SPARSE DISCRIMINATIVE SIGNATURES (SDS)**

---

**Input:**  $\mathbf{D}$ , a dictionary matrix in  $\mathbb{R}^{P \times K}$ ,  
 $y$ , a signal in  $\mathbb{R}^P$ ;  $\beta_1, \beta_2$ , the regularisation parameters  
 $\epsilon$  or  $s$ , a stop criteria - reconstruction error or iterations

**Output:**  $x$ , a sparse representation of signal  $y$  in  $\mathbb{R}^K$ ,  
 $I$ , support set of the estimated signal, i.e. a set containing positions,  
of non-zero elements of  $x$

1 Initialise sparse representation  $x_0 = 0$ , the index set  $I_0 = \phi$ , matrix of chosen atoms  $\mathbf{D}_0 = []$ , and the iteration counter  $t = 1$ ;

2 **while**  $t < s$  or  $r < \epsilon$  **do**

3     Calculate the residual:

$$r_t = y - D_{t-1} x_{t-1}^T$$

4     Find the index of the column of  $\mathbf{D}$ , that maximises the objective function:

$$i_t = \arg \min_{\mathbf{d}_i \in \mathbf{D}} F(d_i^T r_t, \beta_1, \beta_2)$$

if max occurs for multiple indices, choose one arbitrarily

5     Augment the index set:

$$I_t = I_{t-1} \cup \{i_t\}$$

and the matrix of chosen atoms:

$$D_t = [D_{t-1} D_{i_t}]$$

6     Update signal estimate:

$$x_t = D_t^{-1} y$$

7     Update iteration counter:

$$t = t + 1$$

8 Assign:

$$x = x_t; I = I_t;$$

**Return:**  $x, I_t$

---

---

**Algorithm 2:** SPARSE APPROXIMATION & DISCRIMINATION (SAD)

---

**Input:**  $\mathbf{Y}_{train}$ , a training dataset in  $\mathbb{R}^{P \times N}$ ,  
 $\mathbf{Y}_{test}$ , a test dataset in  $\mathbb{R}^{P \times M}$ ,  
 $I$ , support set of the estimated signal,  
 $L_{train}$ , label set for  $\mathbf{Y}_{train}$

**Output:**  $L_{predict}$ , predicted label set for  $\mathbf{Y}_{test}$ ,  
 $\mathbf{C}$ , confusion matrix for ROC analysis

- 1 Set  $\epsilon$ , error threshold,  $s$ , no. of iterations,  $\beta_1, \beta_2$ , the regularisation parameters
- 2 Select a discriminative dictionary for training set  $\mathbf{Y}_{train}$  using Algorithm 1:

$$[\sim, I_t] = SDS(\mathbf{D}, \mathbf{Y}_{train}, \epsilon, s, \beta_1, \beta_2);$$

$$\mathbf{D}_{opt} = \mathbf{D}[I_t, :]$$

- 3 Generate coefficient matrix for  $\mathbf{Y}_{test}$ :

$$\mathbf{X}_{test} = \mathbf{D}_{opt}^\dagger \mathbf{Y}_{test}$$

- 4 Initialise the test label set to null:

$$L_{predict} = \emptyset$$

- 5 Generate similarity matrix for classification

$$S_{test} \leftarrow MakeSimilarityMatrix(\mathbf{X}_{test})$$

- 6 Get  $L_{predict}$  using labels  $Ind_{test}$  of the closest matches based on k-NN Classification:

$$Ind_{test} \leftarrow SortSimilarityMatrix(S_{test})$$

$$L_{predict} \leftarrow GetMajorityLabel(Ind_{test})$$

- 7 Make confusion matrix - GenerateConfusionMatrix( $L_{org}, L_{predict}$ )

**Return:**  $L_{predict}$

---

We can also define a class group covariance matrix:

$$\Sigma_i = \frac{1}{N_i - 1} \sum_{j=1}^{N_i} (x_{i,j} - \bar{x}_i)(x_{i,j} - \bar{x}_i)^T \quad (3.1.1.6)$$

The *intra-class* scatter matrix,  $S_w$ , can be defined as:

$$S_w = \sum_{i=1}^c (N_i - 1) \Sigma_i = \sum_{i=1}^c \sum_{j=1}^{N_i} (x_{i,j} - \bar{x}_i)(x_{i,j} - \bar{x}_i)^T \quad (3.1.1.7)$$

The *inter-class* or between class scatter matrix,  $S_b$ , can be defined as:

$$S_b = \sum_{i=1}^c N_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T \quad (3.1.1.8)$$

Finally, the Fisher discrimination function is defined as:

$$G(\mathbf{X}) = S_w^{-1} S_b, \quad (3.1.1.9)$$

Using (3.1.1.9), (3.1.1.3) is updated:

$$\arg \min_{\mathbf{X}, \mathbf{D}} \left[ G(\mathbf{X}) + \beta_1 \sum_{k=1}^K \|x_n\|_1 + \beta_2 \|\mathbf{Y} - \mathbf{D}\mathbf{X}^T\|_2^2 \right], \quad (3.1.1.10)$$

where  $\beta_1$  and  $\beta_2$  are positive scalars, chosen as a trade-off between reconstruction error, coefficients contribution and discrimination. An iterative optimisation routine motivated by pursuit algorithms [168, 135] has been suggested to solve (4.2.3.1). A modified version (Algorithm 2) is proposed in this work and a discriminating term is added to the original objective function.

The steps involved are detailed in Algorithm 1. In step 3 of the algorithm the residual variable  $\mathbf{R}_0$  is initialised to the input signal matrix  $\mathbf{Y}$ . Step 4 selects each atom, minimises (4.2.3.1). A projection matrix is computed in step 6.

The SDS technique (Algorithm 1) is a modified matching pursuit algorithm [167] that ensures the selected dictionary atom maximises the intra-class vs inter-class similarity

ratio.

### 3.1.2 | Orthogonal or Over-complete Dictionaries

A dictionary is considered to be over-complete when the number of basis vectors or atoms is greater than the dimensionality of the input data. In contrast, a dictionary is orthogonal when  $\mathbf{D}^T\mathbf{D} = I$ , eliminating any redundant dictionary atoms albeit may not offer the maximum sparsity.

Learning an over-complete dictionary using greedy methods ensures maximum sparsity, but the dictionary can be most coherent, i.e. highly redundant. Enforcing the incoherence condition on an over-complete dictionary [38] whilst solving for sparsity is a difficult task. In this work, we learn any form of non-linearity in the input data using a structured orthogonal dictionary and use the derived coefficients as target signatures for classification purposes.

Recently, kernel based DL [189, 96, 173] methods have been proposed as an effective way of capturing non-linearity in the input space and learn sparse encodings, simultaneously. However, these methods require the kernel or Gram matrix,  $\mathbf{K} \in \mathbb{R}^{N \times N}$ . For large-scale datasets, computing such a matrix is a computationally complex task, both in space and time. Work by Golts and Elad [72] incorporated an effective way of approximating a kernel matrix with an over-complete dictionary and Gangeh *et. al* [68] show that an kernelized orthogonal dictionary can be learnt. However, in this paper, we combine the kernel approximation method [82] with a discriminative orthogonal dictionary learning step. This separates our work from [72] and [68] improving classification accuracy and reduces the algorithm run-time significantly. Most real-world signals or images have an intrinsic non-linear similarity measure and can be harder to discriminate.



## 3.2 | Kernel Dictionary Learning

Kernel dictionary learning with applications to signal classification offers a solution to learning non-linearity in the input data. However, decomposing a kernel matrix for large datasets is a computationally intensive task. Existing papers on dictionary learning using optimal kernel approximation method improve computation run-time but learn an over-complete dictionary. In this chapter, we show that if we learn a discriminative orthogonal dictionary instead then learning and classification run-time can be significantly reduced. The proposed algorithm, kernelized simultaneous approximation, and discrimination (K-SAD), learns a single highly discriminative and incoherent non-linear dictionary on small to medium-scale real-world datasets. Extensive experiments result in  $> 97\%$  classification accuracy and show that the algorithm can scale both in space and time when compared to existing dictionary learning algorithms.

The Mercer kernel defines an implicit, non-linear transformation mapping the input data into a higher or even an infinite dimensional kernel feature space [12]. The *kernel trick* allows training of the input data in the high dimensional feature space without explicitly computing the exact mapping. The Mercer kernel  $\phi : \mathbf{Y} \times \mathbf{Y} \mapsto \mathbb{R}$  for training samples  $y_i$  and  $y_j$  can be expressed as

$$\Phi_{i,j} = \phi(y_i, y_j) = \langle \phi(y_i), \phi(y_j) \rangle, \forall i, j = 1, \dots, N \quad (3.2.0.1)$$

where  $\Phi$  is the implicit non-linear mapping associated with the kernel function  $\phi(\cdot, \cdot)$ . For the input matrix  $\mathbf{Y} \in \mathbb{R}^{N \times P}$ , the kernel matrix,  $\Phi \in \mathbb{R}^{N \times N}$ , contains values of all pairs of input signals, where  $\phi(y) \in \mathbb{R}^L$  is the image of  $y$  in a higher dimensional feature space,  $F$  and  $L \gg P$  is the dimension of  $F$ . Commonly used kernel methods are the linear kernel, polynomial kernels and Gaussian radial basis function (RBF).

### 3.2.1 | Kernelised Orthogonal Dictionary Learning

In this section, we present a non-linear orthogonal dictionary learning algorithm. The key contributions of this approach are: i) we report an improvement in run-time and classification accuracy on existing kernel DL methods [72, 173] by proposing to learn a discriminative orthogonal dictionary instead of an over-complete one; ii) unlike [72], we propose the use of an efficient SVD method for large matrices when approximating the kernel matrix using the Krylov method [82]; iii) we report state-of-the-art classification results and faster run-time on high-dimensional RGB-D and face recognition databases learning a single kernelised orthogonal dictionary; iv) finally, unlike [72, 68] we also map the kernel dictionary back into the input domain in order to better understand the dictionary structure and diversity.

#### 3.2.1.1 | Problem

We assume that for an input training matrix  $\mathbf{Y} \in \mathbb{R}^{N \times P}$  its kernel matrix  $\Phi \in \mathbb{R}^{N \times N}$  is of rank  $r \leq N$ . Hence,  $\Phi \approx \mathbf{B}^T \mathbf{B} = \phi(\mathbf{Y})^T \phi(\mathbf{Y})$ . Finally using  $\mathbf{B}$ , we compute “virtual samples”  $\Phi_{train} \in \mathbb{R}^{M \times N}$ , where  $M \ll P$ . Section 3.2.1.2 details how we approximate  $\Phi_{train}$ .

**Proposition 1.** *Given the virtual samples,  $\Phi_{train}$ , a dictionary,  $\mathbf{D} \in \mathbb{R}^{M \times M} \mid \mathbf{D}^T \mathbf{D} = \mathbf{I}$ , where  $M \ll P$ , the original discriminatory sparse coding problem (2.4.2.1) can be rewritten as:*

$$\min_{\mathbf{X}} \|\Phi_{train} - \mathbf{D}\mathbf{X}^T\|_2^2 + \beta_1 \|\mathbf{X}\|_0 + \beta_2 G(\mathbf{X}),$$

has a unique solution  $\mathbf{X}^* = T_{\beta_1}(\mathbf{D} \Phi_{train}, G(\mathbf{X}))$ . For a given vector,  $\phi$ , the hard thresholding operator,  $T_{\beta_1}$ , is defined as  $[T_{\beta_1} \phi]_i = \phi_i$ , if  $|\phi_i| > \beta_1$  and 0 otherwise.

*Proof.* See Appendix A. □

We add a discriminatory function  $G(\mathbf{X})$  (See Section 3.1.1) that maximises inter-class

variance and minimises intra-class variance of dictionary coefficients [52, 189].

**Proposition 2.** *Assuming  $\mathbf{D}^T \mathbf{D} = \mathbf{I}$ , the orthogonal kernel dictionary learning step can be written as:*

$$\min_{\mathbf{D}} \|\Phi_{train} - \mathbf{D}\mathbf{X}^T\|^2, \quad (3.2.1.1)$$

and has a unique solution  $\mathbf{D}^* = \mathbf{U}\mathbf{V}^T$ , where  $\mathbf{U}, \mathbf{V}$  denote the orthogonal matrices defined by the following SVD  $\Phi_{train}\mathbf{X}^T = \mathbf{U}\Sigma\mathbf{V}^T$ .

*Proof.* See Appendix A. □

### 3.2.1.2 | Proposed Approach

Some of the limitations with linear and non-linear sparsity based classification algorithms are:

- i) The Eigenvalue decomposition of the kernel or the Gram matrix may not scale with large data sets,  $O(N^2)$  and  $O(N^3)$  in space and time, respectively, where  $N$  is the number of observations.
- ii) Overcomplete dictionaries (where the number of dictionary atoms is greater than the dimension of the data) can be highly redundant and may not have a structure.

Recent work by Golts and Elad [72] also propose a solution to the implicit kernel problem, i.e. efficient computation of the kernel matrix  $\Phi$ . However, unlike our approach, they do not enforce incoherency or discriminatory constraints on the dictionaries and learn over-complete dictionaries for each target class. Our algorithm learns only one.

**Stage 1: Low-rank Kernel Approximation** The *low-rank kernel approximation* (LKA) stage is a pre-processing step that maps the high-dimensional input data on a low-dimensional non-linear feature space. We use the Nyström method, first introduced by Williams and Seeger [184] through uniform sampling of the input data. This method

---

**Algorithm 3: K-SAD**


---

**Input:**  $\mathbf{Y}_{train}, \mathbf{Y}_{test}$ , sampler,  $sr$ , kernel\_type,  $kt$ ,  $c$ 
**Output:**  $labels$ 

```

1 begin
  // Stage I - LKA. See Section 3.2.1.2
2   $\mathbf{Y}_s \mapsto vector\_quantisation(\mathbf{Y}_{train}, sr, kt, c)$ 
3   $\mathbf{C}_{train} \mapsto compute\_kernel(\mathbf{Y}_{train}, \mathbf{Y}_s)$ 
4   $\mathbf{C}_{test} \mapsto compute\_kernel(\mathbf{Y}_{test}, \mathbf{Y}_s)$ 
5   $\mathbf{H} \mapsto compute\_kernel(\mathbf{Y}_s, \mathbf{Y}_s)$ 
6   $\mathbf{H}^\dagger \mapsto \mathbf{\Lambda} \mathbf{\Sigma}^\dagger \mathbf{\Lambda}^T$ 
7   $\mathbf{\Phi}_{train} = \left(\mathbf{\Sigma}_k^\dagger\right)^{1/2} \mathbf{\Lambda}_k^T \mathbf{C}_{train}^T$ 
8   $\mathbf{\Phi}_{test} = \left(\mathbf{\Sigma}_k^\dagger\right)^{1/2} \mathbf{\Lambda}_k^T \mathbf{C}_{test}^T$ 
  // Stage 2 - ODL. Section 20
9  Set initial  $\mathbf{D}_0$ 
10 forall  $t \in [0, T]$  do
11    $\mathbf{X}_t = T_{\beta_1}(\mathbf{D}_t \mathbf{\Phi}_{train}, G(\mathbf{X}))$ 
12    $\mathbf{\Phi}_{train} \mathbf{X}_t^T = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$ 
13    $\mathbf{D}_{t+1} = \mathbf{U} \mathbf{V}^T$ 
14  $\mathbf{D} \in \mathbf{D}_{k+1}$ 
  // Stage 3 - K-NN Classifier
15  $\mathbf{X}_{test} = \mathbf{D}^T \mathbf{\Phi}_{test}$ 
16  $labels \leftarrow ModelKNNClassifier(\mathbf{X}_{test}, \mathbf{D})$ 
17 forall  $i \in [1, N]$  do
18    $\lfloor Distance(\mathbf{D}, \mathbf{X}_{test})_s$ 
19  $labels \leftarrow Sort(Distance(\mathbf{D}, \mathbf{X}_{test}))$ 
20 return  $labels$ 

```

---

computes a low-rank approximation to  $\mathbf{\Phi}$  of the form  $\tilde{\mathbf{\Phi}} = \mathbf{C} \mathbf{H}_k^\dagger \mathbf{C}^T$ . The Nyström method permutes  $\mathbf{\Phi}$  as:

$$\mathbf{C} = \begin{bmatrix} \mathbf{H} \\ \mathbf{S} \end{bmatrix} \mathbf{\Phi} = \begin{bmatrix} \mathbf{H} & \mathbf{S}^T \\ \mathbf{S} & \mathbf{A} \end{bmatrix} \quad (3.2.1.2)$$

where  $\mathbf{C}$  denotes the  $N \times c$  matrix formed by  $c$  columns,  $\mathbf{H} \in \mathbb{R}^{c \times c}$  is a matrix consisting of the intersection of  $c$  columns with corresponding  $c$  rows of  $\mathbf{\Phi}$ ,  $\mathbf{A}$  corresponds to a matrix composed of the remaining  $(N - c)$  rows and columns and  $\mathbf{S} \in \mathbb{R}^{(N-c) \times c}$  is a mixture of both. In this work, we use two sampling techniques and fix the size of  $c$ , i) random uniform sampling, and ii) vector quantisation (VQ) which uses the k-means clustering

method. The uniform sampling method selects  $c \ll N$  columns from  $\mathbf{Y}_{train}$  at random compared to the clustering or the VQ method that uses  $c$  cluster centres. Finding the best sampling technique for LKA is out of the scope of this work and interested readers can see [184] for a detailed analysis. Using (3.2.1.2) we construct  $\tilde{\Phi}$  as follows  $\tilde{\Phi} = \mathbf{C}\mathbf{H}^\dagger\mathbf{C}^T$ , where  $(\cdot)^\dagger$  denotes the pseudo-inverse operator.

Since  $\mathbf{H}$  is a symmetric positive semi-definite (SPSD) matrix, it can also be written in terms of its eigenvalues and eigenvectors. Hence, we re-write  $\mathbf{H}$  as

$$\mathbf{H} = \mathbf{\Lambda}\mathbf{\Sigma}\mathbf{\Lambda}^T \quad \text{and} \quad \mathbf{H}^\dagger = \mathbf{\Lambda}\mathbf{\Sigma}^\dagger\mathbf{\Lambda}^T \quad (3.2.1.3)$$

and  $(\mathbf{H}^\dagger)^{1/2} = (\mathbf{\Sigma}^\dagger)^{1/2} \mathbf{\Lambda}^T$ . Finally, we re-write  $\Phi_{train}$  as follows

$$\Phi_{train} = \left(\mathbf{\Sigma}_k^\dagger\right)^{1/2} \mathbf{\Lambda}_k^T \mathbf{C}^T \quad (3.2.1.4)$$

We solve line 6 of Algorithm 3 using the randomised version of the block Lanczos method [82] which is adapted for large datasets and produces nearly optimal accuracy. We repeat the above steps for the test dataset and get  $\Phi_{test}$ .

**Stage 2: Discriminative Coefficient based Orthogonal DL** Stage 2 of algorithm 3 presents the pseudocode of the ODL stage. The input to the algorithm is a training matrix  $\Phi_{train}$ . The optimisation problem in (2.4.2.1) does not optimise the learned coefficients for maximum discrimination. The discriminative term in (3.2.1.1),  $G(\mathbf{X})$  is expressed as  $G(\mathbf{X}) = \text{Trace}(S_w^{-1}S_b)$  [52]. Algorithm 3 lists the pseudocode of the steps involved.

### 3.3 | Evaluation on benchmark datasets

The SAD algorithm has been extensively evaluated on sonar data (Chapter 7) and LiDAR data (Chapter 4 and 6).

In this section, we have compared the K-SAD algorithm with other kernel dictionary

learning approaches. We evaluate our approach on four publicly available benchmark datasets: i) The RGB-D object dataset [108], ii) *The ORL AT&T face dataset*<sup>1</sup> [152], iii) *Extended Yale face dataset*<sup>2</sup> [110] and iv) *MNIST Digit Dataset*<sup>3</sup>. All our experiments were carried out on an Intel quad-core i7-4800MQ 64-bit computer with a CPU clock speed of 2.7 GHz and 16 GB RAM.

The Washington RGB-D dataset is a collection of 300 household objects grouped into 51 categories collected using the Microsoft Kinect sensor. The images are of size  $\approx 85 \times 85 \times 4$ . Several representation based methods, e.g. instance distance learning (IDL) [109], query adaptive similarity measure (QSM) [51], convolutional-recursive deep learning (CNN-RNN) [158], convolutional k-means descriptor (CKM) [24], depth kernel descriptors (KDES) [26] and hierarchical matching pursuit (HMP) [27] have reported results on the Washington RGB-D datasets. Lai *et. al* in [109] compute a single feature vector combining image, texture and depth features. Such features are then used for classification. Deep learning based methods, e.g. the CKD [24, 158], have reported state-of-the-art results where feature responses are learned in the vicinity of interest points and later combined into a descriptor. The CKD descriptor incorporates depth information which is then computed on image patches whose dimensions are pre-defined.

As suggested in [108] 10 trials with pre-defined training and test datasets<sup>4</sup> were adopted in our experiments and average accuracy is reported. We compare our results against state-of-the-art results in [51] and report classification accuracy and training and classification run-time. All the methods shown in Table 3.1 use the same training and test partitioning of the dataset. For our experiments we do not down-sample the images or extract any features [26, 109]; 51 dictionary atoms (one atom per category) are initialised at random from the input data and adopted using the DL method. A polynomial kernel of degree 8 with hyper-parameters  $\beta_1 = 0.1$  and  $\beta_2 = 0.01$  (see (3.2.1.1)) was used in our experiments.

---

<sup>1</sup><http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>

<sup>2</sup><http://vision.ucsd.edu/~leekc/ExtYaleDatabase/>

<sup>3</sup><http://yann.lecun.com/exdb/mnist/>

<sup>4</sup>[http://rgbd-dataset.cs.washington.edu/dataset/rgbd-dataset\\_eval/](http://rgbd-dataset.cs.washington.edu/dataset/rgbd-dataset_eval/)

Table 3.1: Accuracies(%) on RGB-D Washington dataset. A set of pre-defined training and test datasets ([108]) were used for these experiments.

Method	Accuracy(%)
SP-HMP [27]	$87.5 \pm 2.9$
IDL [109]	$85.4 \pm 3.2$
CKM [24]	$86.4 \pm 2.3$
CNN-RNN [158]	$87.6 \pm 2.0$
KDES [26]	$86.2 \pm 2.1$
Kernel SVM [109]	$83.8 \pm 3.5$
QSM [51]	$92.7 \pm 1.0$
<b>Our approach</b>	<b><math>97.572 \pm 0.265</math></b>

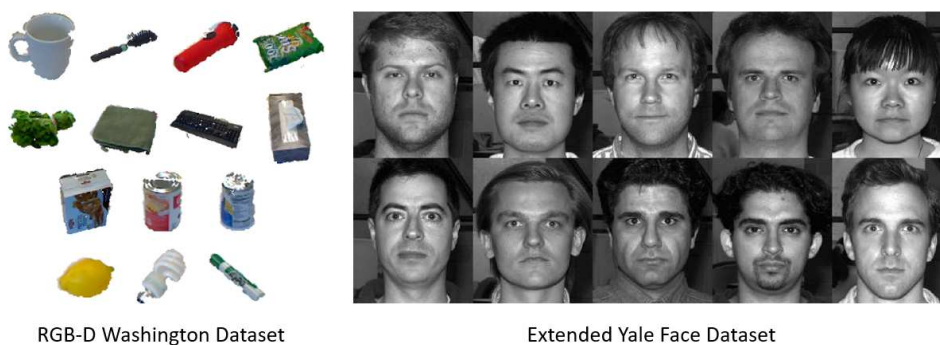


Figure 3.1: RGB-D Washington and YaleB Face dataset.

	Method	Accuracy	Total Execution Time(s)
<b>AT&amp;T Dataset</b>	OMP [168]	93.75±2.12	-
	$L_1 - L_s$ [105]	95.90 ± 1.15	-
	KK-SVD [173]	93.75 ± 0.05	8.3
	FDDL [189]	94.25 ± 0.03	1744.3
	LKA+FDDL [189]	92.75 ± 0.03	92.78
	<b>Our approach</b>	<b>96.58 ± 0.02</b>	<b>1.3</b>
<b>YaleB Dataset</b>	OMP [168]	91.97±0.96	-
	$L_1 - L_s$ [105]	94.22 ± 0.71	-
	KK-SVD [173]	91.53 ± 0.09	41.96
	LKA+FDDL [189]	87.2 ± 2.12	9000
	FISTA [18]	94.50 ± 0.82	-
	PALM [193]	93.19 ± 0.642	-
	LKDL [72]	96.33	-
	<b>Our approach</b>	<b>98.26 ± 0.03</b>	<b>31.4</b>

Table 3.2: Accuracies(%) on AT&amp;T and YaleB Datasets

### 3.3.1 | The AT&T (formerly ORL) Face Dataset

The AT&T face dataset is composed of 40 subjects and 10 images with pose and expression variation per subject. The dataset has images of size  $112 \times 92$ , captured on several different occasions in an up-right frontal position under a homogeneous background. We compare our approach to orthogonal matching pursuit [168],  $L_1-L_s$  [105] and FISTA [18]. Technical details of the above algorithms can be found in a recent survey [193]. In comparison to the methods discussed in the recent review [193] our method gives the least classification error. In our experiments, we use the full feature space of 10,304 pixels as input to the stage 1 of our algorithm, unlike [193]. We randomly initialise 240 dictionary atoms (6 per subject) from the non-linear mapped input data. A polynomial kernel of degree 8 was chosen for this experiment with  $\beta_1 = 0.1$  and  $\beta_2 = 0.01$ . Table 3.2 illustrates state-of-the-art results on the AT&T face dataset.



### 3.3.2 | Extended YaleB Face Dataset

The “Extended YaleB” face recognition database, in contrast to the AT&T database, is a larger database with 2,432 frontal images taken under varying lighting conditions and expressions. There are 38 subjects with  $\approx 64$  8-bit images per subject of size  $192 \times 168$ . Table 3.2 illustrates state-of-the-art results reported for this database against methods presented in [193]. In comparison to the methods and other kernel based DL methods our approach shows improvement in classification accuracy and is faster. For our experiments we do not re-size our images as done in [193] and use the full input space as an input to stage 1 of our algorithm. A polynomial kernel of degree 8 with 740 dictionary atoms (20 atoms per subject), initialised by uniformly sampling the kernel space was used.

### 3.3.3 | MNIST USPS Digit Dataset

The USPS MNIST dataset consists of 60,000 training images and 10,000 test images of size  $28 \times 28$ . The parameters used for this experiment are:  $\beta_1 = 0.1, \beta_2 = 0.01$ , 20 DL iterations, 300 dictionary atoms and a polynomial kernel of order 8. We compare classification accuracies and run-time execution against state-of-the-art results reported by Golts *et. al* [72]. Table 3.3 compare classification accuracy of our approach against approaches presented in [72, 173]. We use the KKSVD code made available by the authors. The original algorithm in [173] is not feasible for such a large dataset hence we employ the kernel approximation (Stage 1, 3.2.1.2) algorithm first and then use the KKSVD algorithm for kernel dictionary learning. We call this method “LKA + KKSVD” in Table

Table 3.3: Accuracies(%) on USPS Digit Dataset

Method	Accuracy(%)	Total Execution Time(s)
FDDL [189]	95.79	-
FDDL+ LKDL [72]	96.03	-
LKA + KKSVD [173]	74.62	9825.4
SVM (Gaussian Kernel)	<b>98.6</b>	-
<b>Our approach</b>	<b>96.42</b>	<b>92.86</b>

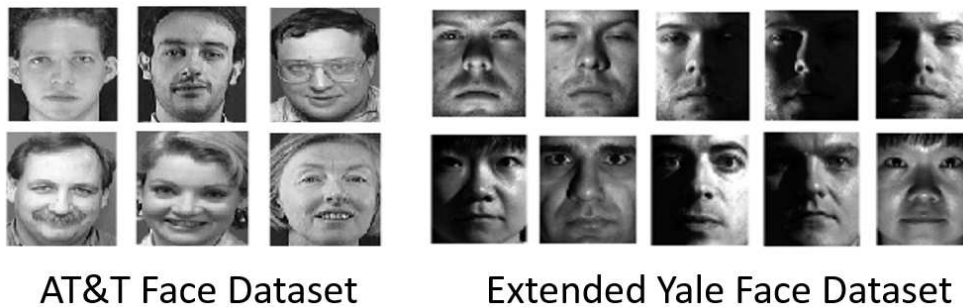


Figure 3.2: Exemplars of the learnt kernel dictionary.

3.3. Compared to [72] (using same set of parameters), the orthogonal discriminatory dictionary learnt using our approach is  $\approx 55$  times faster.

### 3.4 | Summary

This work improves on two central problems in DL algorithms i) handles non-linearity in the input space by improving classification accuracy on existing publicly available datasets; ii) reduces the learning and classification algorithm run-time through kernel matrix approximation. This work combines an orthogonal incoherent discriminatory dictionary learning method in the non-linear space with an efficient approximation of a kernel matrix. Unlike some existing techniques, which have produced these ideas individually, our approach learns a single non-linear orthogonal dictionary which is incoherent, minimising cardinality and maximising the discrimination capabilities in the non-linear space. We complement the small-runtime required by the orthogonal DL step with a fast kernel approximation stage in our algorithm. We report state-of-the-art results on large-scale high-dimensional datasets and report an average classification accuracy of  $\approx 97\%$  on 8-bit digits, face and RGB-D images. Unlike most sparsity based classifiers our approach uses the coefficients as target signatures. Finally, unlike [72, 68], the reverse mapping, *i.e.* pre-images, of the kernel dictionary, Figure 3.2, were computed using [107].

# Chapter 4

## Anomaly Detection & Classification from Aerial LiDAR Signals

We present an algorithm for the analysis and discrimination of small foot-print airborne, terrestrial full-waveform (FW)-LiDAR and time-correlated single photon counting (TC-SPC) LiDAR signals. We consider the problem of extracting multiple peaks and classifying them, simultaneously. We use data from aerial trials, flying over a rural setting in Austria, using three different FW sensors provided by RIEGL Gmbh, Austria. The backscattered laser pulse is time-gated to form a histogram, a full-waveform, whose inherent nature depends on several factors, e.g., the laser wavelength, surface geometry and transmission medium. We show that multiple peaks can be extracted and classified simultaneously into different classes, e.g. man-made terrain, buildings and trees. Since the real data lacks any ground-truth we validate our data on realistic synthetic data. The simulated data is generated using a custom modified open-source ray-tracer, terrain and target modeller. Our approach can extract multiple peaks and assign target label in a single step. We also compare our approach against well established robust peak analysis algorithms. Active imaging using light detection and ranging (LiDAR) has become a powerful tool not only to measure distance or depth but also spectral reflectance of different surfaces. This has advantages in remote sensing [81, 39], defence and security [53], and restoration and archaeology [65]. In a camouflaged scenario, e.g., objects hid-

den behind dense foliage, a FW-LiDAR penetrates such foliage and returns a sequence of echoes including buried faint echoes.

## 4.1 | Mapping using Waveform LiDAR

When mapping the forest or the undergrowth, an explicit assumption is made that a depth profile through a dense forest or an individual tree can be represented by a series of instrumental LiDAR returns from a set of *layers* (see Figure 4.2D) at distinguishable ranges. Thus, layer position defines the *context* of a tree and the number of layers, the number of individual spectra measured within a layer, and their contribution defines its *behaviour*. A depth profile of a tree or forest is made up of countless surface responses from leaf, bark and other man-made surfaces found at varying depths. The illustration in Figure 4.2D highlights one such example with spectral response registered at each layer. The top-most layer records green, since it mostly consists of high chlorophyll content representing natural material. The bottom-most layer records brown dead leaves and undergrowth and finally forest floor records spectra different to other layers, color coded here as grey. This sequence clearly deviates under diverse vegetation or with varying tree heights. The relationship between over-lapping layers may embed co-occurring patterns resulting in anomaly detection and tree species classification [53]. This can be achieved by breaking down a data sample in its entirety into several neighbouring sub-sequences and measure the co-occurrence of local-patterns within every sub-sequence. Authors in [53] learn lower-dimensional subspaces in order to seek unusual local patterns that do not appear in reference range-amplitude series. However, an implicit assumption is made in their work, the flying altitude is known and that each individual FW-MSL measurement is registered with respect to the flying altitude, i.e., that the length of the two samples and the range bin index are the same.

**Large vs Small-footprint Scanning** Large foot-print scanning can be considered as a precursor to find interesting signals [53], worthy of more detailed sampling and analysis.

This can be an attractive solution to keep surveying cost low. In contrast, we present an approach that models individual peaks in fine resolution LiDAR waveforms in order to detect and classify each peak, simultaneously.

The key contributions listed in this chapter are: i) We generate adaptive elementary functions, either from prior knowledge or learn from the data, for each peak and solve for a non-negative sparse contribution vector by introducing additional constraints; ii) if a trained set of elementary functions is given, we extract peak labels based on the sparse solution; iii) unlike previous works [8, 86, 20, 89], our approach can not only estimate multiple peaks but also classifying them simultaneously.

**Dissimilarities with Previous Methods:** The way we model a sequence of uniformly spaced LiDAR bins and generate initial parameter estimates is very similar to [89, 177]. However, we use a multiple parametric function (a generalised Gaussian) models for each peak with an additional parameter to control its shape. Additionally, we also learn a library of elementary functions from the data. The concept of using a library of shapes is not a new one [120, 86]. Unlike these methods, we combine the decomposition and classification problem into a single mathematical formulation motivated by sparse approximation and learn a dictionary of elementary elements. This is significantly different to the non-negative least square (NNLS) and Levenberg-Marquardt technique (LM) formulation proposed in [89, 187].

## 4.2 | Anomaly Detection in LiDAR signals

**Light detection and ranging (LiDAR)** is an active depth (spatial) sensing modality, operated from ground, aerial and space-based platforms. This has found a variety of applications in remote sensing and military scenarios for its ability to extract spatial information from a 3D scene. Passive multi-spectral imaging (MSI) and hyperspectral imaging (HSI) with dozens and hundreds of wavelengths respectively, measure spectral signatures for each individual image pixel with no spatial information. A fusion of spectral

imagery with 3D spatial information may seem desirable. This comes at a cost of both processing time and complexity. The evolution from passive MSI and HSI into spectrally aware active laser sensors seems natural since simultaneous measurements can eliminate data synchronization errors (both in space and time). Full-waveform multispectral LiDAR (FW-MSL) signals are active spatial measurements that are made simultaneously at different wavelengths using LiDAR based sensors. In the past, LiDAR sensing has seen applications in ecological studies for tree species classification and segmentation [141], geo-morphology [32] and foliage filtering [99]. The relevance of urban scene classification with applications to remote sensing has also been studied extensively. [99, 119, 48].

More recently, small foot-print sensing using MSL has been possible in retrieving structural and physiological properties of vegetation [180]. Although popular with the remote sensing community, the choice of MSL sensors and raw waveforms for anomaly detection in complex environments (urban city or dense forest) has not been investigated. Echo characterisation and classification is an active area of research but this requires sophisticated echo modelling algorithms which operate on the entire dataset. The aim here is to propose a framework that is smarter in its search for anomalies.

Automatic Target Recognition (ATR) scenarios may benefit, for example, in automatic detection of known targets hidden or camouflaged under dense foliage. The nature of anomaly detection differs from the ATR notion. The type of target, its structure and material may be unknown. Consider a surveying scenario where a forest is scanned looking for normality and abnormality in what is being perceived through different sensor modalities. A set of measurements can be made on what is being observed and one may be interested in narrowing down the target search window by finding spatial and spectral anomalies for further structural analysis. This can be carried out in two phases:

- *Phase 1* of the survey operates on a *large-footprint* setting which aims to identify regions with spectral and spatial (time) anomalies, i.e., similar spectral signatures found at depths which do not comply with learnt models. In this chapter we present a method to support Phase 1. Once these regions are identified, Phase 2 is applied

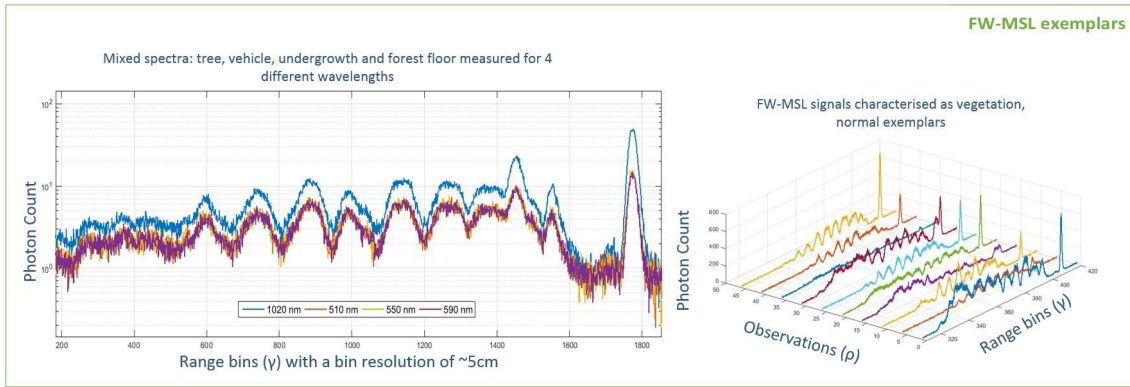


Figure 4.1: Example FW-MSL signals used to detect anomalies. (Left) A typical FW-MSL return of a large foot-print with a range resolution of  $\sim 5\text{cm}$  per bin. The backscattered returns shown here are recorded for 1020, 510, 550 and 590 nm wavelengths. This depth profile is a mixture of different spectra: vehicle, conifer, undergrowth and bare-ground. When observed from an altitude the tree top is the first peak, 600<sup>th</sup> bin on the left. (Right) Stacked FW-MSL exemplars.

to regions that contain such anomalies.

- *Phase 2* A *small-footprint* scanning scheme can be employed along with expensive signal modelling algorithms to deliver a dense 3D point cloud for structural analysis.

In order to support Phase 1 and Phase 2, a multi-channel (4 wavelength) LiDAR systems has been designed and developed [178]. In order to support future instrument design and parameter selection experiments on simulated and synthetic data have been carried out on a range of wavelengths: 4, 8, 16 and 32. Figure 4.1 illustrates example FW-LiDAR signals measured over a plot of vegetation.

## 4.2.1 | Contributions and Outline

In this chapter, a two-stage surveillance framework involving a sensing scheme that makes use of both small and large footprint LiDAR scanning is proposed. In order to model natural trees or forest plots, a layered representation is considered that conforms to a sparse approximation and discrimination model. The framework includes:

- This work aims to avoid computationally expensive signal modelling algorithms in order to characterize backscattering echoes for target localisation on large FW-MSL

datasets. To support such an aim, a cueing algorithm that reduces target search space by identifying anomalies in a large urban or forest scene is proposed. Such an approach is novel since it can be an efficient way of combining large and small foot-print scanning, especially when surveying large environments and the multi-spectral nature of the sensors used.

- A semi-supervised subspace (a set of linearly dependent basis, i.e., dictionary) learning method which learns normal LiDAR signals of different conifer species and forest terrain comprising of 8 different spectra (bark, needle, terrain, under-growth, steel, concrete, brick and clay tiles); and each new observation is transformed and reconstructed using the learnt dictionary. High reconstruction errors reveal hidden anomalies. The optimisation routine used here is influenced by the compressed sensing literature except that the objective function is modified in order to classify tree species and other signals. See Algorithm 1 and 2 in Chapter 3.
- The proposed framework can emphasize any unknown as opposed to a known signature [83] both in spatial and spectral context. To the best of our knowledge, this approach is the first attempt at applying such techniques to foliage penetration and anomaly detection on raw FW-MSL measurements.
- This work also shows that the proposed approach is widely applicable as the definition of an anomaly is not restrictive to a particular target type. The concept of context is associated with different FW-MSL measurements across different layers. In the past, this has been over-looked, echo modelling schemes have treated each individual echo independently and characterise with its location and amplitude. This work shows that relationship learning across layers and wavelengths can discover co-occurring patterns which can be highly relevant to anomaly detection.



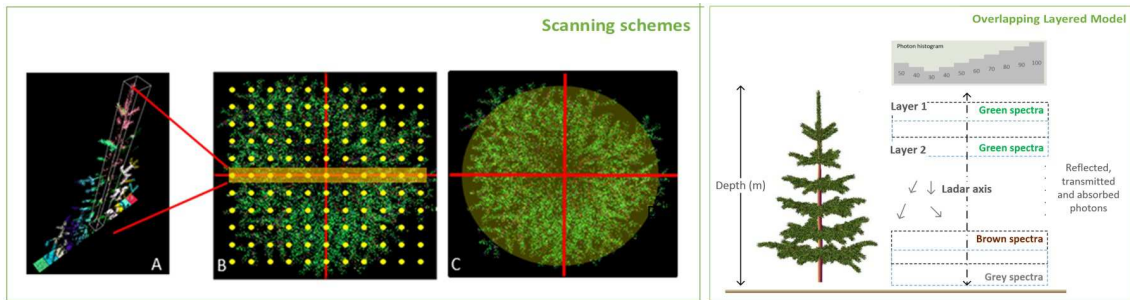


Figure 4.2: (A) A cross-section of a single conifer tree. (B) A *push-broom* style (small-foot print) dense scan pattern resulting in a cross-section slice of a tree. A *flash* style (large-foot print) scan pattern is illustrated in (C). (D) The forest/tree layered model.

## 4.2.2 | Problem Definition

Anomaly detection can be defined as a problem of detecting data samples that significantly deviate from an expected model. Methods that can capture essential patterns both in temporal and spectral dimensions within the original data space are desirable. Such patterns contribute significantly to the generating cause of an observation, a measured signal, and such representation are in a lower dimensional space.

Recent developments in *signal approximation* [38] and dimensionality reduction have an underlying concept, *Sparsity*, which means that an original signal, which is dense in certain basis, once its is transformed in to another convenient basis i.e., the contribution vector is almost sparse. It is assumed that an expected prior conforms to a sparse representation model, where the coefficients represent the contribution of a set of *atoms*, together called a subspace. This work also employs method that extract and learns such a subspace and their coefficients from raw FW-MSL data. Together, they contribute towards an *accurate* representation of the original FW-MSL measurement data. The following section introduces a layered representation of a forest, followed by an FW-MSL approximation model.

### 4.2.2.1 | Layered Representation of a FW-LiDAR Signal

An explicit assumption is made here that a depth profile through a dense forest or an individual tree can be represented by a series of instrumental LiDAR returns from a set

of *layers* (see Figure 4.2D) at different ranges. Thus, layer position defines the *context* of a tree and the number of layers, the number of individual spectra measured within a layer, and their contribution defines its *behaviour*.

A depth profile of a tree or forest is made up of countless surface responses from leaf, bark and other man-made surfaces found at varying depths. This is a *leaf-on* condition, dense healthy forests, mostly during summer. During winter, a *leaf-off* condition, most of the laser light penetrates trees easily. The illustration in Figure 4.2 highlights one such example. The top-most layer records green, since it mostly consists of high chlorophyll content representing natural material. The bottom-most layer records brown dead leaves and undergrowth and the forest floor records spectra different to other layers. This sequence clearly deviates under diverse vegetation or with varying tree heights. Hence, the intuition here is to capture the relationship between over-lapping layers in order to capture any co-occurring patterns.

This can be achieved by breaking down a data sample, in its entirety in to several neighbouring sub-sequences and measure the co-occurrence of local-patterns within every sub-sequence. For the sake of simplicity, running an overlapping sliding window over wavelength( $\lambda$ )-range( $\gamma$ ) series resulting in multiple sub-sequences. The aim is to learn lower-dimensional subspaces in order to seek unusual local patterns that do not appear in reference range-amplitude series. An implicit assumption made here is that the flying altitude is known and that each individual FW-MSL measurement is registered with respect to the flying altitude, i.e., that the length of the two samples and the range bin index are the same. Next, the concept of normality and abnormality in the context of range-amplitude-wavelength, FW-MSL data, is explained.

#### 4.2.2.2 | The Concept of Normality

In a semi-supervised anomaly detection technique, it is assumed that a database of *normal* exemplars that conform to non-anomalous data is known. Within the realm of this work, these exemplars may belong to several different classes, i.e., different tree species or

LiDAR returns of vegetation and bare forest floor. A continuous range-amplitude depth profile of a tree or forest is made up of several different spectra: bark, needle, dead leaves, undergrowth and forest floor. Such data samples are qualified as *normal*, free from any man-made objects. Deviation from such learnt model qualifies as *abnormality*.

Scenes observed using active/passive range sensors are made up of objects or targets, background clutter, sensor and environmental noise. Background clutter can be defined as something that is neither anomalous nor an object of interest. Interestingly, an anomaly characterised by a context or an outcome of a condition can be referred as a *contextual anomaly*, or a *conditional anomaly*. In the context of FW-LiDAR measurements, the input data is a range-amplitude-wavelength series and the context here is occurrence of an event at a particular range  $\gamma$  or wavelength  $\lambda$ .

In a layered representation (Figure 4.2) an anomaly can occur in any range bin, i.e., depth. This could be due to the variability in tree height across a section of the forest, tree health (climatic conditions) or a more interesting cause: presence of man-made object(s). It is highly impractical to collect data samples for all abnormal cases, but, what is available in abundance is a large amount of training data, normal events only, plots of natural forests with trees. The anomaly detection problem can be defined as follows:

*One wishes to learn the abundance of distinctive material spectra and their relationships across different layers within a range-amplitude series. The aim is to detect a deviation from the learnt normal models resulting in an abnormal event.*

#### 4.2.2.3 | Mapping Raw LiDAR Data into Sub-spaces

FW-MSL signals measured for  $\Lambda$  wavelengths can be expressed as:

$$\mathbf{Y} = [Y_1, \dots, Y_\Lambda]. \quad (4.2.2.1)$$

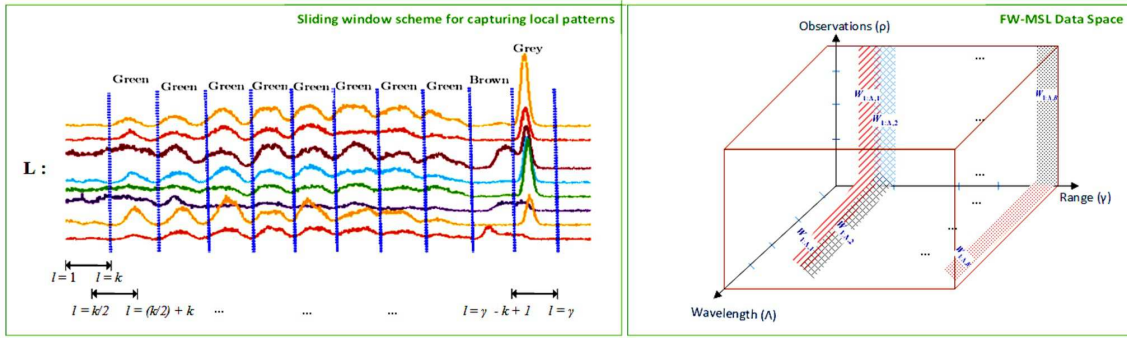


Figure 4.3: Sliding window concept in order to extract over-lapping range-amplitude series.

The original range-amplitude-wavelength series  $\mathbf{Y}$  for different wavelengths ( $\Lambda$ ) is mapped on to a set of overlapping sub-sequences  $\mathbf{W}$  and then transformed in to a feature subspace  $\mathbf{F}$ . The mapping can be written as follows:

$$\mathbf{Y} \rightarrow \mathbf{W} \rightarrow \mathbf{F} \quad (4.2.2.2)$$

The first half of the mapping is explained below then a mapping into  $\mathbf{F}$  is explained in more detail at the end of Section 4.2.3.1 using 4.2.3.2 and 4.2.3.3.

**Overlapping sub-sequence representation ( $\mathbf{Y} \rightarrow \mathbf{W}$ )** Figure 4.3 illustrates the windowing concept and subspace representation of normal vs abnormal events occurring in the range-amplitude-wavelengths space.  $\mathbf{Y} \in \mathbb{R}^{R \times P \times \Lambda}$  for  $P$  training samples,  $\Lambda$  number of wavelengths and  $r$  number of range bins. So the  $\mathbf{Y} \rightarrow \mathbf{W}$  can be written as:

$$\mathbf{W} = [W_{\lambda}, \dots, W_{\Lambda}], \quad (4.2.2.3)$$

where  $W_{\lambda}$  is a sub-sequence representing a layer of the forest/tree of size  $B = \frac{N}{R}$  that are partitioned on the range ( $r$ -axis), where  $N$  controls the window size and  $m$  is the amount of overlap. For a fixed wavelength  $\lambda$  this can be written as:

$$W_{\lambda} = \left\{ \underbrace{y_{\lambda,1}, \dots, y_{\lambda,N}}_{W_{\lambda,1}}, \underbrace{y_{\lambda,1+m}, \dots, y_{\lambda,N+m}}_{W_{\lambda,2}}, \dots, \dots, \underbrace{y_{\lambda,R-N+1}, \dots, y_{\lambda,R}}_{W_{\lambda,B}} \right\} \quad (4.2.2.4)$$

**Subspace representation** Each data sub-sequence,  $W_{\lambda,i}$ , where  $i = [1, 2, \dots, B]$  is the sequence index in 4.2.2.4 can be described with an over-complete basis (dictionary or subspace),  $D_{\lambda,i}$ , a coefficient matrix  $X_{\lambda,i}$  plus additive constant background noise:

$$W_{\lambda,i} \approx D_{\lambda,i} X_{\lambda,i}^T, \quad (4.2.2.5)$$

where  $W_{\lambda,i} \in \mathbb{R}^{P \times N}$  and  $D_{\lambda,i} \in \mathbb{R}^{P \times K}$  is an over-complete ( $K \gg P$ ) matrix and the resulting coefficient matrix  $X_{\lambda,i} \in \mathbb{R}^{N \times K}$ . The goal here is to find an optimal subset (*Subspace*)  $D_{\lambda,i}$  such that the set  $W_{\lambda,i}$  can be accurately reconstructed by  $D_{\lambda,i}$  and the size of this subspace is as small as possible. A simple way of doing this is to randomly and uniformly select basis vectors to build the subspace. This strategy can be risky as it may not capture the variability in the training data. A principled way of selection which selects an optimal subset of  $D_{\lambda,i}$  as a subspace is presented below.

Using all the vectors of  $D_{\lambda,i}$ , denoted by  $\{d_k\}_{k=1}^K$  (see Section 4.2.3.1), all vectors of  $X_{\lambda,i}$ , denoted by  $\{x_{\lambda,i}\}_{k=1}^K$  a signal  $Y_{\lambda,i}$  can be represented as a sparse linear combination of these columns (*atoms*). For a full-rank matrix  $D_{\lambda,i}$  and  $K \ll P$ , there are an infinite number of solutions to this representation (4.2.2.5). A stop constraint is needed on this solution, either set a sparsest approximation constraint, i.e.  $\min_{X_{\lambda,i}} \|X_{\lambda,i}\|_0$ , where  $\|\cdot\|_0$  is the  $l_0$  quasi-norm, which counts the non-zero entries of a vector or when the error term  $\epsilon$  goes below a certain threshold. The objective function for minimisation can be expressed as follows:

$$\arg \min_{D_{\lambda,i}, X_{\lambda,i}} [\|W_{\lambda,i} - D_{\lambda,i} X_{\lambda,i}^T\|_2^2 + \beta_1 \|X_{\lambda,i}\|_0], \quad (4.2.2.6)$$

where the parameter  $\beta_1 > 0$  is a regularization scalar that balances the trade-off between the reconstruction accuracy and sparsity. In the following section an iterative way to solve 4.2.2.6 is explained in more detail.

### 4.2.3 | Adopted Approach

A solution to (4.2.2.6) leads to a good approximation of measurement matrix  $W_{\lambda,i}$ . A discriminating trait is added to the original problem and finally the reconstruction error is used to differentiate normal from abnormal data. This framework allows for the classification of LiDAR signals (test samples) into different classes (e.g., tree species). A natural forest environment is made up of several tree species with variable tree height. Hence, one cannot fix one tree *layer* model for different tree species. One solution to capture such variability is to have species bound subspaces and coefficients.

As illustrated in the previous section the data is layered in to overlapping sub-sequences. Pseudocode for the training step is shown in Algorithm 2. The optimisation routine with an discriminative objective function is presented in Algorithm 1. The complete approach can be briefly listed in three steps:

1. *Subspace selecting & Learning* In the training mode a matching pursuit method is employed in order to learn optimal subspaces,  $D_{\lambda,i}$  and multi-wavelengths coefficient matrix  $\mathbf{F}$ , i.e.,  $\mathbf{W} \rightarrow \mathbf{F}_{train}$ . Such a representation can be extended by including a class discriminative (non-anomalous) criterion in the objective function such that during the approximation stage discriminating subspaces can be learned for different normal classes (conifer species, bare forest floor, and undergrowth). In testing mode, using the learnt subspace a new coefficient matrix  $\mathbf{F}_{test}$  is computed.
2. *Relationship learning & anomaly detection*  $\mathbf{F}_{train}$  is decomposed using a semantic learning technique, transformed and reconstructed. A test sample is projected back to the subspace and reconstructed, a high reconstruction error reveals the anomalies present in the test data samples.

### 4.2.3.1 | Step 1: Sparse Discriminative Signatures (SDS)

We introduced SDS in Section 3.1.1 and the idea of using discriminatory coefficients as feature vectors. Using (3.1.1.9) we re-write (4.2.2.6) as

$$\arg \min_{D_{\lambda,i}, X_{\lambda,i}} \left[ G(X_{\lambda,i}) + \beta_1 \sum_{k=1}^K \|\{X_{\lambda,i}\}_k\|_1 + \beta_2 \sum_{k=1}^K \|Y_{\lambda,i} - D_{\lambda,i} \{X_{\lambda,i}\}_k^T\|_2^2 \right], \quad (4.2.3.1)$$

where  $\beta_1$  and  $\beta_2$  are positive scalars chosen as a trade-off between reconstruction error, coefficients contribution and discrimination. An iterative optimisation routine motivated by pursuit algorithms [168, 135] is used. The algorithm in its entirety is illustrated in Algorithm 2.

### 4.2.3.2 | Step 2: Layered Representation of SDS

We repeat this step for each individual layer (sub-sequence) and stack the coefficient matrix  $X_{\lambda,i}$ , where  $\lambda = [1, 2, \dots, \Lambda]$ . For layer 1, the  $F_{\Lambda,1}$  matrix can be expressed as:

$$F_{\Lambda,1} = \begin{bmatrix} X_{\lambda,1} \\ \vdots \\ \vdots \\ X_{\Lambda,1} \end{bmatrix} \in \mathbb{R}^{(\Lambda \times P) \times K}, \quad (4.2.3.2)$$

and a set of such matrices can be represented as:

$$\mathbf{F}_{train} = \left\{ \underbrace{F_{\lambda,1}}_{W_{\lambda,1}}, \underbrace{F_{\lambda,2}}_{W_{\lambda,2}}, \dots, \dots, \underbrace{F_{\Lambda,B}}_{W_{\Lambda,B}} \right\}. \quad (4.2.3.3)$$

The subscript label under  $F_{\Lambda,B}$  shows which layer it belongs to. Conceptual diagram showing how  $\mathbf{F}_{train}$  matrix is constructed is shown in Figure 4.4. The final step is to reduce the dimensionality in the  $\Lambda$  dimension and learn the relationships across different layers at the same time. This is shown below.

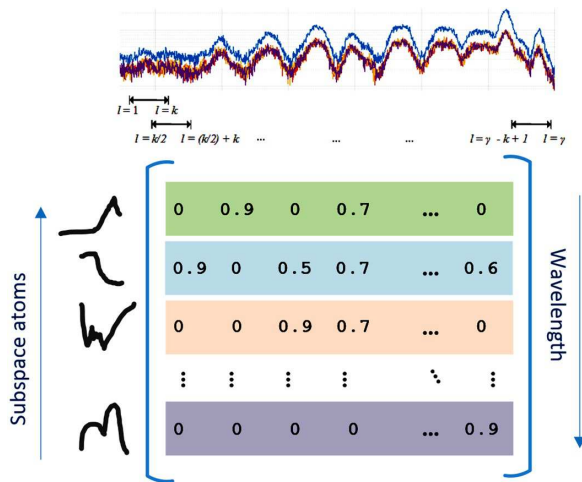


Figure 4.4: Conceptual diagram of multi-wavelength *atom-layer* relationship matrix (coefficient matrix) which is learnt for several over-lapping sub-sequences (columns) on range-amplitude lidar signals.

#### 4.2.3.3 | Step 3: Relationship Learning & Anomaly Detection

The Singular Value Decomposition (SVD) is used to decompose  $F_{train}$ :

$$\mathbf{F}_{train} = U\Sigma V^T, \quad (4.2.3.4)$$

where,  $U = [\mathbf{u}_1, \dots, \mathbf{u}_{\Gamma \times \rho}]$  ( $\rho$  = number of training examples) and  $V = [\mathbf{v}_1, \dots, \mathbf{v}_\eta]$  ( $\eta$  = number of basis vectors) are sets of singular vectors, and  $\Sigma$ , a diagonal matrix of the singular values. In this representation, the vectors of  $U$  correspond to the elementary atoms that make up a LiDAR signal and vectors of  $V$  corresponds to a tree/forest layer (window) in the semantic space. To preserve only  $q$  essential semantic relationships, one needs to approximate the *atom-layer* relationship matrix as follows:

$$\mathbf{F}_{train} = U_q \Sigma_q V_q^T. \quad (4.2.3.5)$$

In the training phase apply (4.2.3.5) to the atom-layer (coefficient) matrix  $\mathbf{F}_{train}$  resulting in  $U_q$  and  $\Sigma_q$ . During testing, the atom-layer coefficient matrix  $\mathbf{F}_{test}$  is transformed into



a lower-dimensional semantic space using:

$$\hat{\mathbf{F}}_{test} = \Sigma_q^{-1} U_q^T \mathbf{F}_{test}, \quad (4.2.3.6)$$

and reconstructed in to the original space using:

$$\tilde{\mathbf{F}} = U_q \Sigma_q \hat{\mathbf{F}}_{test}. \quad (4.2.3.7)$$

This step completes the mapping of  $\mathbf{W} \rightarrow \mathbf{F}$  which results in a mapping  $\mathbf{F}_{train} \in q \times K$  where,  $q \ll (\Lambda \times P)$ .

**Anomaly Detection** It should be noted that while the low-rank nature of  $U_q$  and  $\Sigma_q$  captures only the essential semantic relationships, (4.2.3.7) cannot reconstruct the original atom-layer coefficient matrix perfectly. The underlying intuition here is that if the co-occurrence relationships in the test LiDAR samples are different to those in the training data set, the reconstruction error will be large, i.e., the anomalous elements of the coefficient matrix will have a significantly large value. This is computed by calculating the distance between the learnt representation and reconstructed signal. The reconstruction error on each individual layer is computed:

$$\hat{A}_{score} = \|\mathbf{F}_{train} - \tilde{\mathbf{F}}\|_2^2, \quad (4.2.3.8)$$

Equation (4.2.3.8) computes a score with regard to an individual layer (sliding window), an anomaly score for each FW-MSL signal with  $N$  layers can be written as:

$$A_{score} = \frac{1}{N} \sum_1^N \tilde{A}_{score} \quad (4.2.3.9)$$

**Note** If the classification mode is selected, the above steps are followed except that training labels for different classes are made available. In such a scenario, the discriminative objective function (4.2.3.1) is used to learn atom-layer relationships and assign

Category	Property	Value
Waveform	Pixel Grid (pixels)	40 x 40
	Total waveforms	1600
	No. of range bins	400
	Bin resolution (ns/m)	1/0.01
Laser Pulse	Footprint (m)	0.5
	Wavelength (nm)	560
	Width (ns)	2, 4, 8, 16
	Beam divergence (rad)	
	Outgoing pulse shape	Gaussian
3D Scene	Flying altitude (m)	350
	Tree species used	Picea Engelmannii
		Picea Nigra
Pinus Parviflora		

Table 4.1: LiDAR Simulator Configuration

them class labels. For each test sample a K-NN match is computed in order to predict its class label and then compute (4.2.3.6 and 4.2.3.7). Initial experiments show that this reduces false alarms but at an expense of CPU time in solving (4.2.3.1).

### 4.3 | Anomaly Detection Results

We use simulated LiDAR waveform data of forest vegetation and real-world aerial laser scanning (ALS) data collected over sub-urban Austria using RIEGL’s waveform sensors. The real data lacks semantic labels, e.g. vegetation, terrain or buildings hence we analyse our peak extraction and discrimination approach on synthetic LiDAR waveforms. Our simulations are as close to the real data as possible. An operationally viable LiDAR model was set up with a systems configuration as similar to a real sensor, RIEGL’s LMS-Q1560 [145] and LMS-Q780 [144]. Table 4.1 illustrates the configuration settings used in our simulations.

### 4.3.1 | Synthetic Data - Multi-Beam Modeller

The concept of simulating small or large footprint LiDAR waveforms is not a new one [103, 187, 178]. Unlike previous models, we generate realistic forest terrains. Our simulator models a 3D forest-terrain, generates a realistic vegetation using a closed-source popular tree modeller and finally ray-traces and generate multiple LiDAR waveforms. The steps involved are illustrated as work-flow in Figure 4.5.

**Ray-tracing Configuration:** One of the key operational requirements is for the outgoing laser pulse to have enough energy to penetrate dense forest foliage. An outgoing waveform of pulse width 2ns was used to generate ground truth waveforms and point cloud. The standard setting of LMS-Q560 and LMS-Q780 is approximately 4ns. We use intermediate pulse widths of 4, 8 and 16ns to generate three different waveform sets for three different tree species. We use the 2ns dataset as the ground truth. When scanning an urban setting or dense forest we employ two scanning schemes (See Figure 4.2); firstly, a *push-broom* style scan where we carry out a dense grid style scan and secondly, a *flash* style scan with larger beam resulting in a large foot-print scan that illuminates an entire tree in one go. In Figure 4.5, we illustrate the waveform generation process using the depth and intensity images. A synthetic scene is ray-traced using a custom modified Povray<sup>1</sup> that also generates a depth image. Equation (4.3.1.1) is used to convolve the scene with a Gaussian instrumental (3-5ns) to generate synthetic FW-MSL waveforms. For the sake of simplicity the same instrumental response is used across all wavelengths. The received power  $P_a(r)$  of a LiDAR signal after interacting with  $S$  surfaces, at a range  $R$ , with  $d_a$  as the aperture diameter of the receiver sensor can be expressed as:

$$P_a(r) = \sum_{i=1}^S \frac{d_a^2}{4\pi\lambda^2 R_i^4} P_d(r) * \Gamma(r) * E(r) * \sigma_i(r), \quad (4.3.1.1)$$

where  $*$  denotes a convolution operation over  $P_d(r)$ , the transmitted power,  $\Gamma(r)$ , the system impulse,  $E(r)$ , environmental aerosol contribution and  $\sigma_i(r)$ , is the backscattering

---

<sup>1</sup><http://www.pov-ray.org>

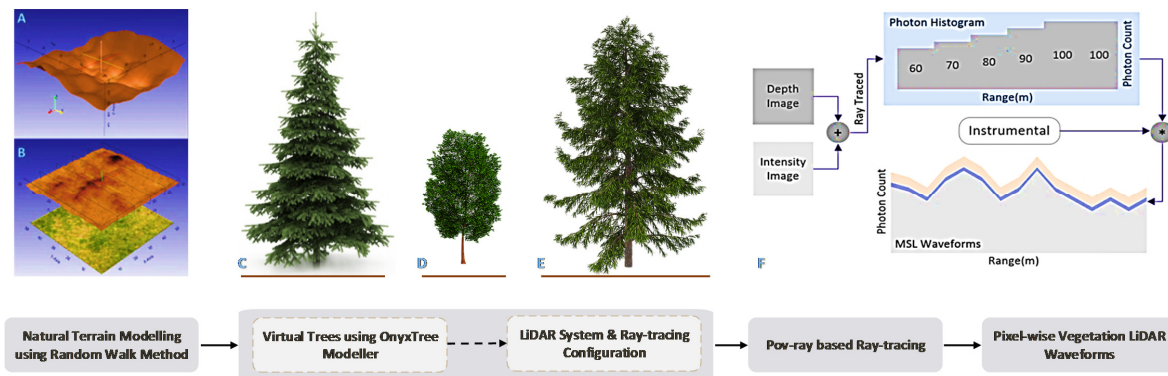


Figure 4.5: Realistic LiDAR waveform generation workflow. (A) Natural terrain generation method using fractal based Brownian Motion Fields. (B) Layer 1 is a fractal Brownian field (fBF) and layer 0 is a texture layer that is mapped to the elevation model. A closed-source tree modeller, Onyxtree is used to generate 3D tree models. (C) Tree 1 - *Picea Engelmannii*. (D) Tree 2 - *Pinus Nigra*. (E) Tree 3 - *Pinus Parviflora*. (F) The depth and intensity image from the rendered scene is then used to generate synthetic LiDAR waveforms.

cross-section.

The outgoing pulse width setting in our simulation is in agreement with a real sensor. This is so that the outcome is also inline with the waveform processing that will be applied to the real data. Also, the outgoing laser pulse is restricted to only nadir view to reduce disparity in target backscatter or atmospheric attenuation. The transmitted laser pulses from most real-world sensors are asymmetric, i.e., the trailing edge has a slightly longer tail versus the leading edge. However, in our simulations we approximate the outgoing pulse using a Gaussian distribution based on our observation of the actual outgoing pulses.

**Terrain and Vegetation Modeller:** We assume that the forest terrain is a simple continuous-time stochastic process and model it as a fractal or factorial Brownian Field (fBF) [53]. The terrains are made up of two layers as shown in Figure 4.5(A). Layer one, is an elevation field and texture is mapped to the elevation model in layer two. An elevation function is defined using a generalised Brownian motion without independent increments:

$$E [B_{t_1}, B_{t_2}] = \frac{\sigma^2}{2} [t^{2H} - (t_2 - t_1)^{2H} + s^{2H}] \quad (4.3.1.2)$$

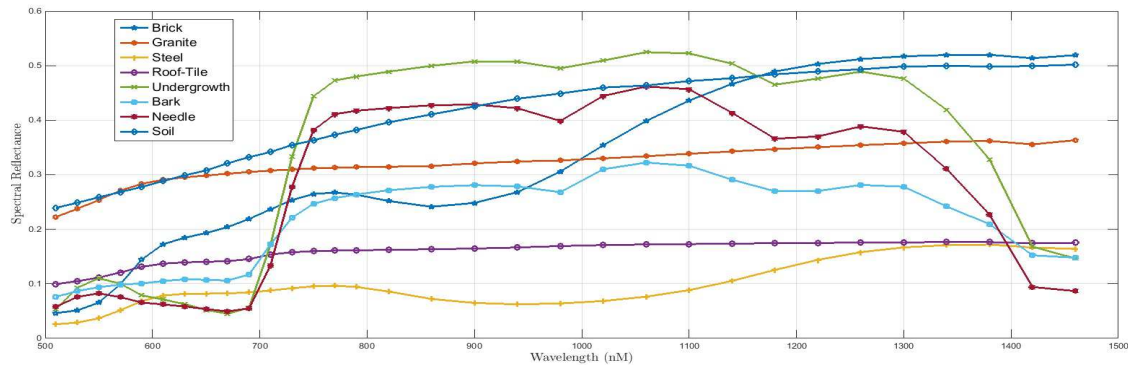


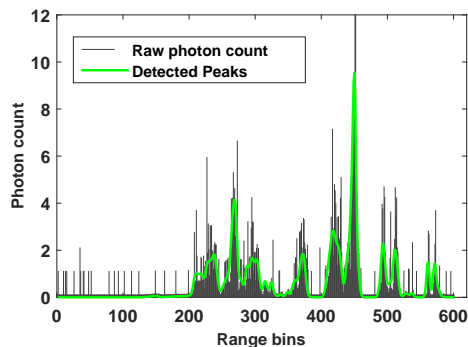
Figure 4.6: Measured material spectral reflectance for natural (tree bark, needle and soil samples) along with data from the Aster spectral library for selected *man-made objects*: building material and galvanized steel.

where  $B_t$  is a continuous-time Gaussian Process on  $[0, T]$  and  $\sigma$ , The roughness parameter  $H$  (Hurst index) is varied in order to control the raggedness of the motion and this can result in a variety of surfaces such as smooth desert type terrains and mountainous regions. Combinations of more than one surface can be used to build rocky terrains. We use a popular closed-source tree modeller, OnyxTree<sup>2</sup>, to model different species of trees. Three different species, *Pinus Parviflora*, *Pinus Nigra* and *Picea Engelmannii*, were used with variable height, spectral reflectance<sup>3</sup> and leaf area index (LAI). We also add a man-made object, a vehicle placed under the tree, in order to analyse the penetration and discrimination capabilities of our algorithm. Figure 4.7a illustrates a simulated single pixel waveform with added Poisson noise. In order to make the setting more realistic we add random undergrowth in between the trees. The reflectance spectra of all the materials used in the experiments is illustrated in Figure 4.6.

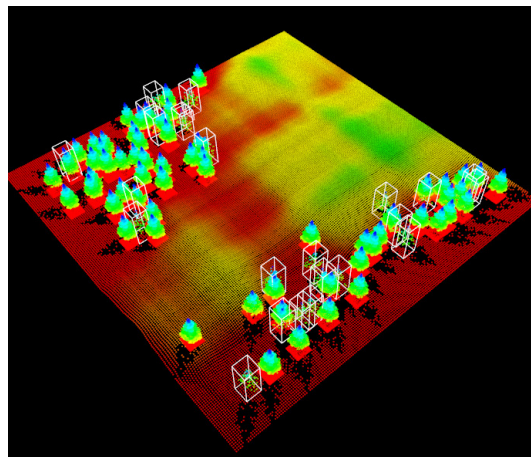
**Man-made Objects** Most of the man-made objects come from manufacturing processes or architectural constructions and are a complex arrangement of basic ensembles (cylinders, spheres or planes). Such objects are used in the experiments in order to study the spectral responses of different surfaces (elevated areas, small objects and vegetation surfaces) with multi-beam and multi-spectral LiDAR sensor. Man-made structures with

<sup>2</sup><http://www.onyxtree.com>

<sup>3</sup>The Aster2.0[15] spectral library provided by JPL, NASA is used for reflectance values. Passive measurements of tree samples under laboratory conditions [178] was carried out using a stabilized light source and spectrometer, with several needles and bark samples.



(a) Simulated waveform for a single pixel. The horizontal axis is the round-trip distance and the vertical its amplitude or photon counts. The peaks are extracted using the SPeED algorithm proposed in this thesis (Green line).



(b) Ray-traced 3D scene with three different tree species.

Figure 4.7: (Left) Example FW-LiDAR signal. (Right) A ray-traced scene with three different tree species.

elevated roofs and planar platforms along with vehicles are rendered and placed in a forest setting.

Figure 4.7b illustrates a test scene with three different tree species: i) *Picea Engelannii*; ii) *Pinus Nigra*; and iii) *Pinus Paviflora*. The waveforms from this scene were used to test the SPeED algorithm and also classify the tree species.

### 4.3.2 | Layered Anomaly Score

Scenes rendered using the proposed modeller, as illustrated in Figure 4.3, are used for experimentation. This small plot of forest is made up of two conifer tree species and several man-made objects are placed in between. These objects are vehicles closer to the ground and camouflaged with a random undergrowth introduced in between the trees.

An overhead surveillance flight simulation is assumed and flight path is traced and several regions of the forest are illuminated using large foot-print waveforms, backscattered returns are recorded using 4.3.1.1. Out of 105 test samples 23 FW-MSL signals contain several anomalies. The detections are shown in Figure 4.11b. Experiments on simulated and synthetic data have been carried out on a range of wavelengths (4, 8, 16, 32), rang-

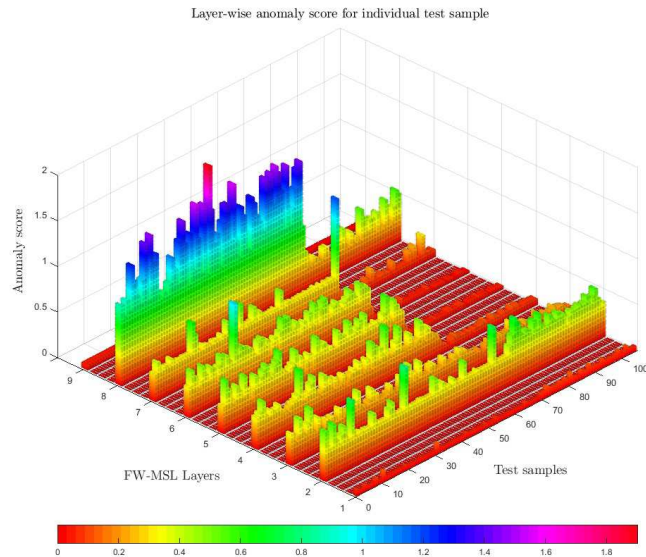
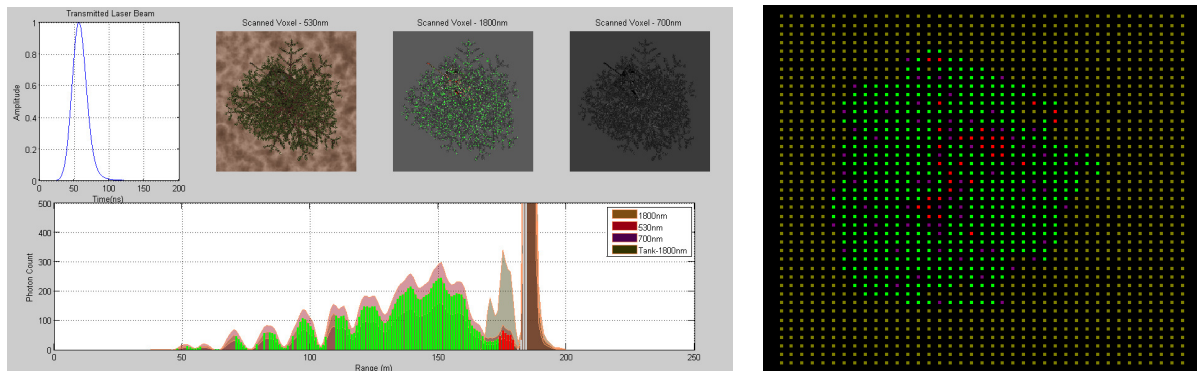


Figure 4.8: Layer wise reconstruction error computed using 4.2.3.7 is illustrated here. These results also label the species of the trees using a discriminative optimisation routine 3.1.1.9. Samples 34 and 69 record high reconstruction error as this is bare forest floor with some undergrowth. Layer 1 and 9 record the lowest reconstruction error as these layers record mostly background and system noise.

ing between 380 to 2500nm. Figure 4.10 illustrates the extracted spectra from the scene shown in Figure 4.9a using 32 bands. Further analysis is carried out only on the detected anomalous waveforms in order to produce a dense 3D point cloud. Figure 4.11a illustrates some of these detections. Also, shown is an anomalous waveform and one of the learnt models which is constructed using the learnt dictionaries for each layer. The anomalies, both in the point cloud and waveform have been highlighted with a box. The conifer in Figure 4.11a(A) conceals a large part of a vehicle that has two surfaces at varying depths.



(a) An integrated waveform for 40x40 pixels of a scene made up of forest floor, a tree and a vehicle hidden underneath. (b) Colour coded scene: Red - anomalies; Green - tree; and Floor - brown.

Figure 4.9: Anomaly detection on simulated scenes.

This is obvious in 4.11a(D), highlighted with a box between layer 7 and 8. Subtle anomalies are also detected, e.g., in Figure 4.11a(C) where only a tiny corner of the vehicle falls within the sensor field of view. Figure 4.8 shows reconstruction error computed using 4.2.3.7 for each individual layer. These results also label the species of the trees using a discriminative optimisation routine 3.1.1.9. Spectral and structural anomalies are added in the form of man-made objects and bare ground. Most of these objects are closer to the ground, Layer 7 and 8 clearly show the existence of these anomalies at 15, 20, 34, 69, 103, etc. Samples 34 and 69 record high reconstruction error as this is bare forest floor with some undergrowth. Layer 1 and 9 record the lowest reconstruction error as these layers record mostly background and system noise. Figure 4.9a illustrates an integrated waveform for a scene, made up of three materials: i) forest floor, ii) a conifer tree; and iii) a vehicle hiding in the undergrowth.

## 4.4 | Summary

This chapter suggests a two phase approach to anomaly detection and target detection in dense cluttered environments. This approach is the first piece of work that we are aware of which aims to look for spectral and depth anomalies in raw FW-MSL waveforms. Eight different spectra are introduced in a cluttered forest scenes in the form of man-made objects and vegetation. The suggested approach is widely applicable no matter the sensor modality. This has been considered as a precursor to find *interesting* signals, worthy of more detailed sampling and analysis in order to detect man-made objects, e.g. vehicles under foliage, or mines underwater.

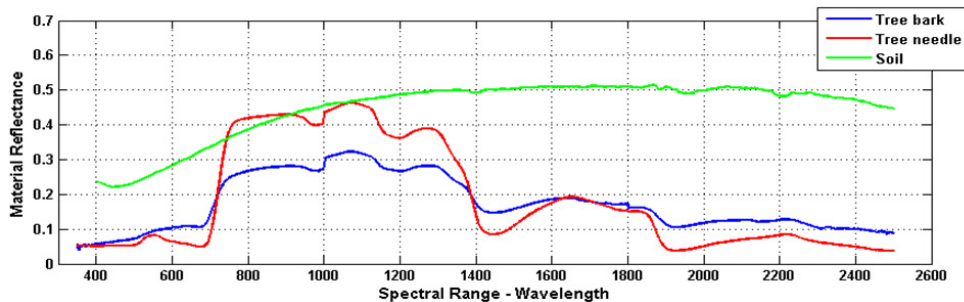
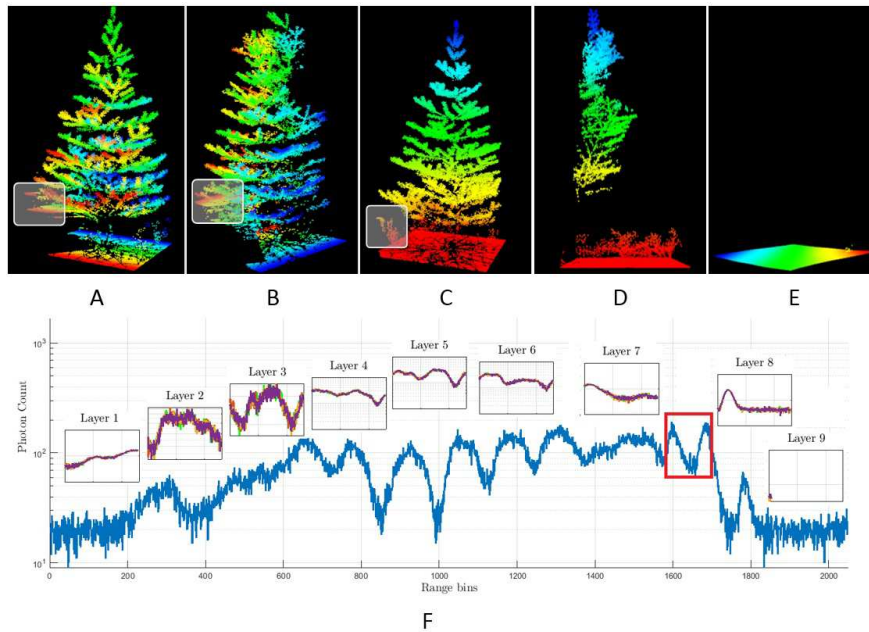


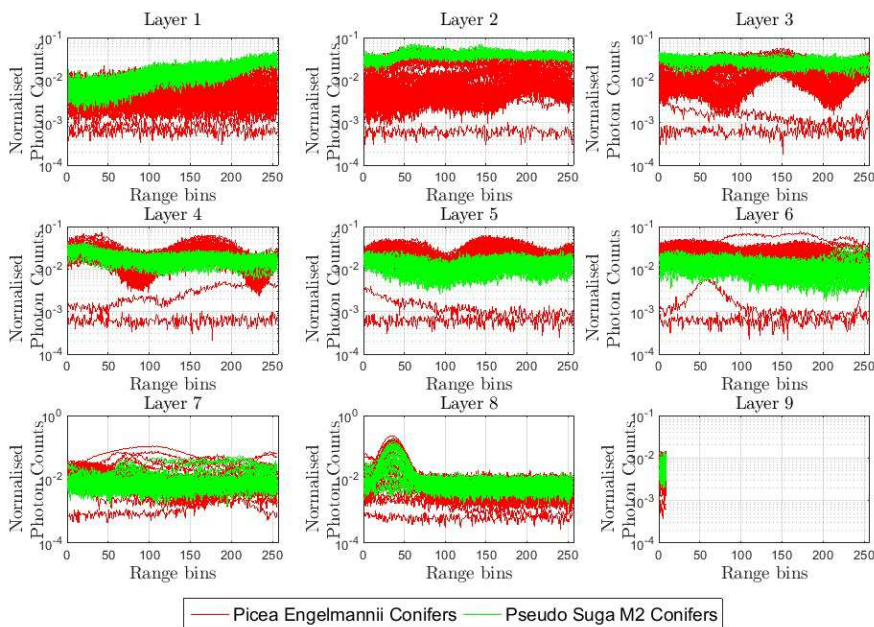
Figure 4.10: Material spectrum of an individual tree with a hidden anomaly.



The proposed approach is not restrictive and separate passive hyper-spectral imagery registered with LiDAR measurements can also be used as long as the training and test measurements are registered or geo-referenced with respect to the sensor. Any number of tree species can be learnt within this framework and as a by-product, the proposed approach allows segmentation of individual foot-prints for semantic scene labelling. This will be explored in detail in future works. FW-MSL data generation method proposed in this chapter generates synthetic data using natural terrain modeller along with a ray-tracer. Comparing this to past MSL work on studying tree physiology [178], the synthetic data is as good as real measurements.



(a) (A) A 3D point cloud of test sample 37, which is detected as an anomaly using the proposed algorithm confirms the presence of large two surfaced object. The detection peaks are shown in between layer 7 and 8. (B) Large section of a man-made object detected on test sample 41. (C) A small subtle anomaly is being detected on layer 8 of test sample 21. (D) and (E) are two anomalies where the ground spectra along with undergrowth dominates the coefficient space. (F) Layer-wise reconstruction of representative models are shown along with test sample 37. The reconstruction is build using dictionaries learnt on individual layers for 32 different wavelengths.



(b) Test FW-LiDAR signals belonging to 2 different conifer species for a single wavelength partitioned in to 9 layers using  $\mathbf{Y} \rightarrow \mathbf{W}$ . In one of the experiments 105 samples belonging to two different conifer species are used. Anomalies are introduced in the form of man-made objects at different depths and layers.

Figure 4.11: A 3D point cloud of the test sample and FW-LiDAR signals belonging to 2 different conifer species. Anomalies are introduced in the form of man-made objects at different depths and layers.

# Chapter 5

## Peak Modelling & Classification from Aerial LiDAR Signals

Section 2.2.1 provides the basics of waveform modelling and a review on techniques that model LiDAR waveforms. In this chapter, we present a novel approach, a multi-parameter learning technique called simultaneous peak extraction and discrimination (SPeED). For the sake of completion, we highlight some of the most popular techniques over the last decade and their strengths and weaknesses. These are listed in Table 5.1.

Hernández *et al.* [86] propose a method to fit the waveforms with a set of four piecewise exponential functions. This method creates a grammar of such functions that can be defined and adapted. They have shown how Bayesian analysis of TCSPC data can both detect very low signal levels, even buried in the background, and resolve surfaces at the order of 1 cm at a distance of 330 m. Ye *et al.* [190] proposed a parallel implementation of [86]. The processing time for each pixel takes 1 second on a high-end laptop machine with 16GB RAM and 2.6GHz clock speed. Altmann *et al.* [8] employ a Bayesian hierarchical model and propose a RJMCMC algorithm to detect and identify targets at long ranges using a TCSPC sensor. In their work, they assume the presence of a single target per pixel and prior distributions for their models, which might not be realistic for specific applications.

**Super-resolution:** More recently, Bhandari *et al.* show comparable results to the original RJMCMC approach [86] using a non-iterative method to process the ToF data. Their approach, under comparable resolution, claims to allow deterministic and fixed time processing of lower complexity. However, they assume a maximum of two targets per pixel. Jutzi *et al.* [100] employ an iterative Levenberg-Marquardt method and model surface properties using a set of Gaussian functions. They show that surfaces with a distance corresponding to less than 0.15m can be resolved. Unlike methods [20, 100], we do not make single, opaque or dual target assumptions and demonstrate the capability of encoding locally computed curvature information with their spectral response into target specific signatures. Our experiments show that surfaces with a distance corresponding to less than 0.04m can be resolved compared to previous methods.

**Motivation for a New Approach:** The above discussed methods have their own merits and demerits. So far we have tried to highlight some of them in Table 5.1. We have identified few key operational requirements for waveform fitting algorithms:

- Speed - time taken to extract finite number of peaks (e.g. for our experiments we set  $n = 10$ ) peaks per waveform.
- Resolvability - the minimum surface distance (m) that can be resolved between two peaks. We desire high resolvability at higher speed;
- Echo Model - does the method allow or use a library of echo models;
- Simultaneous classification - does an algorithm detect and classify each peak simultaneously?
- Multi-peak detection - does the method support full-waveform LiDAR?
- Time and memory complexity - algorithmic complexity per iteration, high complexity is not desirable;
- RMSE - we evaluate the quality by computing the root mean square error between original and fitted waveform, low RMSE is desirable.

CRITERIA	ALGORITHMS						
	RJMCMC [86]	NNLS-1 [177]	NNLS-2 [46]	EM [138]	FRI [20]	MP-MCMC [120]	SPeED
<b>Speed</b>	Poor	High	High	Medium		Higher	Highest
<b>Resolvability</b>	Med-High	Med-Low	Med-Low	Med-Low	High	Med-High	Highest
<b>Echo Model</b>	Piece-wise Exponential	Gaussian	Generalised Gaussian	Gaussian		Library	Adaptive echo library
<b>Simultaneous Classification</b>	No	No	No	No	No	Yes	Yes
<b>Multi-peak Detection</b>	Yes	Yes	Yes	Yes	Yes	Yes	Yes
<b>Time Complexity</b>	High	Medium	Medium	Med-High	NA	Low-Med	Low-Med
<b>Memory Complexity</b>	Medium	Large	Large	Large	NA	Medium	Medium
<b>RMSE</b>	Low	Low-Med	Low-Med	Low-Med	NA	Low	Lowest
<b>Drawback</b>	-Speed -Complexity	-Fix echo model -Resolvability -No noise est.	-Resolvability -No noise est.	-Fix echo model -Resolvability -No noise est.	-Dual peak assumption	-Complex to implement -No noise est.	-Noise est.

Table 5.1: Comparison Criteria for Waveform Fitting Algorithms

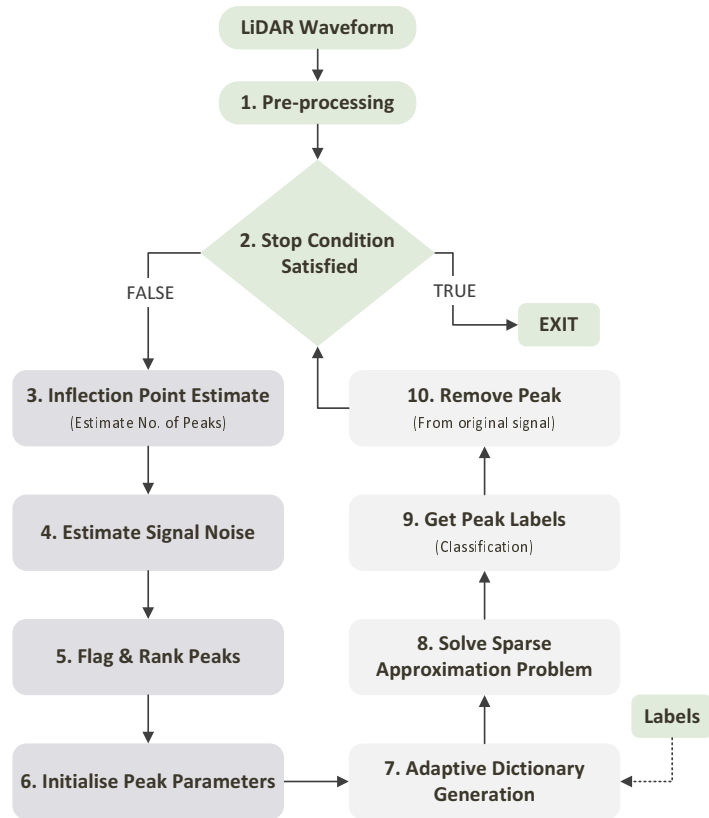


Figure 5.1: The Proposed Approach.

The methods compared in Table 5.1 have their own advantages and disadvantages. The method we propose can:

- resolve relatively close surfaces ( $< 0.05\text{m}$ ); this was tested against comparable techniques with fixed instrument response function (IRF);
- detect multiple (finite) number of peaks and label them, simultaneously, i.e. full-waveform compatible;
- be less complex (memory and time);
- estimate accurate peak parameters using a pre-defined dictionary (using their respective labels for classification); or
- generate an adaptive dictionary (generated on-line for each peak), based on the initial peak estimates.

Figure 5.1 illustrates the steps involved:

1. Pre-process waveform by smoothing (improve signal-to-noise ratio) the original waveform with a Gaussian of pre-defined half-width (Step 1 in Figure 5.1).
2. Check if the stop criteria is satisfied. The algorithm stops if the maximum number of peaks is reached or the reconstruction error is below a certain threshold (Step 2 in Figure 5.1).
3. Find inflection points  $n$  - this implies that the waveform will be decomposed into  $n$  elementary components. We initialise the peak parameters, estimate background noise and rank them (Steps 3 - 6 in Algorithm 5.1).
4. If a pre-defined set of elementary functions, a dictionary,  $\mathbf{D}$ , is known, we solve Eq. 5.1.3.1. However, if a dictionary is not known, we use the initial peak parameters of the top ranked peaks (Step 6, Figure 5.1), to generate an adaptive dictionary (Eq (5.1.3.2)),  $\mathbf{D}$ , (Step 7, Figure 5.1).
5. Solve Eq (5.1.3.1) and classify peak (Steps 8 - 9, Figure 5.1).
6. Remove the peak from the original waveform and go to Step 2.

## 5.1 | Simultaneous Peak Extraction & Discrimination

### 5.1.1 | Pre-processing & Inflection Point Estimates

The number of peaks needed to approximate a LiDAR waveform can be derived from the inflection points within that waveform. We make an assumption [89] that for  $n$  peaks within a waveform there will be  $2n$  inflection points at most. We pre-process the waveform, convolving the waveform with a Gaussian function of pre-defined width. The choice of the half-width can be tailored to the sensor system or dataset [89] or tailored to the half-width of the impulse response (which is known in most cases). If the half-width is unknown, then the return pulse from a calibration target (e.g. a Spectralon response at normal incident angle [52]) or a flat surface can be used. A smoothing function (Gaussian) is used to smooth the signal. The integrated area is set to one in order to maintain the

energy level (i.e. photon counts) of the original waveform. This is also a coarse way of diminishing false inflection points caused due to random system or atmospheric noise (e.g. dark photons). However, such a strategy can also result in low resolvability, i.e. two peaks appear close enough together that only two inflection points, instead of four, are detected, evident in [89]. In contrast: i) we do not approximate each peak with a Gaussian; ii) our experiments also show that using a much smaller half-width (1.5 bins) and ‘remove peak and repeat’ (Step 10 Figure 5.1) strategy we are able to maintain high-resolvability with very low RMSE (see Table 5.2).

### 5.1.2 | Noise Estimation, Parameter Initialisation & Ranking

We use the inflection points to extract initial peak parameters on position ( $\mu$ ), variance ( $\sigma$ ), i.e. full-width at half maximum (FWHM) and amplitude ( $A$ ) using the positions of the inflection points. For a given LiDAR waveform,  $y_i$ , using the positions of consecutive inflection points,  $t_{2k-1}$  and  $t_{2k}$ , the position  $\mu_k$ , half-width,  $\sigma_k$  and amplitude  $A_k$  of the  $k^{th}$  peak are given by:

$$\mu_k = \frac{(t_{2k-1} + t_{2k})}{2}, \quad (5.1.2.1)$$

$$\sigma_k = \frac{|t_{2k-1} - t_{2k}|}{2}, \quad (5.1.2.2)$$

$$A_k = y_i(\mu_k). \quad (5.1.2.3)$$

Finally, we also estimate mean background noise. An initial estimate from the region within the original waveform where no peaks are expected. For our experiments, we use the first 10 and last 10 bins of the waveform. We use the above values to generate a large dictionary (i.e. a set of elementary functions).

### 5.1.3 | Peak Estimation and Discrimination - A Sparse Solution

We propose a novel way (Steps 7 - 10, Figure 5.1) of estimating peak parameters inspired by the concept of sparsity. We formulate the peak extraction and discrimination of LiDAR



signals as a sparse approximation problem. Chapter 3 introduces the concept of sparse approximation and dictionary learning. We re-write the original problem (see Section 3.1.1.2) of signal approximation for a LiDAR signal and present a dictionary selection and learning approach.

We propose two scenarios:

- Dictionary Selection - A dictionary matrix is known or fixed, e.g. mathematical priors (generalised Gaussian (GG)). This is because, in our experiments, simulations and real-world, the instrument response function (IRF) can be modelled using a GG. In this scenario, we make use of Algorithm 4 as an optimisation technique in the SPeED pipeline (Figure 5.1) to solve (5.1.3.1). The fixed dictionary atoms and their corresponding sparse measurement vectors provide the contributions of each atom and the peak parameters, e.g. mean ( $\mu$ ), amplitude ( $A$ ), shape ( $\rho$ ).
- Dictionary Learning - A dictionary can be learnt or modified from the input data. This is the case when training data is made available. Algorithm 5 illustrates how we treat the input (per class) as a data matrix and update the dictionary atoms. It is important to note that the atoms are orthonormal. The advantage of doing this is to reduce redundancy in the dictionary matrix.

Given a LiDAR signal,  $y \in \mathbb{R}^P$  and a known set of elementary functions, a dictionary,  $\mathbf{D} \in \mathbb{R}^{P \times K}$ , we aim to find a non-negative sparse coefficient vector,  $x \in \mathbb{R}^K$  which implies the contribution of each elementary function in  $\mathbf{D}$ . Hence, for a selected support set  $s$ , the sparsest solution  $x_s$  can be found:

$$\operatorname{argmin}_{x_s \geq 0} \|y - \mathbf{D}x_s^T\|_2^2 + \beta_1 \|x_s\|_1, \text{ s.t. } \|x_s\|_0 \leq T_0 \quad (5.1.3.1)$$

---

**Algorithm 4:** ADAPTIVE DICTIONARY (**D**) SELECTION
 

---

**Initialise:**  $s = \phi, K = 0, x = 0$  and  $r_0 = y$ 
**Output:**  $x_s$ 

```

1 begin
2   while  $k < K \ \& \ \max(\mathbf{D}^T r_k) > 0$  do
3      $(i, j) \leftarrow \max(\mathbf{D}^T r_k)$ 
4      $s \leftarrow s \cup j$ 
5      $x_s \leftarrow \underset{x_s \geq 0}{\operatorname{argmin}} \|y - \mathbf{D}_s x_s^T\|_2 + \beta_1 \|x_s\|_0$ 
6      $r_{k+1} \leftarrow y - \mathbf{D}_s x_s^T$ 
7      $k \leftarrow k + 1$ 
8    $x|_s \leftarrow x_s$ 
    
```

---

### 5.1.3.1 | Dictionary Generated using Mathematical Priors

If the dictionary matrix is unknown, we generate one for each peak based on some prior knowledge (e.g. initial peak parameters or past literature [119, 177, 86]) and solve Eq. (5.1.3.1). For our experiments, we use a generalised Gaussian peak library:

$$\mathcal{GG}(y, \mu, A, \rho) = \frac{A^{1/2}}{2\Gamma(1 + 1/\rho)} \exp(-A^{\rho/2}|y - \mu|^{\rho}), \quad (5.1.3.2)$$

where,  $y$  is a zero vector of finite length,  $\mu$  is the peak location,  $A$  is the amplitude and  $\rho$  controls the shape of the peak. Samples are then drawn using the Gamma ( $\Gamma$ ) distribution.

For our experiments we work with a finite number of shapes ( $\rho = 1.5 - 8$ ). For each  $\rho$ , we generate GG samples, i.e.  $N$  dictionary atoms, using arbitrary location, inverse scale, and

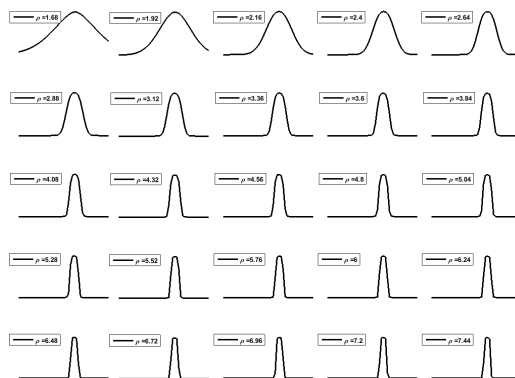


Figure 5.2: Peak dictionary illustrating exemplar atoms generated using Eq. (5.1.3.2) for a peak

**Algorithm 5:** ADAPTIVE DICTIONARY LEARNING

---

**Initialise:**  $s = \phi, K = 0, \mathbf{X} = \mathbf{0}$  and  $r_0 = \mathbf{Y}$   
**Output:**  $\mathbf{X}$

```

1 begin
2   while  $k < K \ \&\& \ \max(\mathbf{D}_s^T r_k) > 0$  do
3      $(i, j) \leftarrow \max(\mathbf{D}^T r_k)$ 
4      $s \leftarrow s \cup j$ 
5      $\mathbf{X}_s \leftarrow \underset{\mathbf{X}_s \geq 0}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}_s^T \mathbf{X}^T\|_2 + \beta_1 \|\mathbf{X}\|_1$ 
6     // For a given sparsity level  $t_0$ 
7      $\mathbf{Y} \mathbf{X}_{t_0} \approx \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$ 
8      $\mathbf{D}_s \leftarrow \mathbf{U} \mathbf{V}^T$ 
9      $r_{k+1} \leftarrow \mathbf{L} - \mathbf{D}_s \mathbf{X}_s$ 
10     $k \leftarrow k + 1$ 
11   $\mathbf{X}|_s \leftarrow \mathbf{X}_s$ 

```

---

FWHM parameters,  $\mu, A$  and  $\sigma$ , respectively. The GG dictionary atoms are generated by a transformation of Gamma random samples. Previous methods [119, 177, 89] employ several optimisation schemes that select a single peak from a library of elementary functions. These methods are not accurate when approximating asymmetric peaks. A single peak is a composition of a family of functions. Our approach handles such a situation by extracting a sparse contribution vector (sparsity constraint in Eq. (5.1.3.1)) which is further used, along with peak parameters  $\theta = \{A, \rho, \sigma, X\}$ , as a feature vector to classify each peak.

### 5.1.3.2 | Orthonormal Dictionary Learning

We cannot always model a library of peaks in advance or select and approximate each peak with a single function. This is due to several factors: i) unknown instrumental (e.g. different sensors have a different instrumental), ii) interaction of the unknown instrumental with different materials or geometry may result in several unknown but asymmetric shapes. The assumption we make here is that such peaks are repetitive (this is mostly true for forests and urban man-made objects). We use a small subset of the data as a training set in order to learn new mixtures of elementary functions and their contributions. Given the training data, we propose to use a single orthonormal dictionary  $\mathbf{\Psi}$ , a

GG dictionary (Eq. (5.1.3.2)) as a good initialisation which is updated using Algorithm 5. Figure 5.2 illustrates exemplar dictionary atoms with different shapes, controlled using  $\rho$ . We illustrate and discuss the impact of dictionary learning and selection for waveform modelling in Section 5.3.

We re-write the original sparse approximation as an alternating strategy jointly optimising the coefficients  $\mathbf{X}$  and the dictionary  $\mathbf{D}$  as follows:

1. **Coefficient update** given a dictionary  $\mathbf{D}$ :

$$\underset{\mathbf{X} \geq 0}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}^T\|_2^2 + \beta_1 \|\mathbf{X}\|_1 \quad (5.1.3.3)$$

2. **Dictionary update** given a coefficient  $\mathbf{X}$ :

$$\underset{\mathbf{D}}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}^T\|_2^2, \quad |\mathbf{D}_k| = 1, \forall k = 1, \dots, N \quad (5.1.3.4)$$

under some constraint on  $\mathbf{D}$ , e.g. only updating atoms with more information (SVD), see Algorithm 5.

The proof of steps 6 and 7 (Algorithm 5) can be found in [111].

## 5.2 | Multi-spectral Aerial LiDAR (Austria Trials)

Aerial trials were carried out in collaboration with Carbomap Ltd. UK and RIEGL GmbH, Austria in 2015. Two study areas were chosen for this paper: i) Site 1 - Schwengenthal (46°55'39.7" N, 15°56'56.4" E), Austria (blue box in Figure 5.3); ii) Site 2 - Bertholdstein (46°55'55.8" N, 15°57'24.2" E), Pertlstein, Austria (orange box, Figure 5.3). The data was collected, on two separate flight missions, using three different sensors, LMS-Q1560 [145], LMS-Q780 [144] and VQ-880-G [146], operating at 1550, 1064 and 532nm, respectively. The raw data stream was pre-processed by RIEGL for data reduction and noise removal. The sampled data was stored in the RIEGL Data Recorder DR560 together with scan angle and a high-accuracy GPS time stamp [177]. These can be exported

as the ASPRS laser *.las* or a more waveform friendly format PulseWaves [93]. All the detected pulses and their respective waveforms were stored in the *.pls* and *.wvs* files, respectively. The waveform processing software and algorithms for the Riegl sensors, reported in this work, has been developed around the PulseWaves format.

The Austrian multi-spectral ALS dataset comes with its own set of challenges: i) waveform registration and calibration for three different sensors; and, ii) the lack of any ground truth data. We solve the registration problem by conducting an octree based nearest search. In theory, under no navigational, instrumental errors, the nearest neighbouring points should corresponds to the same geographic location when scanned on two separate missions. However, the angle of incidence, environmental circumstances and aircraft drift means that the exact same location cannot be scanned. An octree based nearest search method is a suboptimal solution. Discrepancies after the data registration process is applied is illustrated in Figure 5.3 (Left), the Schwengenthal site. The blue and pink dots corresponds to the same geographic location scanned by the LMS-Q1560 and LMS-Q780 sensors on two separate flight paths. The octree based nearest-neighbour point cloud matches and waveform matches are shown in Figure 5.5 and Figure 5.4.

### 5.2.1 | Mobile FW-LiDAR Backpack System (Edinburgh Trials)

In collaboration with Carbomap, we developed a backpack system carrying the Riegl VUX-1LR. Figure 5.6 illustrates the backpack system. The timestamp raw waveform

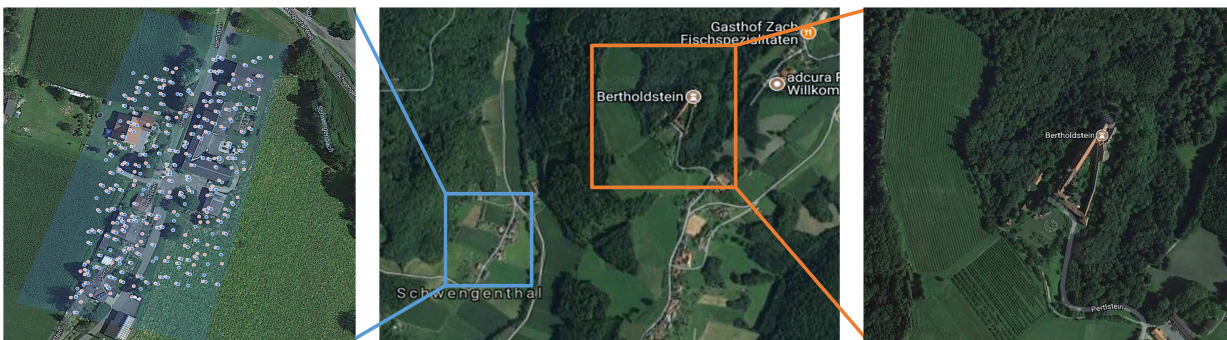


Figure 5.3: The study areas. (Left) Site 1 - Schwengenthal, Austria. Markers in pink and blue represent a small group of co-registered scan points collected using the LMS-Q560 and LMS-Q780 sensors. The waveforms are illustrated in Figure 5.4. (Center) A zoomed out view of the scan area. (Right) Site 2 - Bertholstein, Austria.

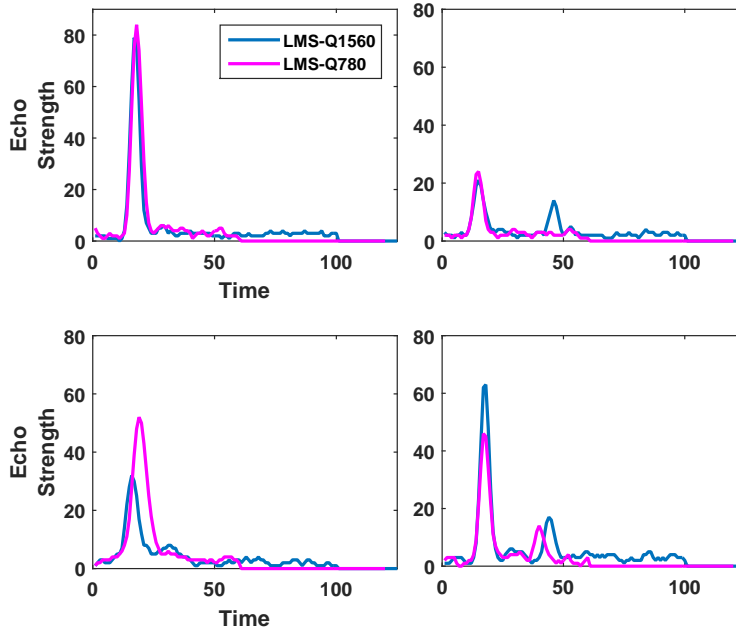


Figure 5.4: Example waveforms from Site 1 - Schwengenthal collected using the LMS-Q1560 (blue) and LMS-Q780 (pink) sensors.

contain range and intensity information for each incoming pulse. The data is stored in the Pulsewave format [93]. Our waveform processing software, written in C++/CUDA, can extract multiple peaks from the raw LiDAR waveforms and outputs a 3D point cloud.

Figure 5.10 illustrates the path, highlighted on a Google map image of University of Edinburgh’s King’s buildings campus. We walked using the backpack system (Figure 5.6) and logging the data on a handheld laptop.

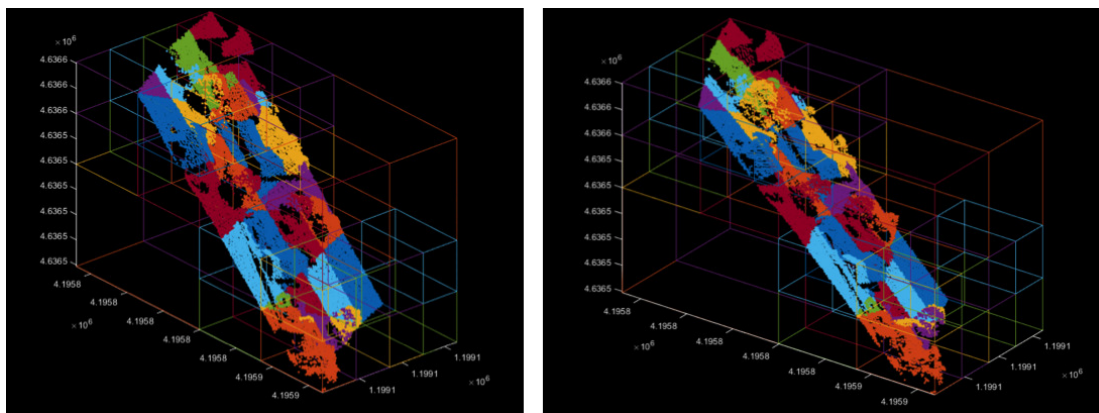
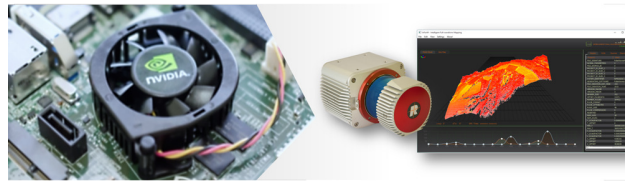


Figure 5.5: Nearest-neighbour correspondences between the LMS-Q1560 and LMS-Q780 point cloud data.



(a) A custom designed pack-frame with the Riegl VUX-1LR, a GPS receiver and a data-logging laptop (not shown in the image).



(b) The waveform processing software (middle) was developed in C++/CUDA and the raw data from Riegl's VUX-1LR was processed on a NVIDIA TX1 (left).

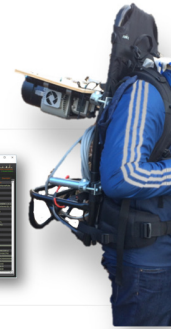


Figure 5.6: Full-waveform LiDAR backpack system for terrestrial scanning

## 5.3 | Point Cloud Classification Results

We employ and analyse our approach on the synthetic scenes and ALS data collected over sub-urban Austria. Since the real data lacks any ground truth, we use the 3D point cloud generated using our simulator as ground truth data. Three different scenes are simulated with different tree species: i) Tree 1 - *Picea Engelannii*; ii) Tree 2 - *Pinus Nigra*; iii) Tree 3 - *Pinus Parviflora*. Each tree is sampled into 40 x 40 pixel grid with a grid size of 0.5m. Unlike [187] we chose to sample the waveforms in 400 time bins for each pixel and corrupt them with Poisson noise.

### 5.3.1 | Metrics used for Comparison

For the analysis, we use metrics that not only measures signal (peak) approximation accuracy but also the impact of the learnt dictionary on peak approximation. Unlike [187], we use a distance metric to quantify 3D point cloud similarity between two objects.

1. **True Discovery Rate (TDR)** - We define TDR or sensitivity as the ratio between total number of true detections, the time bin index of a detected peak that agrees

with the ground truth, and the total number of true peaks.

$$\text{TDR} (\%) = \frac{\# \text{ of true detections}}{\text{total } \# \text{ of true peaks}} \quad (5.3.1.1)$$

2. **False Discovery rate (FDR)** - This metric emphasises the number of false alarms raised by the algorithm. It is the opposite of the true detections and can be defined as

$$\text{FDR} (\%) = \frac{\# \text{ of false detections}}{\text{total } \# \text{ of detected peaks}} \quad (5.3.1.2)$$

We report the TDR and FDR values on the simulated tree dataset using our proposed and the RJMCMC algorithm in Table 5.2.

3. **Modified Hausdorff Distance (MHD)** - This is a popular distance measure [61] for object matching, computing the similarity between two objects based on their shape attributes. If the Euclidean distance between a point  $p$  and  $\bar{p}$  is defined as  $d(p, \bar{p}) = \|p - \bar{p}\|$ , then the distance between a point  $p$  and set of points  $\bar{\mathbf{P}} = \{\bar{p}_i\}_{i=1}^{N_{\bar{p}}}$  can be written as  $d(p, \bar{\mathbf{P}}) = \min_{\bar{p} \in \bar{\mathbf{P}}} \|p - \bar{p}\|$ . We define the generalised Hausdorff distance as

$$d(\mathbf{P}, \bar{\mathbf{P}}) = \max_{p \in \mathbf{P}} d(p, \bar{\mathbf{P}}). \quad (5.3.1.3)$$

The directed distances  $d(\mathbf{P}, \bar{\mathbf{P}})$  and  $d(\bar{\mathbf{P}}, \mathbf{P})$  are combined into an undirected distance measure as follows

$$M(d(\mathbf{P}, \bar{\mathbf{P}}), d(\bar{\mathbf{P}}, \mathbf{P})) = \frac{N_p d(\mathbf{P}, \bar{\mathbf{P}}) + N_{\bar{p}} d(\bar{\mathbf{P}}, \mathbf{P})}{N_p + N_{\bar{p}}} \quad (5.3.1.4)$$

Table 5.2 lists MHD values (lower values signify higher similarity) for the three tree species used in our experiments when compared against the GT 3D point cloud. We use the 3D modelled point cloud (generated using our simulator Figure 4.5) as the ground truth.



4. **Root mean square error (RMSE)** - this is the simplest of all the metrics, measuring the root mean square error between the truth and modelled waveforms using our approach

$$RMSE = \sqrt{\frac{1}{B} \sum_{i=1}^B (l_i - \bar{l}_i)^2}, \quad (5.3.1.5)$$

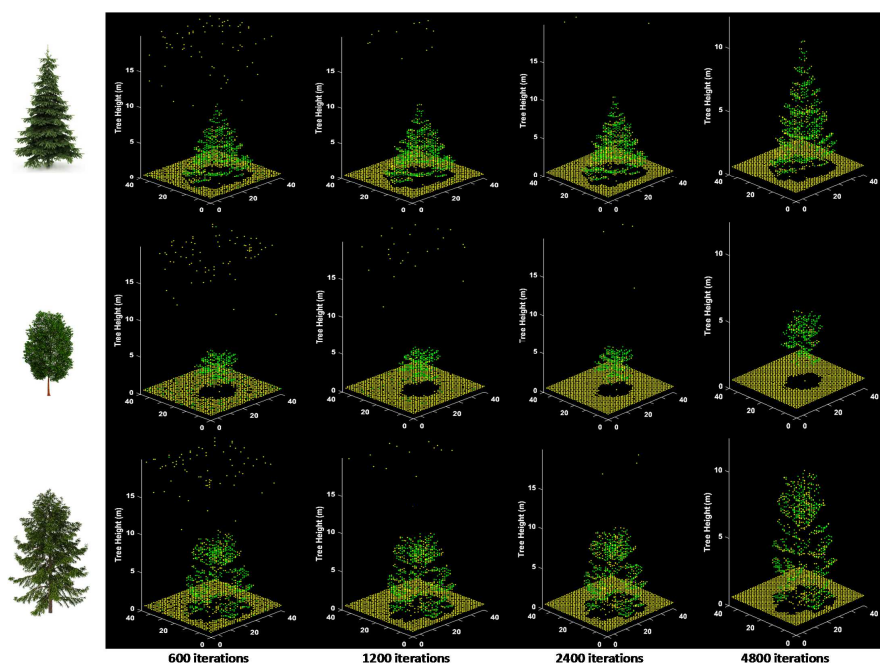
where  $l_i$  is the true waveform,  $\bar{l}_i$ , the estimated waveform of length  $B$ . The RMSE entries in Table 5.2 are averaged values for each tree using different reference beams (2, 4, 8 & 16).

5. **Time and Avg. No. of Peaks** - These metrics record the average time taken to process a single waveform and the average number of peaks extracted per waveform. We average this out for 40 x 40 pixel grids.
6. **Impact of Dictionary Learning** - This is not a metric per say, however, a way to quantify what is the contribution of the learnt dictionary (i.e. does the dictionary have a structure?) towards peak estimation and discrimination. Figure 5.7 illustrates histograms and cumulative distribution function (CDF) of the learnt dictionary atoms. The distinctive shapes of different target CDF's is due to the sparsity constraint, allowing more than one peak to contribute towards peak estimation. Further, such distributions can be highly beneficial in designing probabilistic clustering [124] algorithms.

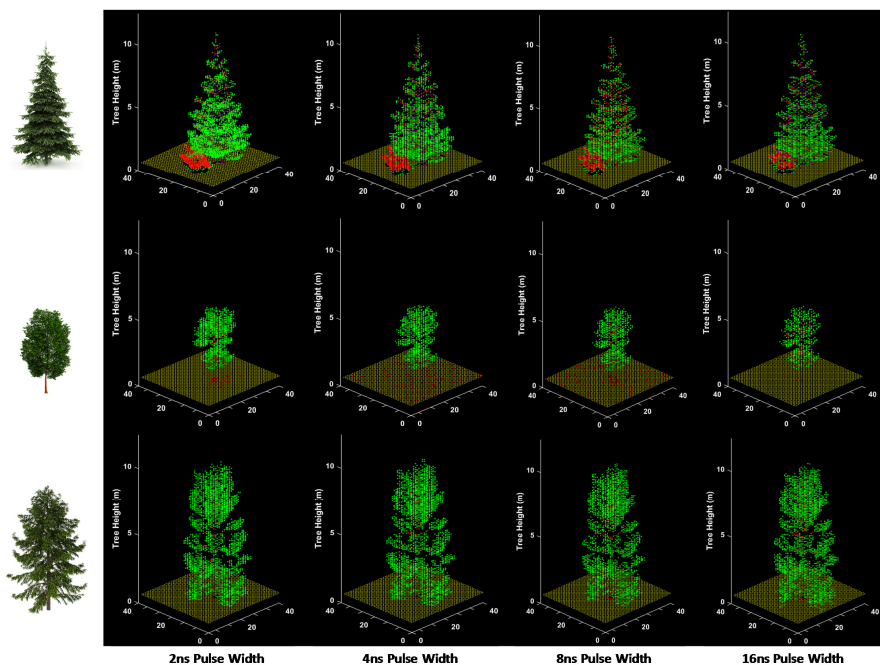
We compare our approach with the RJMCMC approach [86]. Recently, variants of this algorithm [178, 8] have been proposed. Although robust, they have a fundamental flaw, they are not suitable for practical purposes due to high time complexity. When using the SPeED algorithm on the simulated and real-data, we follow the steps illustrated in Figure 5.1 and repeat the experiment using different outgoing pulses (2, 4, 8 & 16ns).

### 5.3.2 | Results on Simulated Data

**Recovering tree cross-section & On-line Classification** - We assess the cross-section of the tree plot by comparing the processed (deconvolved using SPeED and RJMCMC) waveform result with the GT data generated using our simulator. Figure 4.7a shows an exemplar waveform one of the trees and approximated using the SPeED algorithm. The performance of our algorithm is by far the best when the outgoing pulse is as small as 2ns. Unlike previous methods [100, 187, 178, 177], we show that surfaces with a distance corresponding to less than 0.05m can be resolved for multiple peaks per pixel. For Tree 1, *Picea Engleannii*, with an outgoing pulse of 2ns, the improvement on the average number of peaks extracted per waveform when compared to the RJMCMC approach is 2.3 times. We show that, on average, 1.4 times more true detections ( $\approx 91\%$ ) can be achieved and reduce the time required by 234 times (0.047 s) to process an individual waveform. However, the improvement is even more significant when we compare wider outgoing pulses. Wider pulses make it difficult to separate surfaces whose separation is smaller than the waveform width. For example, with Tree 3, *Pinus Parviflora*, we observe  $\approx 2$  times more true detections when compared to the RJMCMC approach. However, the RJMCMC has low false detection rate  $< 10\%$ . RJMCMC is meant to estimate the peak parameters but does not have an on-line classification method. Figure 5.7b illustrates the 3D point cloud data extracted using the SPeED algorithm. Each 3D point is classified into different categories - vegetation, forest floor and vehicle. This is not the case with the RJMCMC approach, however, the peak parameters can be used for classification. Figure 5.7a illustrates the point cloud extracted using the RJMCMC approach (after 4800 iterations). The MHD score for both approaches have been listed in Table 5.2. The SPeED algorithm has lower scores for all the trees used in our experiments even with a wide pulse width of 16ns. Finally, we estimate the peak shape CDF's for different target spectra (vegetation, forest floor and vehicle) learnt using Algorithm 5. It is evident from Figure 5.7 that a diverse set of dictionary atoms are selected for different targets.



(a) Unsupervised classification results (2ns pulses) for three different species using the RJMCMC approach at 600, 1200, 2400 and 4800 iterations. (Top) *Picea Engelannii*, (Middle) *Pinus Nigra* and (Bottom) *Pinus Paviflora*.



(b) Unsupervised classification results (using 4 different pulse widths - 2, 4, 8 and 16) for three different species. (Top) *Picea Engelannii*, (Middle) *Pinus Nigra* and (Bottom) *Pinus Paviflora*.



Picea Engelannii



Pinus Nigra



Pinus Parviflora

Method	Pulse Duration	Picea Engelannii				Pinus Nigra				Pinus Parviflora			
		2ns	4ns	8ns	16ns	2ns	4ns	8ns	16ns	2ns	4ns	8ns	16ns
SPeED	TDR (%)	90.98	87.38	72.38	68.89	92.80	87.56	83.38	76.25	94.15	90.46	85.93	79.93
	FDR (%)	21.70	17.57	15.12	17.12	13.34	10.48	10.78	10.04	4.120	4.00	3.86	4.69
	MHD	0.232	0.223	0.263	0.299	0.148	0.146	0.163	0.204	0.280	0.293	0.324	0.407
	Time (s)	0.047	0.042	0.034	0.034	0.027	0.024	0.023	0.020	0.046	0.043	0.041	0.043
	Avg. No. Peaks	3.134	2.848	2.292	2.237	1.793	1.638	1.565	1.340	2.998	2.877	2.728	2.547
	RMSE	0.022	0.038	0.122	0.117	0.042	0.082	0.117	0.236	0.064	0.067	0.098	0.089
RJMCMC (2ns)	Iterations	<b>600</b>	<b>1200</b>	<b>2400</b>	<b>4800</b>	<b>600</b>	<b>1200</b>	<b>2400</b>	<b>4800</b>	<b>600</b>	<b>1200</b>	<b>2400</b>	<b>4800</b>
	TDR (%)	57.11	61.86	64.55	64.32	62.40	64.55	66.44	60.58	38.25	41.26	42.62	42.92
	FDR (%)	3.070	0.620	0.140	2.16	4.190	1.50	0.220	9.26	3.67	0.720	0.140	1.51
	MHD	2.525	1.001	0.643	0.522	4.784	1.964	0.579	0.324	2.619	1.156	0.818	0.729
	Time (s)	1.723	2.866	6.06	11.014	1.424	2.915	5.325	10.11	1.32	2.57	5.1	9.55
	Avg. No. Peaks	1.213	1.272	1.325	1.351	1.08	1.098	1.115	1.118	1.05	1.265	1.30	1.325
	RMSE	0.351	0.198	0.183	0.188	0.197	0.184	0.176	0.199	0.410	0.392	0.354	0.368

Table 5.2: Statistics of the Peak Estimation and Discrimination (TDR - True detection rate (%), FDR - False detection rate (%), MHD - Modified Hausdorff distance, RMSE - Root mean square error). The 2ns pulse is used for the RJMCMC approach at 600, 1200, 2400 and 4800 iterations.

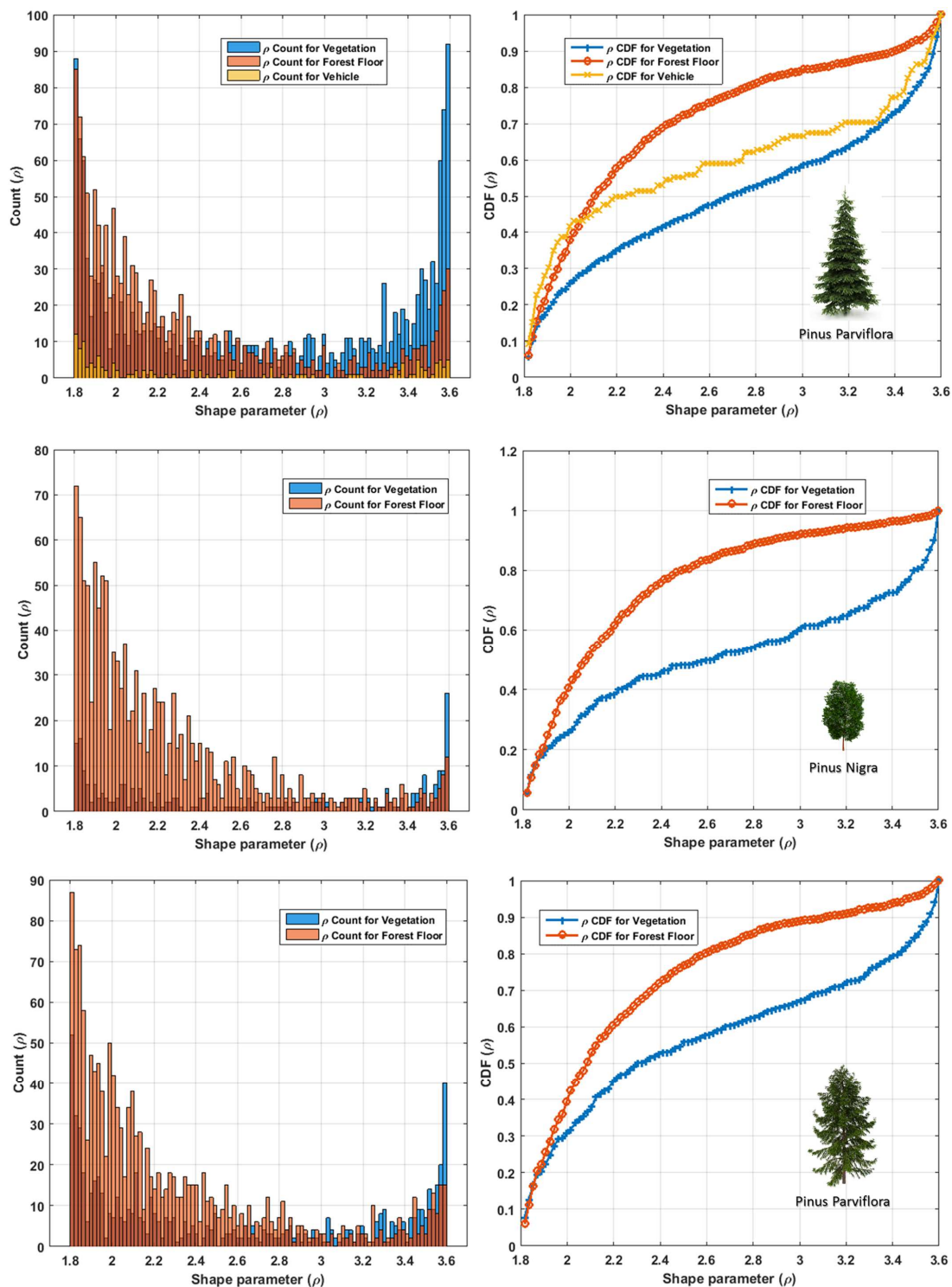


Figure 5.7: (Left column) Shape parameter ( $\rho$ ) histograms of different targets - vegetation (blue), forest floor (orange) and vehicle (yellow) for each tree species. (Right column) The cumulative peak shape  $\rho$  distribution for different targets within each scene.

<b>k</b>	<b>S</b>	<b>Expt. 1</b>	<b>Expt. 2</b>	<b>Expt. 3</b>
0.99	(100,100)	1.000	1.000	0.987
0.90	(100,100)	0.994	0.990	0.970
0.50	(100,100)	0.994	0.991	0.967
0.20	(100,100)	0.988	0.972	0.953
0.99	(10,100,100,10)	0.991	1.000	0.950
0.90	(10,100,100,10)	0.998	0.990	0.920
0.50	(10,100,100,10)	0.997	0.988	0.945
0.20	(10,100,100,10)	0.994	0.972	0.917

Table 5.3: Average prediction accuracy on unseen data

<i>k</i>	<i>S</i>	<b>Training Time (s)</b>			<b>Training Time per datapoint</b>		
		Expt. 1	Expt. 2	Expt. 3	Expt. 1	Expt. 2	Expt. 3
0.99	(100,100)	1.858	0.961	0.459	0.543	0.413	0.132
0.90	(100,100)	1.510	0.987	0.460	0.441	0.424	0.132
0.50	(100,100)	0.990	0.436	0.348	0.289	0.187	0.100
0.20	(100,100)	0.513	0.065	0.086	0.150	0.028	0.024
0.99	(10,100,100,10)	3.593	1.594	0.923	1.051	0.685	0.266
0.90	(10,100,100,10)	2.441	1.094	0.760	0.714	0.470	0.218
0.50	(10,100,100,10)	1.703	1.019	0.378	0.498	0.438	0.108
0.20	(10,100,100,10)	1.190	0.446	0.278	0.350	0.191	0.080

Table 5.4: Average training time for MLP

**Off-line Classification using MLP** - The characteristics of the LiDAR waveform peaks were captured using different parameters,  $\theta = \{A, \rho, \sigma, X\}$ , which lay the basis for the classification. Three different experiments were run. The first two experiments (1 and 2) were binary classifiers with the only two classes being vegetation and forest floor. The last experiment had four classes - leaves, bark, vehicle and forest floor. Given the relatively simple nature of the dataset (7 parameters associated with one label), a simple multi-layer perceptron (MLP) [84], a deep neural network, is used for classification. The hyper-parameters to be tuned for the implementation of the MLP are:

- **k**: Proportion of data used as training set. ( $1 - k$  is then the testing set)

- `hidden_layer_size(S)`: The number of neurons in each hidden layer.

In Table 5.3, a selection of the results from the experiments are shown. Other hyperparameter combinations were tried but the optimal ones are reported. All the neurons are activated by the 'relu' function. The optimization method chosen is the Adam optimizer with an initial learning rate of 0.001. Each experiment is run 20 times and the results are averaged over all runs. In Table 5.4, we report the time taken to train the MLP network.

### 5.3.3 | Results from Multi-Spectral Aerial LiDAR Scans

The multi-spectral dataset, used in our experiments, was collected on two separate occasions flying over a semi-urban environment in Austria. Riegl GmbH carried out these missions in 2015. The raw waveforms, in the *PulseWaves* [93] format and the 3D point cloud in the *.las*, laser format [153], was made available. The original point cloud data (*.las*) was generated using Riegl's propriety peak detection algorithm. As a pre-cursor to peak modelling, we solve the registration problem by using performing a nearest search operation for each 3D point using an Octree algorithm (Figure 5.3).

Experiments on real-world datasets show that SPeED can not only improve on the existing peak detection algorithms used by Riegl but also classify individual peaks. Similar to the simulation experiments, the characteristics of the real LiDAR waveform peaks were captured using different parameters,  $\theta = \{A, \rho, \sigma, X\}$ , which lay the basis for the classification. However, in a separate experiment, we add geometric features: anisotropy, sphericity, planarity, and linearity (introduced in Chapter 6), making  $\theta = \{A, \rho, \sigma, X, S, P, A_\epsilon, P_\epsilon, S_\epsilon, L_\epsilon\}$ .



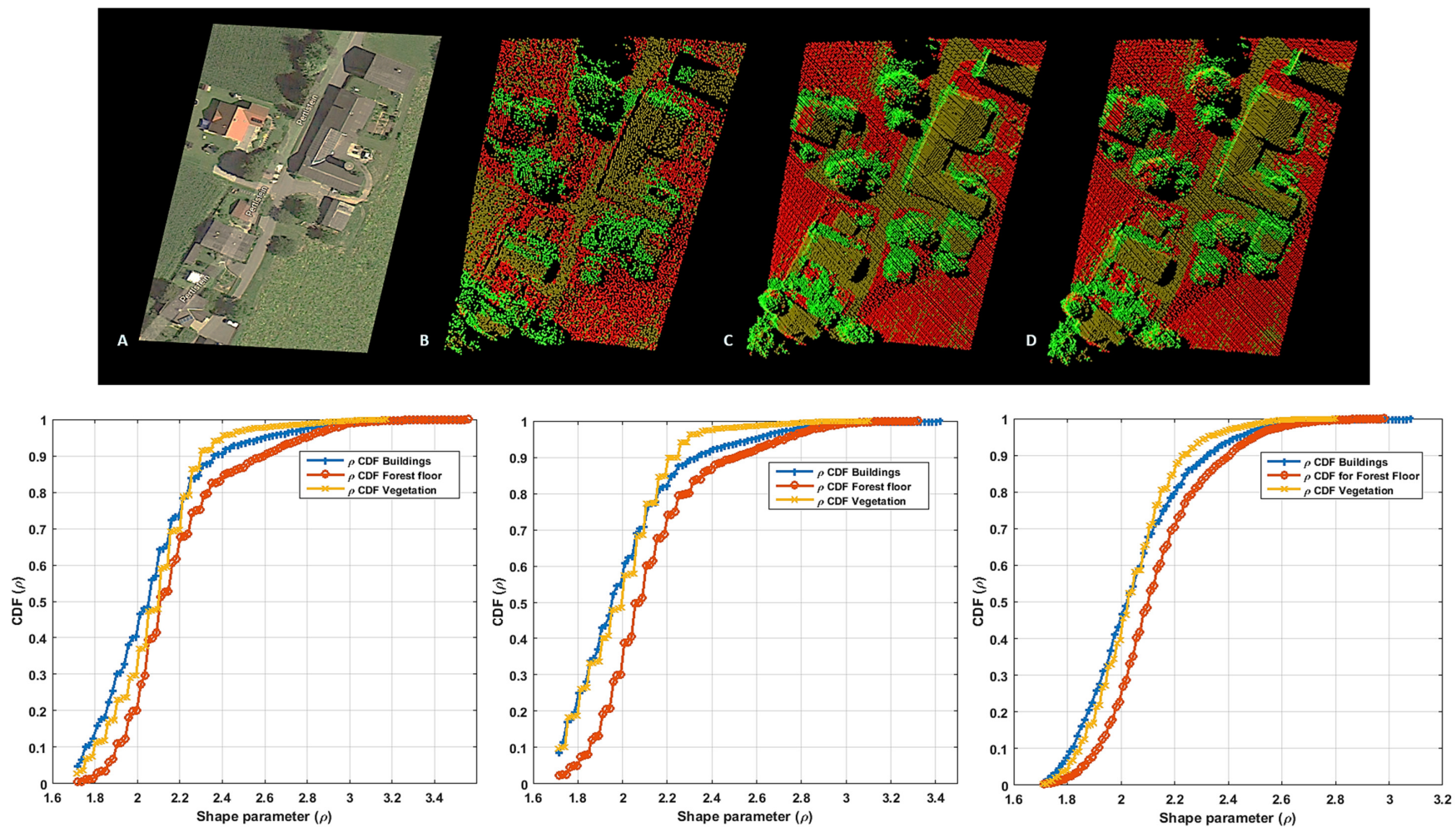


Figure 5.8: (Top) Unsupervised classification results using the SPeED algorithm on the Riegl dataset collected in Austria. (A) Observed scene, (B) 1550nm, (C) 1064nm and (D) Combined result. (Bottom) Cumulative peak shape ( $\rho$ ) distributions for different targets.



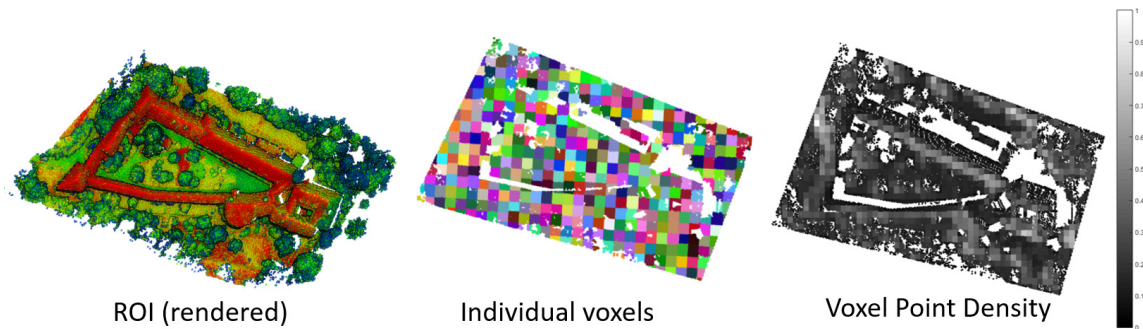


Figure 5.9: Site 2 - Bertholstein, Austria 3D point cloud classification results. (Left) - Classification results rendered and colour coded: buildings (Red), high vegetation (dark green), lower vegetation (green). (Middle) - We compute geometry features (introduced in chapter 6) in individual 3D voxels (each voxel is colour coded). (Right) Each voxel is colour coded (white - high point density and black - low point density).

**Site 1: Schwengentha, Austria** - Figure 5.8 (Top) illustrates waveform decomposition and classification results of the Schwengentha site in Austria, Figure 5.3. The scene illustrates points classified into three different classes - buildings (brown), forest floor (red) and vegetation (green) at 1550nm, 1064nm and the combination of these wavelengths, simultaneously. It is important to note that the peak decomposition or extraction and classification is carried in a single step and an unsupervised clustering methodology for classification is used; the peak dictionaries are built using a generalised Gaussians using equation (5.1.3.2). The classification approach on site 1 does not consider any geometric properties (presented in Chapter 6). The results are not quantitatively assessed as no ground-truth data is available. Figure 5.8 (Bottom) shows the cumulative peak shape ( $\rho$ ) distributions for different targets - buildings (blue), forest floor (orange) and vegetation (yellow).

**Site 2: Bertholstein, Austria** - For site 2, we classify the waveform data in two steps: i) first individual peaks are modelled and classified using the dictionary approach, SPeED (5.1); and ii) we compute geometric features (presented in chapter 6) from the point cloud data. The motivation behind this is if we get any improvement combining geometry and spectral information. We combine these in to a feature vector and classified using an unsupervised clustering approach. The results are presented in Figure 5.9. The colour coded image is rendered for better visualisation.

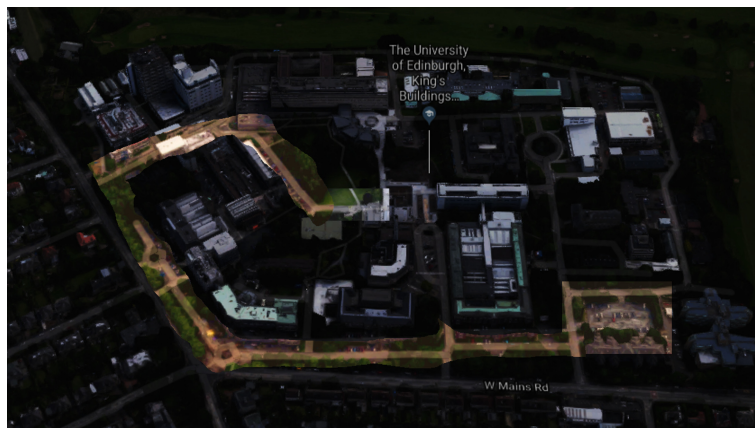


Figure 5.10: Terrestrial LiDAR data was collected at the King's buildings campus using the backpack system.

### 5.3.4 | Results on Terrestrial LiDAR Data (Edinburgh Trials)

The Riegl's VUX-1LR is a compact full-waveform LiDAR system capable of recording multiple peaks. We designed a backpack system using the VUX-1LR sensor and walked around University of Edinburgh's King's buildings. The setup is presented in to Section 5.2.1. Figure 5.10 illustrates the path taken wearing the backpack system. The sensor comes with an additional GPS receiver as an add-on and this was calibrated at the beginning of the trial in a carpark. The SPeED algorithm was applied on the raw data from VUX-1LR and geo-referenced point cloud data is shown in Figure 5.11. The colours here represent: first returns (blue), first of many (green) and last returns (orange).

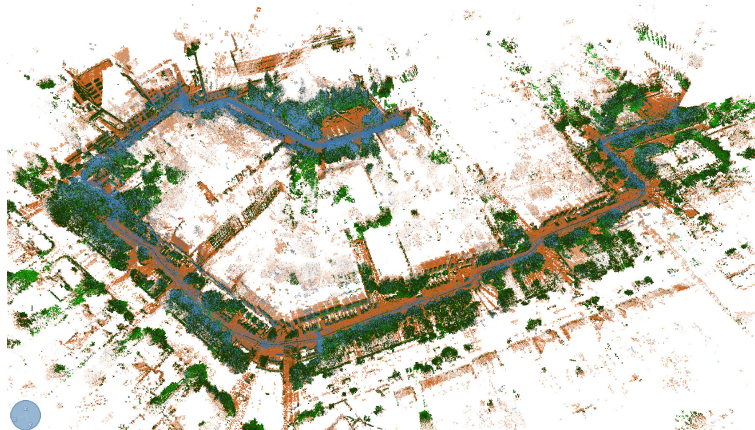


Figure 5.11: Geo-referenced point cloud data generated using the waveform processing algorithm, SPeED. The colours here represent: first returns (blue), first-of-many (green) and last returns (orange)

## 5.4 | Summary

In this work, we presented a new approach to peak modelling and classification for waveform enabled LiDAR systems. Not all existing approaches perform peak modelling and classification simultaneously on-line. Instead, they rely on a complex peak extraction method (which are not robust to multiple peaks) that leads to a 3D point cloud which can be further analysed for structure. On the contrary, the approaches that have an on-line classification stage use peak parameters - amplitude, shape and width to determine class membership. These approaches tend to enforce shape memberships which are binary. Our approach, SPeED, is significantly different, we believe that an individual peak is composed of elementary shape functions (mathematically modelled or learnt from the data) and its coefficient or contribution vector is a significant parameter in class separation. Hence, we deviate from the binary membership to an orthonormal membership provided the *dictionary* matrix is available or can be learnt from the training data. In order to quantify the robustness and efficiency of our approach, we created a ray-tracing system which generates synthetic scenes with three different tree species - *Picea Engelmannii*, *Pinus Nigra* and *Pinus Parviflora*. We compare our approach with the RJMCMC approach and report results when compared using several metrics. The improvement on the average number of peaks extracted using the SPeED algorithm when compared against the RJMCMC approach is 2.3 times and further reducing the time complexity by 234 times. This makes this approach highly suitable of real-time applications. When comparing the MCMC based methods, the MP-MCMC approach [120] (0.07s per waveform or 50,000 waveforms/hour), on a high-end 8-core machine with 6GB RAM, shows significant time reductions to the RJMCMC approach [86] and its parallel implementation [190] (1s per waveform). In comparison, our approach, on a quad-core machine with 16GB ram, reduces the time required to process each waveform by 1.8 times (0.039s per waveform or 92,000 waveforms/hour). Unlike previous methods [100, 187, 178, 177], we show that surfaces with a distance corresponding to less than 0.05m can be resolved for multiple peaks per pixel.

## Chapter 6

# Object Classification from Single Photon Counting LiDAR Signals

This chapter reports the analysis and discrimination of underwater multi-spectral single photon light detection and ranging (LiDAR) signals. We propose to use two algorithms: i) an unsupervised non-linear approach, the K-SAD method (Chapter 3), which in this particular case, uses the joint spectral and geometric representations of underwater targets as input signals. The resulting sparse coefficients, target signatures, are then used for classification; ii) a supervised approach, The SPeED algorithm that uses the multi-spectral peak parameters as input signals. Using a labelled dataset, we extract individual peaks (resulting in a 3D point cloud) and classify them, simultaneously. In addition we propose two spectrally enhanced local-shape representations: spectral spin images (SSI) and spectral shape representation (SSR). Our approach learns an optimal set of dictionary elements, *atoms* and their respective contributions, *coefficients* using a non-linear kernel. The shape and spectral adaptive quantisation of the dictionary atoms are optimised using a generalised kernel objective function for maximum discrimination. Multi-spectral single photon counting (SPC) measurements were made on scaled exemplars (known and unknown targets) placed underwater that recreate different marine environments. Finally, target material and shape discrimination results are presented.

## 6.1 | Introduction

Multi-spectral (MS) Full-waveform (FW) terrestrial and aerial light detection and ranging (LiDAR) has enabled researchers to simultaneously measure range and material reflectance. This has advantages in remote sensing [81], bathymetric mapping [54], defence and security [53], restoration and archaeology [65]. In several such cases, e.g. discrete return (DR) LiDAR, the 3D data made available lacks semantic and contextual knowledge of the surfaces and volumes of the scene. Researchers have tackled this problem using spatial filtering methods based on local regions [80] and descriptor selection [185, 136].

In order to account for occlusion and viewpoint, part-based local discriminatory models were proposed [2, 64]. Spin Images [97] is one of the most popular due to its simplicity in construction and its robustness to occlusions. Given a set of 3D point cloud, spin images creates a 2D local representation that captures shape information for each point. This work presents two novel shape representations for multi/hyper-spectral data, dubbed *spectral spin images* (SSI) and spectral shape representations (SSR). Both, SSI and SSR fuse volumetric properties of 3D objects and their spectral properties. It is only recently that LiDAR systems have been built and tested for underwater applications. Commercial and academic focus [54] on bathymetric LiDAR has been on shallow waters and use either monochromatic laser sources or a maximum of two wavelengths. This work is the first to report signal analysis and discrimination of such underwater LiDAR data for underwater mine counter-measures (MCM).

### 6.1.1 | Data Acquisition: Setup & Targets

The multi-spectral depth imaging system [114] used in this study is based on the time-of-flight (ToF) approach using a time-correlated single photon counting (TCSPC) technique. Figure 6.1a illustrates a schematic of the experimental set-up. The TCSPC module, Hydraharp in Figure 6.1 time-stamps each photon event reflecting off a target and records it using a single-photon detector. The photon counts can then be time gated in to histograms, a *full-waveform*, whose inherent nature may depend on several factors, e.g.,

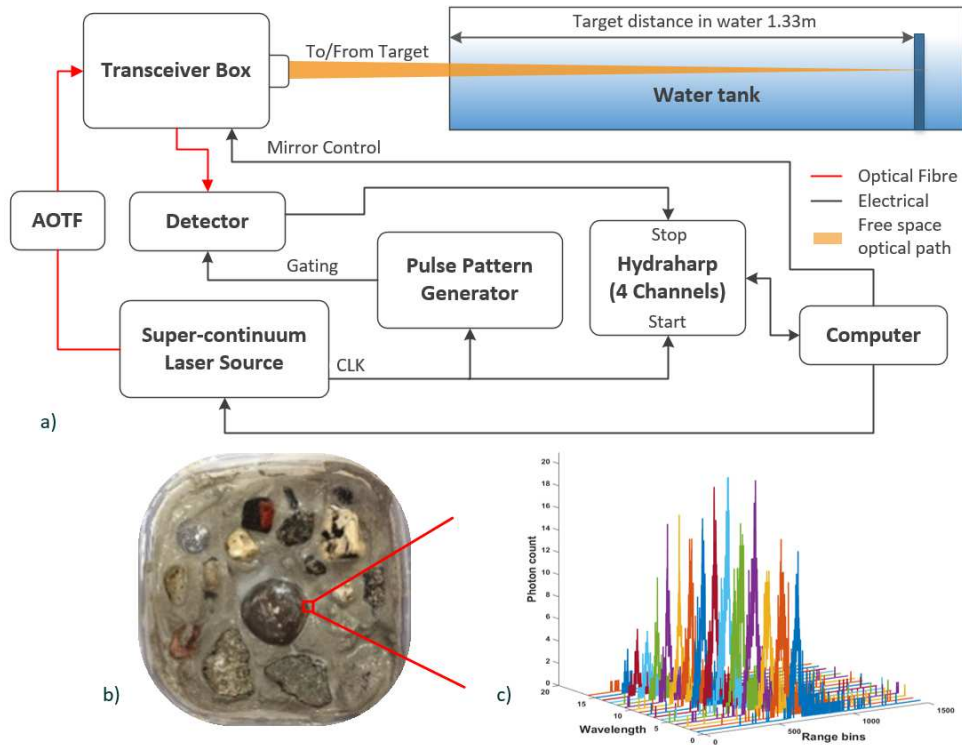


Figure 6.1: a) A schematic of the experimental set-up; b) (left) an exemplar under investigation with different targets; c) full-waveform LiDAR signatures for a single pixel.

the laser wavelength, surface geometry and transmission medium. Figure 6.1b illustrates an image of the *exemplar* investigated in a tank filled with clear unfiltered tap water.

Making such measurements is the first step to demonstrate how LiDAR could be used as an alternative to acoustic sensors for MCM purposes in very challenging environments.

## 6.2 | LiDAR Data Analysis

### 6.2.1 | 3D Shape Analysis: Point & Histogram Features

In stage 1, the following assumptions were made when processing raw multi-spectral SPC data: i) one peak per waveform was extracted at each wavelength; ii) SPC waveforms are aligned and normalised with respect to the Spectralon target, with 10% reflectance; iii) the laser beam width was less than the surface differential. For a beam width of  $300\ \mu m$ , the smallest target diameter under-investigation was around  $0.5\ cm$ ; finally, iv) a

Table 6.1: Experiment Key Parameters

Parameter	Value
Environment	Clear unfiltered tap water
Laser System	NKT Photonics supercontinuum laser source and tuneable filter fibre-coupled to the transceiver unit
Illuminating Wavelength	500nm - 725nm
Laser Beam Diameter	$\approx 300\mu\text{m}$
Laser Repetition Rate	19.5MHz
Acquisition Mode	Exemplar 1: 200 x 200 pixels Area: 5cm x 5cm Pixel acquisition time: 10ms
Histogram bin width	2ps
Histogram Length	4500 bins (after gating)
Avg. Optical Power	$\approx 300\text{nW}$

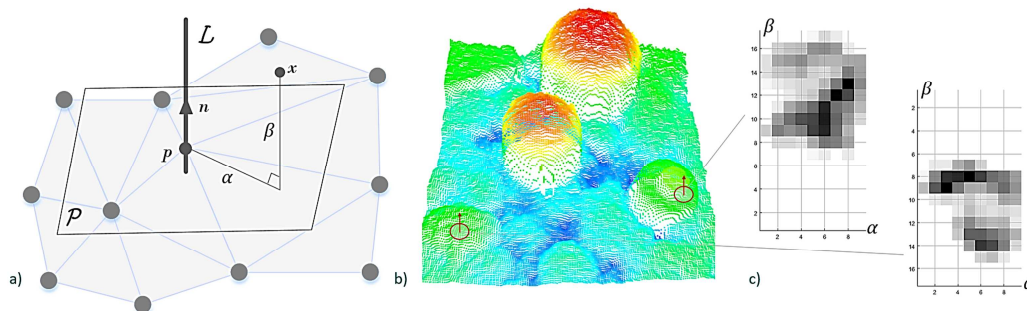


Figure 6.2: (a) The cylindrical co-ordinate system and its oriented-point basis. (b) Two different oriented-points and their neighbourhood points shown with a red ellipse on one of the exemplars used. (c) Spin maps for the two oriented-points.

fixed stand-off distance from the sensor is assumed and the sensor is stationary.

### 6.2.2 | Spin Images

The FW processing module outputs a dense 3D point cloud,  $\mathbf{B} = \{x_1, \dots, x_N\}$ , where  $x_N \in \mathbb{R}^{1 \times 3}$ . In order to capture local surface variations Spin Images (SI) [97] are computed on the point cloud  $\mathbf{B}$ .

SI is a technique that maps all the vertices  $\mathbf{B} \in \mathbb{R}^3$  onto  $\mathbb{R}^2$  based on an *oriented-point*

basis  $(p, n)$ . Oriented-points, or a *local-reference coordinates* are 3-D surface points which have a normal vector associated, i.e. direction. The normals are computed for all the points in  $\mathbf{B}$  around a local-region, controlled using radius,  $r$ . Using  $(p, n)$  a local 2-D basis is formed which corresponds to a local-reference coordinate systems. Figure 6.2(a) illustrates a tangent plane  $\mathcal{P}$ , an oriented point  $p$  with normal  $n$  and the line  $\mathbb{L}$  through  $p$ . The point  $p \mapsto \{x, y, z\}$  is mapped onto a new 2-D coordinate system  $(\alpha, \beta)$ ; where,  $\alpha$  is the non-negative perpendicular distance to the normal line  $\mathcal{L}$  and  $\beta$  is the signed perpendicular distance to the tangent plane  $\mathcal{P}$ . For each oriented-point in  $\mathbf{P}$  a spin map,  $\mathcal{S}_{\mathcal{P}}$  is created. The mapping function of a spin map,  $\mathcal{S}_{\mathcal{P}} \mapsto (\alpha, \beta)$  can be expressed as:

$$\left( \sqrt{\|x - p\|^2 - (n \cdot (x - p))^2}, n \cdot (x - p) \right). \quad (6.2.2.1)$$

The projection function of Eq. (6.2.2.1) is applied to all the vertices of a 3D point cloud resulting in a set of points in  $(\alpha, \beta)$  coordinates. Only one oriented-point is required in order to express all the 3D points in a local neighbourhood. Once the mapping is computed, 2D points  $(\alpha, \beta)$  are accumulated into discrete bins. The maximum size of an object expressed in oriented point coordinates is controlled by  $\alpha_{max}, \beta_{max}$ . These are computed by taking the maximum of all the values of  $\alpha$  and  $|\beta|$  of all the different oriented-point bases. The bin size,  $b$ , which effects the spin image size, should be chosen to be large enough so that the local descriptiveness of an object remains relatively high. The spin image size  $(i, j)$  can be expressed as:

$$i = \frac{2\beta_{max}}{b} + 1, \quad j = \frac{\alpha_{max}}{b} + 1 \quad (6.2.2.2)$$

All the neighbouring points to an oriented-point basis are accumulated into discrete bins in the following manner:

$$i = \left\lceil \frac{\beta_{max} - \beta}{b} \right\rceil, \quad j = \left\lceil \frac{\alpha}{b} \right\rceil \quad (6.2.2.3)$$

In [97], authors suggest a bi-linear interpolation scheme to handle outliers, ensuring the



spin images are not sparse to a large degree. The contribution of a 2-D point is spread to the four surrounding bins using a bi-linear interpolation technique. The weights that are used to increment the bins are:

$$a = \alpha - ib, \quad b = \beta - jb \quad (6.2.2.4)$$

Figure 6.2(b-c) illustrates two local oriented-point bases and their corresponding greyscale spin images. Each 3D point  $x_N$  has an associated spin image which embedded neighbouring geometric properties of a surface in a 192 dimension feature vector,  $S_B$ . There are other methods by which regional geometric variation can be captured, see below.

### 6.2.3 | Curvature Representation (CR)

The FW processing leads to a dense 3D point cloud. In order to capture local surface variations, a regional variance-covariance matrix was computed on the 3D point cloud. The radius,  $r$ , of the region shown in Figure 6.3 can be altered depending on the point cloud density. West et al. [183] show how Eigenvalues can be used to describe the local, spatial distribution of the 3D point cloud. They compute discrete moments within a neighbourhood that can describe the planarity, linearity, sphericity and anisotropy.

Table 6.2 illustrates how these properties are computed, provided the Eigenvalues  $\mathcal{E}_1 > \mathcal{E}_2 > \mathcal{E}_3$ . The Eigenvalues computed are rotational [183] and view-point invariant. Four local 3D surface features, Anisotropy,  $A_{\mathcal{E}}$ , Planarity,  $P_{\mathcal{E}}$ , Sphericity,  $S_{\mathcal{E}}$  and Linearity,  $L_{\mathcal{E}}$

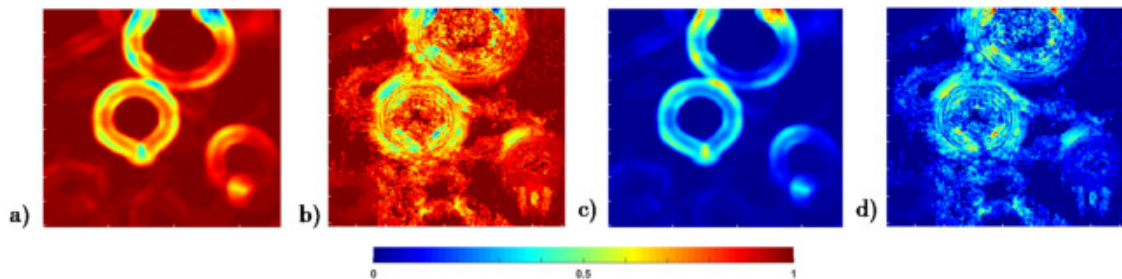


Figure 6.3: Raster images of different geometric features computed for each 3D point, controlled using a local neighbourhood radius,  $r$ . (a) Anisotropy  $A_{\mathcal{E}}$ , (b) Planarity  $P_{\mathcal{E}}$ , (c) Sphericity  $S_{\mathcal{E}}$  and (d) Linearity  $L_{\mathcal{E}}$  Table 6.2.

Table 6.2: Curvature Representations using Eigenvalues

Linearity $L_{\mathcal{E}}$	$\frac{\mathcal{E}_1 - \mathcal{E}_2}{\mathcal{E}_1}$	Sphericity $S_{\mathcal{E}}$	$\frac{\mathcal{E}_3}{\mathcal{E}_1}$
Planarity $P_{\mathcal{E}}$	$\frac{\mathcal{E}_2 - \mathcal{E}_3}{\mathcal{E}_1}$	Anisotropy $A_{\mathcal{E}}$	$\frac{\mathcal{E}_1 - \mathcal{E}_3}{\mathcal{E}_1}$

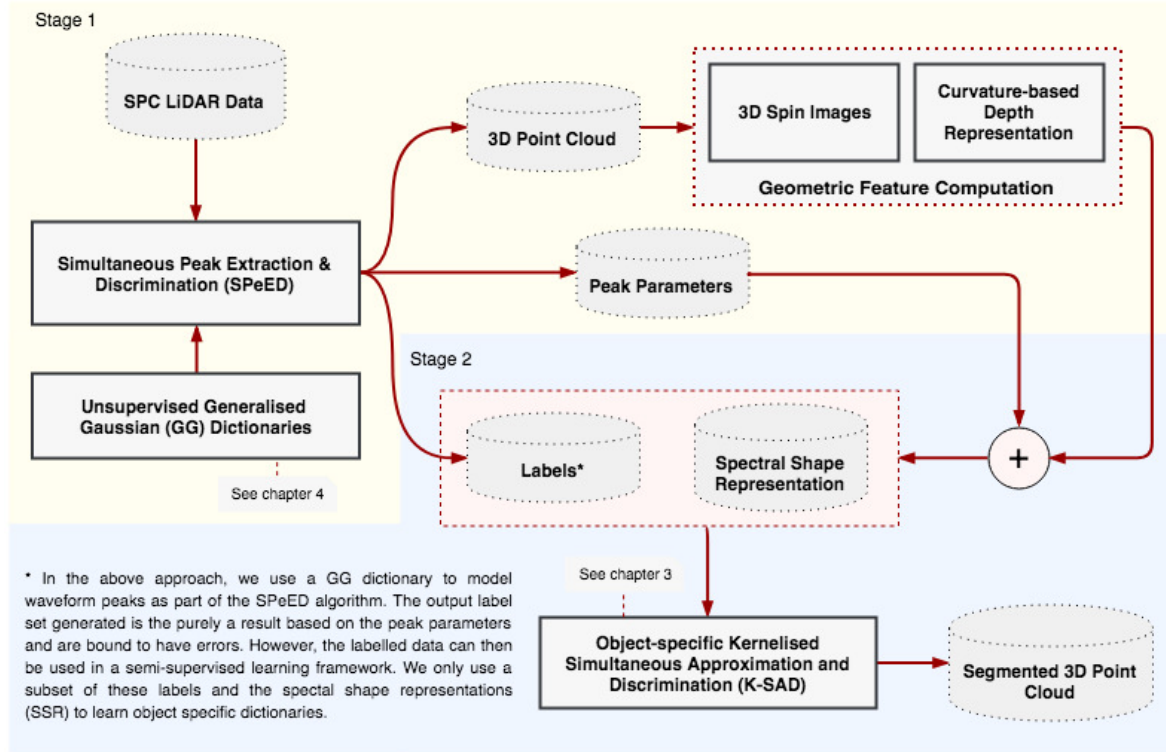


Figure 6.4: The adopted approach to process multi-spectral TCSPC LiDAR data. Figure 5.1 illustrates the flowchart for the supervised algorithm, SPeED and algorithm 3 lists the pseudo-code for the K-SAD algorithm. In this chapter, as noted in the infogram above, the K-SAD algorithm is used as a semi-supervised classification algorithm.

are computed within a neighbourhood, governed by radius  $r$ , of each 3D point. Finally, the depth  $D_z$  per pixel completes the SSR vector.

### 6.3 | TCSPC Object Discrimination Algorithm

The approach, illustrated in Figure 6.4, is divided into two stages: i) Stage 1 - An unsupervised method for point cloud computation and geometric feature computation; and ii) Stage 2 - A semi-supervised method for point cloud classification using spectral shape representations.

### 6.3.1 | Stage 1 - Waveform Processing

We propose the idea of combining spectral (material) signatures with geometric cues. Spectral response and the depth is computed by processing the raw LiDAR waveform using the SPeED algorithm. Curvature based geometric cues are computed (Section 6.2.3) on the resulting point cloud.

We generate the 3D point cloud using the SPeED (Chapter 4) algorithm followed by the computation of geometric features (Section 6.2.3). The steps involved in the 3D point cloud generation are:

1. Pre-process waveform by smoothing (improve signal-to-noise ratio) the original waveform with a Gaussian of pre-defined half-width (Step 1 in Figure 5.1).
2. Check for the stop criteria. The algorithm stops if the maximum number of peaks is reached or the reconstruction error is below a certain threshold (Step 2 in Figure 5.1).
3. Find inflection points  $n$ . This implies that the waveform will be decomposed in to

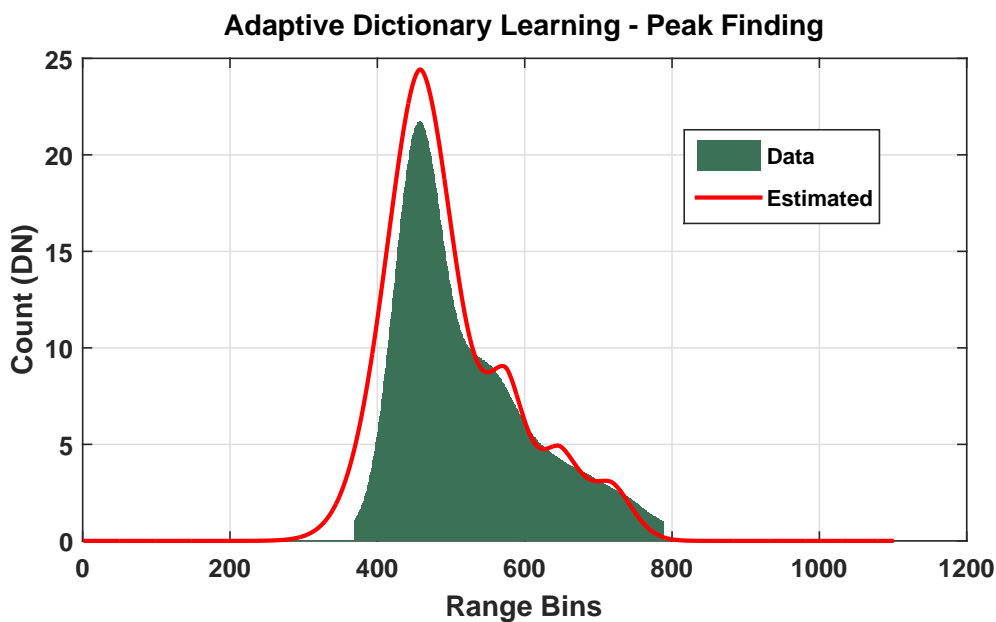


Figure 6.5: Multiple peaks estimated (red curve) using the SPeED algorithm on the original SPC histogram (green).

$n$  elementary components. We initialise the peak parameters, estimate background noise and rank them (Steps 3 - 6 in Algorithm 5.1).

4. If a pre-defined set of elementary functions, a dictionary,  $\mathbf{D}$ , is known, we solve Eq 5.1.3.1. However, if a dictionary is not known, we use the initial peak parameters of the top ranked peaks (Step 6, Figure 5.1), to generate an adaptive dictionary Eq. (5.1.3.2),  $\mathbf{D}$ , (Step 7, Figure 5.1).
5. Solve Eq. (5.1.3.1) and classify peak (Steps 8 - 9, Figure 5.1).
6. Remove the peak from the original waveform and go to Step 2.

**Geometric Feature Computation** In Chapter 4, a generalised Gaussian (GG) dictionary matrix was used to process aerial and terrestrial LiDAR data. The transmitted beam by the super-continuum laser source in our experiments (shown in Figure 6.1) was an exponential pulse. Figure 6.5 illustrates one such return, green curve. An exponential pulse has the form

$$f(t) = k(e^{-t\mathcal{T}_1} - e^{-t\mathcal{T}_2}), \quad (t \geq 0),$$

$$\text{where, } k = \frac{\mathcal{T}_2 e^{t_p \mathcal{T}_1}}{(\mathcal{T}_2 - \mathcal{T}_1)}, \text{ and } t_p = \frac{\ln(\mathcal{T}_2/\mathcal{T}_1)}{(\mathcal{T}_2 - \mathcal{T}_1)}, \quad (6.3.1.1)$$

and the degree of modulation on the backscattering beam depends on the surface geometry and its spectral reflectance. For the SPeED algorithm, a dictionary matrix is generated which is build using the GG Eq. (5.1.3.2) and the exponential model Eq. (6.3.1.1).

For the analysis of urban and forest scenes using laser scanning, several classification and feature relevance algorithms [81, 183] have been proposed. However, these methods do not truly embed the full-waveform properties and the spectral reflectance of the observed objects. The proposed SSR representation captures such variations from the waveform and the point cloud data. For each waveform the peak parameters (Eq. 5.1.2.1 and 5.1.3.2) are extracted using the SPeED algorithm.

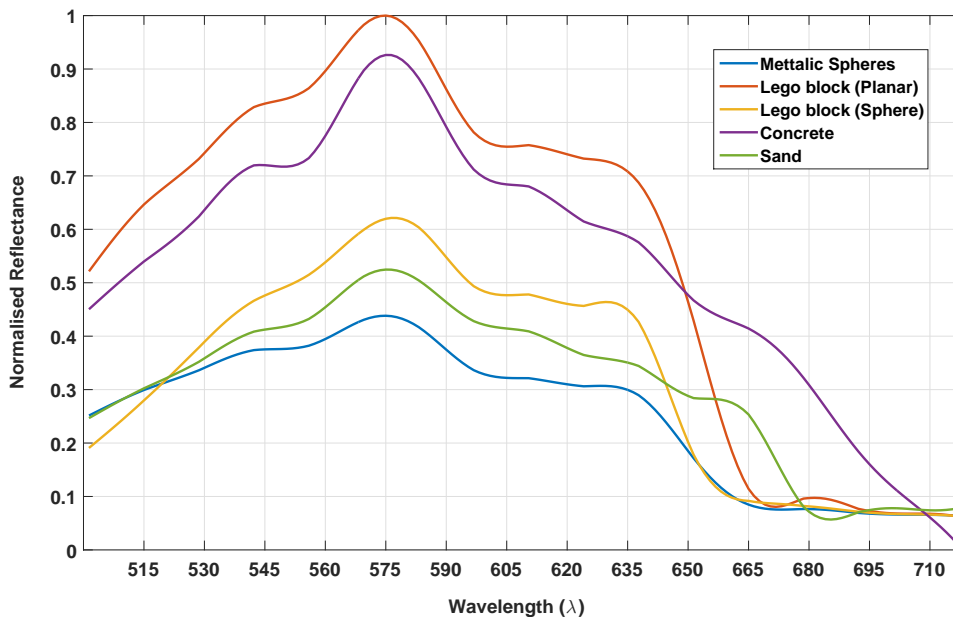


Figure 6.6: Reflectance spectra extracted using our technique for different materials (plastic, metal, sand, concrete).

### 6.3.2 | Stage 2 - Non-linear Dictionary Learning of SSR

The three spectral parameters,  $\rho$ ,  $\sigma$  and  $\beta$  in Eq. 5.1.3.2 is combined with the spin images (SI), Section 6.2.3 and curvature representation (CR), Section 6.2.3 to form the Spectral Spin Image (SSI) and the Spectral Shape Representation (SSR), respectively.

A combined per-pixel representation with SSI can be written as:

$$\{y\}_{n=1}^N = \left[ \{\beta_\lambda\}_{\lambda=1}^\Lambda, \{\rho_\lambda\}_{\lambda=1}^\Lambda, \{\sigma_\lambda\}_{\lambda=1}^\Lambda, S_B \right] \in \mathbb{R}^{240} \quad (6.3.2.1)$$

Alternatively, when SSR is used, the combined per-pixel representation is written as:

$$\{y\}_{n=1}^N = \left[ \{\beta_\lambda\}_{\lambda=1}^\Lambda, \{\rho_\lambda\}_{\lambda=1}^\Lambda, \{\sigma_\lambda\}_{\lambda=1}^\Lambda, A_\epsilon, P_\epsilon, S_\epsilon, L_\epsilon \right] \in \mathbb{R}^{52} \quad (6.3.2.2)$$

In 6.3.2.1,  $\rho$ ,  $\sigma$  and  $\beta$  correspond to the spectral features and  $S_B$  corresponds to the spin images, Section 6.2.3. Eq. 6.3.2.2 is identical to 6.3.2.1, except we replace the  $S_B$  vector with four scalars, represented by  $A_\epsilon$ ,  $P_\epsilon$ ,  $S_\epsilon$ ,  $L_\epsilon$  and  $D_z$ , as discussed in Section 6.2.3.

For each pixel, the combined spectral and depth representation is a vector of length 240 and 52 for SSI and SSR, respectively. For the experiments reported here,  $\Lambda = 16$ . The

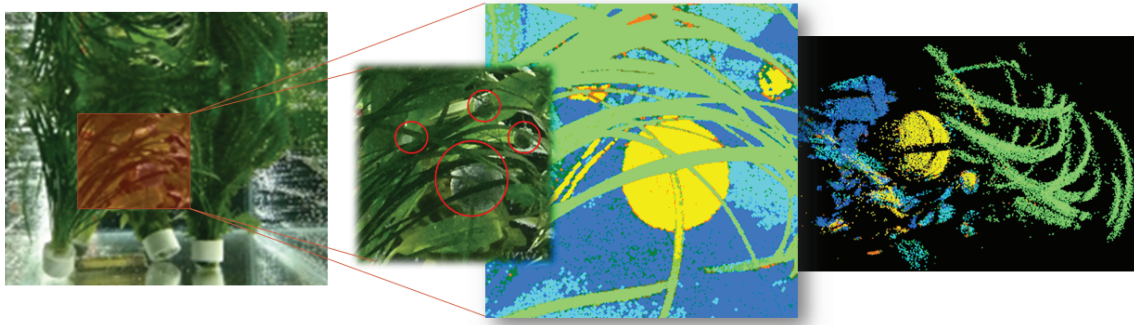







Figure 6.7: Results of underwater foliage penetration and point cloud classification.

stage 1 outcome, a label set, is purely based on the waveform parameters,  $\rho$ ,  $\sigma$  and  $\beta$ . Classification purely based on spectral properties is bound to have errors. This is also evident from the multi-spectral aerial LiDAR scans (Section 5.3.3). However, a subset of randomly selected pixels and their labels can be used to train a semi-supervised learning framework. We use only a small subset of pixels, the spectral shape representations (SSR) and their corresponding labels to learn object-specific dictionaries and classify them using the K-SAD algorithm (See Algorithm 3).

## 6.4 | Results and Discussion

The discriminatory performance of our approach is analysed using several handmade in-house exemplars. These are listed in Table 6.3. Target Tin\_1 - Tin\_4 have several objects of varied material, listed in Table 6.4, securely attached to the surface of a mason jar tin. We use Float\_1 target to analyse the classification accuracy when targets are hidden behind dense foliage underwater. Metallic balls and plastic objects were suspended using a fishing thread behind aquatic plants. Classification results are illustrated in Figure 6.7.

Table 6.3: Targets under investigation. These are designed using materials (Table 6.4)

<b>Image</b>					
<b>Target ID</b>	Tin_1	Tin_2	Tin_3	Tin_4	Float_1

Two sets of experiments were carried out: i) *Material Discrimination*: classify target signatures into three constituent materials, *sand*, *plastic* and *metal*, used to make the exemplar; ii) *Mine Discrimination*: classify different mine types, which not only differ in shape but also in material. Four different mines were used, *Plastic 1*, a cuboid shaped mine, *Plastic 2*, a spherical shaped mine, *Metal 1* and *Metal 2*, small and large spherical shaped metallic mines, respectively.

### 6.4.1 | Experiment 1 - Material Discrimination

The confusion matrix for material discrimination is shown in Table 6.5. A subset of 8520 target signatures, equally divided into three different materials was selected. A 10-fold cross-validation classification was then performed using the proposed approach resulting in a mean classification error rate of **0.021%**. Table 6.1 summarises the sensor equipment and acquisition parameters used. The exemplar was lowered underwater and kept at a distance of 1.33m from the SPC sensor. In order to limit the influence of any ambient illumination, the experiment was conducted in a dark room. The instrumental response was measured using a reference, a Spectralon panel, which was placed approximately at normal incidence to the beam. For each pixel a 4500 bin time-gated histogram is created with a depth resolution of  $300\mu m$ . Such measurements are repeated for 16 wavelengths ranging between 500nm to 725nm, equally spaced by 15nm. The choice of the wavelengths hinges on a detailed study previously performed. A detailed explanation of a similar experiment is reported in [114].

### 6.4.2 | Experiment 2 - Mine Discrimination

The aim of this experiment was to seek answers for the following questions: i) can mines with structural variation but similar spectral signatures be classified correctly?, and, ii) what impact do the *Geometric* features, Section 6.2.3, have on classification? Table 6.7 lists the confusion matrix for mine classification when geometric properties are included, using the full SSR representation, Eq. 6.3.2.2 or 6.3.2.1. A subset of 7350 target



	<b>Material</b>	U-can Mortar
	<b>Manufacturer</b>	TBP Ltd, UK
	<b>Used in</b>	Tin_2 and Tin_4
	<b>Material</b>	Textured spray paint
	<b>Manufacturer</b>	The Valspar Corp, UK
	<b>Used in</b>	Tin_1- Tin_4 and Float_1
	<b>Material</b>	Waterproof glue
	<b>Manufacturer</b>	Gorilla Glue Europe, UK
	<b>Used in</b>	Tin_1- Tin_4 and Float_1
	<b>Material</b>	Sure Catch Power Line
	<b>Manufacturer</b>	SureCatch World, Malaysia
	<b>Used in</b>	Float_1
	<b>Material</b>	Building sand
	<b>Manufacturer</b>	B&Q, UK
	<b>Used in</b>	Tin_1
	<b>Material</b>	Mason tin lids
	<b>Manufacturer</b>	ASDA, UK
	<b>Used in</b>	Tin_1- Tin_4
	<b>Material</b>	Lego block
	<b>Manufacturer</b>	The Lego Group, Denmark
	<b>Used in</b>	Tin_1, Tin_3 and Float_1
	<b>Material</b>	Lego block
	<b>Manufacturer</b>	The Lego Group, Denmark
	<b>Used in</b>	Tin_1 and Tin_3
	<b>Material</b>	Metallic spheres
	<b>Manufacturer</b>	RS Components, UK
	<b>Used in</b>	Tin_1- Tin_4 and Float_1

Table 6.4: List of all the materials used in our experiments.

signatures, equally divided into five different classes was selected. The mean classification error went up by **3.6%** when geometry based DR was neglected. The effect on accuracy of classification for four different mines with and without DR is listed in Table 6.6. Figure 6.8 illustrates the learnt coefficients,  $\mathbf{Q}$ , clustered into different mine types. The 3D point cloud, shown within, is segmented not only on the basis of their spectral content but also

Table 6.5: Confusion Matrix - Material Discrimination

	<b>Sand</b>	<b>Plastic</b>	<b>Metal</b>
<b>Sand</b>	<b>0.9721</b>	0.0144	0.0133
<b>Plastic</b>	0.0151	<b>0.9823</b>	0.0024
<b>Metal</b>	0.0140	0.0035	<b>0.98239</b>



Table 6.6: Effect of *Depth Representation* (DR) on accuracy

	Plastic 1	Plastic 2	Metal 1	Metal 2
Without DR(%)	92.65	95.65	97.62	98.10
With DR(%)	<b>97.55</b>	<b>99.05</b>	<b>99.46</b>	<b>98.91</b>

Table 6.7: Confusion Matrix - Mine Discrimination

	Plastic 1	Plastic 2	Metal 1	Metal 2	Sand
Plastic 1	<b>0.9755</b>	0	0.0020	0	0.0224
Plastic 2	0.0054	<b>0.9905</b>	0.0007	0.0020	0.0014
Metal 1	0.0014	0.0027	<b>0.9946</b>	0	0.0014
Metal 2	0.0014	0.0068	0	<b>0.9891</b>	0.0027
Sand	0.0102	0.0007	0	0.0007	<b>0.9884</b>

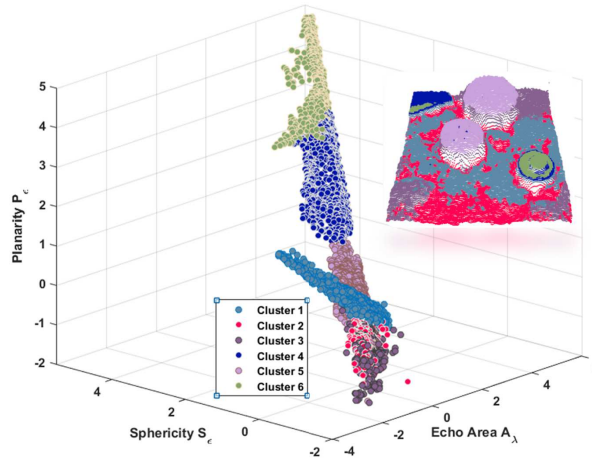


Figure 6.8: Mine clusters using the spectral depth codes,  $\mathbf{Q}$ , on the spectral axis, echo area; and the depth axis, linearity and sphericity, respectively.

their geometric features. For illustration purposes, the clusters are plotted along three dimensions, area-under-curve,  $A_\lambda$ , sphericity,  $S_\mathcal{E}$  and linearity,  $L_\mathcal{E}$ , respectively.

## 6.5 | Summary

A novel spectral-depth representation is presented that is highly discriminatory in characterising different target signatures underwater. Several custom-made realistically scaled exemplars with known and unknown targets have been investigated using a multi-spectral single photon counting LiDAR system. Multi-spectral measurements were made underwater on targets with different shapes and materials, Section 6.1.1. Using the proposed spectral shape representation sparse codes are optimised for maximum discrimination between different materials and mines, demonstrating accuracies of **97.8%** and **98.7%**, respectively. Combining depth with spectral data, the approach is very effective at discriminating targets of different shapes, but with similar spectral response, or conversely of similar shape but having different spectra. When spectral features alone are considered, the discrimination error reported for Plastic 1 mine is 7.35%. But, when spectral and depth representation is considered, the error reduces to **2.45%**, Table 6.6. This work is the first to report the analysis and discrimination of multi-spectral underwater single photon counting LiDAR signals as an alternative to acoustic MCM.

# Chapter 7

## Object Classification from Sonar using Dictionary Selection

Modern day sonar systems operate at high data rates and pixel resolution which may result in information overload for human operators, but conversely provide sufficient temporal and spatial resolution for automatic target recognition (ATR). This work proposes an approach that finds efficient representations for training and classification of different mine like objects (MLOs) and natural rocks in underwater signals, e.g. sonar echoes and images. The focus is on the design and selection of a compact, optimal and non-linear subspace, a dictionary, based on the mathematical models of acoustic signals. Here, the traditional sparse approximation formulation is decoupled and modified by an additional discriminating objective function and a corresponding selection strategy is proposed. During training, using a set of labelled sonar signals, a single optimised discriminatory dictionary is learnt which can then be used to represent MLOs. During classification, this dictionary together with optimised coefficient vectors is used to label scene entities. Evaluation of our approach has resulted in classification accuracies of 96%, 95% and 94% on sonar echoes (the Connectionist Sonar benchmark dataset), realistic synthetic side-scan images and real Synthetic Aperture Sonar (SAS) data, respectively.

Unlike the methods briefly discussed in Section 2.1, this work focusses on a signal approximation and discrimination approach that applies to both 1-D and 2-D data. Our

approach learns an optimised lower-dimensional subspace of labelled training data, making it quite adaptable to classify unseen target signatures. The mathematical representation proposed is fairly simple, easy to implement and does not require highly complex classification routines. The work presented here is developed from an earlier approach [53] to classify various tree species using synthetic 1D photon count histograms. Here it has been extended to represent and classify multiple types of targets in both 1D sonar echoes and 2D sonar images.

A two stage approach to underwater sonar signal discrimination is proposed. Experiments are also carried out on 1-D sonar signals and 2-D imagery. In the 1-D case, the clutter rejection problem is tackled, i.e. differentiating between backscattered sonar returns from man-made objects, e.g. mines, and naturally occurring sea-floor, e.g. rocks. In the 2-D case, the dictionaries are adapted to curve singularities and strong gradients, e.g. extracting features from strong contrast caused due to target shadows underwater.

Below we highlight some of our findings when the SAD algorithm was applied to 1D and 2D sonar datasets:

- A generalised algorithm for traditional 1D sonar returns and more advanced 2-D sonar imagery is proposed. This approach thus decouples the signal coding stage from the dictionary selection stage.
- In contrast to some of the work in dictionary learning [188], [4] and [186] where dictionary atoms are learnt from the data, a dictionary building and selection approach based on mathematical models of underwater sound signals is suggested in section 7.2.
- We use the objective function, presented in chapter 3, that maximises the inter-class discrimination and minimises the intra-class distance of MLOs.
- This work reports 96% accuracy on the Connectionist sonar dataset [74]. This is reported in section 7.3.
- A classification accuracy of 95% is also reported on the simulated side-scan sonar

image dataset, Section 7.4.1; and, real SAS data coming from the COLOSSUS 2 trials operated by the CMRE (Centre for Maritime Research and Experimentation), section 7.4, respectively.

- With a 23.24% decrease in the misclassification error rate on the CMRE MUSCLE SAS imagery, the impact of dictionary selection and merging is discussed in section 7.5.

## 7.1 | Problem Formulation

**Discriminatory Sparse Representation** We formulate the MLO classification problem as a SAD problem (see (4.2.3.1)). Let  $\mathbf{Y}$  be a data matrix in a P-dimensional space, i.e.,  $\mathbf{Y} = [y_1, \dots, y_N] \in \mathbb{R}^{P \times N}$ . Each signal within  $\mathbf{Y}$  can be sparsely represented using a linear combination of the columns in a dictionary matrix,  $\mathbf{D} = [d_1, \dots, d_K] \in \mathbb{R}^{K \times P}$ .

The solution to a sparse representation can then be written as an optimisation problem. In (4.2.3.1), the codebook  $\mathbf{D}$  is a *dictionary* matrix, where  $K \gg P$ . The term  $G(\mathbf{X})$  is a discriminatory function that minimises inter-class and maximises intra-class variance. Detailed explanation on  $G(\mathbf{X})$  and step by step implementation of SAD is given in section 3.1.1. The penalising terms,  $\beta_1$  and  $\beta_2$ , control the importance of the sparsity constraint and the reconstruction error, respectively.

The solution to (4.2.3.1) can be divided into two steps: i) solve (4.2.3.1) with respect to  $\mathbf{X}$  and  $\mathbf{D}$  and the dictionary, i.e, codebook,  $\mathbf{D}$  is retained; ii) for each test signal, the coefficients are obtained by optimising (4.2.3.1) with respect to  $\mathbf{X}$  only. The individual modules within each stage are explained in detail below along with their implementation details.

## 7.2 | Proposed approach

A MLO classification system with discriminative traits is proposed and shown in Figure 7.1. Pseudocode is presented in algorithm 6 and is divided into two stages:

Stage 1 - Training

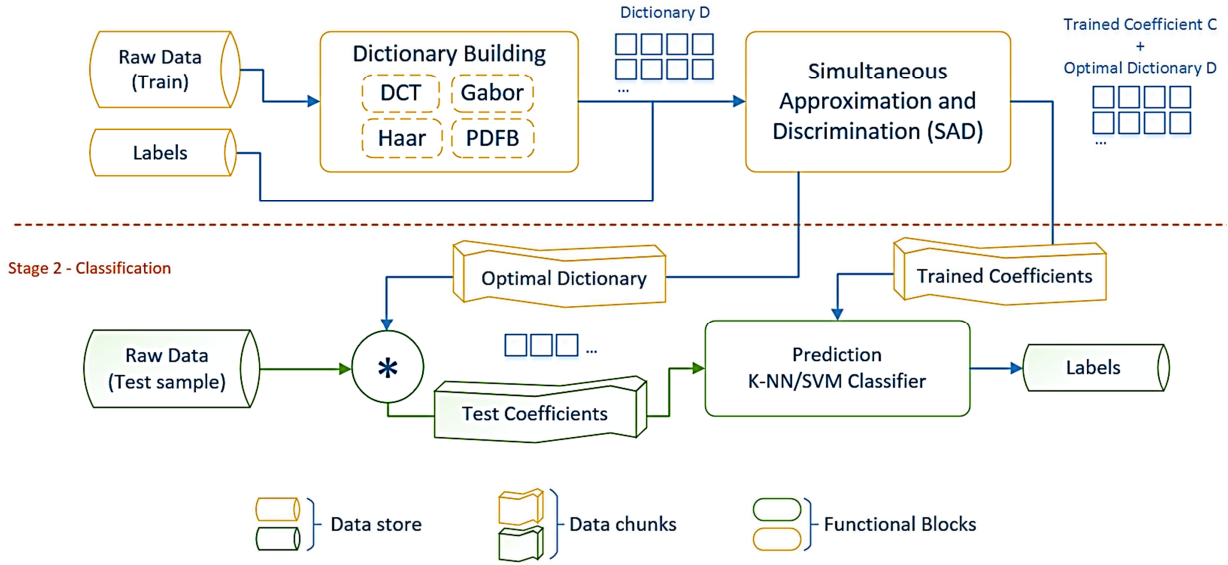


Figure 7.1: The proposed system has been adapted to the MLO classification problem by encompassing a dictionary building routine specifically for underwater sonar signals. The prediction module is a generic classifier with class labels as its output.

1. **Training:** The optimisation stage, section 7.2. Stage 1 takes the sonar signals along with their labels and returns a single optimised discriminatory dictionary for all mine targets and their coefficients. This stage is further divided in to two modules:
  - a) *Dictionary Building* - a large set of orthogonal bases are defined based on mathematical models of MLO. This information is used as a known prior in order to create a large pool of dictionary atoms from which an optimised subset can be selected. See section 7.2.
  - b) *Simultaneous Approximation and Discrimination* - a modified discriminatory constraint is proposed that allows approximation and discrimination of the input signals with a small subset of dictionary atoms. The learnt discriminatory dictionary and their optimised coefficient vectors along with their labels are used by stage 2 for classification. See Section 3.1.1.
2. **Classification:** Each test signal is classified into different MLO categories using the learnt dictionaries and coefficients from stage 1. Finally, confusion matrices and misclassification error plots are generated by comparing the predicted label set

---

**Algorithm 6:** PSEUDO-CODE - PROPOSED APPROACH

---

**Input:** Data matrix  $\mathbf{Y} \in \mathbb{R}^{P \times N}$ , labelled set,  $\Omega$   
**Output:** Dictionary  $\mathbf{D}$ , Coefficients  $\mathbf{X}$  and  $L_{train}$ , training label set

```

1 begin
  // Stage I - Dictionary Building
2   Build dictionaries using Algorithm 7
  // Stage II - SAD (Optimisation)
3   Solve (4.2.3.1) using Algorithm 2
  // Stage III - Prediction
4   Classify test samples using steps 5,6 and 7 of Algorithm 2

```

---



---

**Algorithm 7:** PSEUDO-CODE - DICTIONARY BUILDING

---

**Input:** Signal dimension,  $P$ , number of atoms,  $K$   
**Output:** Dictionary matrix,  $\mathbf{D}_{init}$

// Stage I - Dictionary Building

```

1 Dictionary matrix  $\mathbf{D}_{init} \leftarrow []$ 
2 begin
3    $\mathbf{D}_{dct} \leftarrow \text{MakeDCT}(K, P)$  // (7.2.0.3)
4    $\mathbf{D}_{gabor} \leftarrow \text{MakeGabor}(K, P)$  // (7.2.0.4)
5    $\mathbf{D}_{haar} \leftarrow \text{MakeHaar}(K, P)$  // (7.2.0.5)
6    $\mathbf{D}_{pdfb} \leftarrow \text{MakePDFB}(K, P)$ 
7    $\mathbf{D}_{init} \leftarrow [\mathbf{D}_{dct} \ \mathbf{D}_{haar} \ \mathbf{D}_{gabor} \ \mathbf{D}_{pdfb}]$ 

```

---

against known ground truth.

### Stage I - Dictionary Building

The pseudo-code for stage 1 is presented in Algorithm 7. A crucial step in the proposed MLO classification is to design an appropriate dictionary. A dictionary matrix with maximum “compactness” or “sparseness” properties can be considered as a transformation matrix, synthesised from a filter bank that represents the input data in a different domain, e.g., frequency or time-frequency. Representing input signals for recognition purposes involves dictionary building and selection [38]. Once a dictionary is selected, for the orthogonal case, the coefficients can be computed as an inner product of signal and dictionary atoms. However, in a non-orthogonal case, the coefficients are the inner product of the signal and the dictionary inverse. An *over-completeness* of the dictionary is assumed, i.e, the number of atoms is far greater than the signal dimension, a large pool is generated from which an *orthogonal* sub-dictionary can be efficiently selected.

Transform design has been studied since the 1960s and the elements of dictionary design, their mathematical definitions and properties have been well established. Signal transforms that can be adapted into dictionaries have been broadly divided into two categories: (i) Linear & non-linear transforms, e.g. DCT, Gabor and Haar; and, (ii) High-dimensional analytic dictionaries designed using multi-resolution directional filter banks, e.g. contourlets and curvelets (Algorithm 7).

Given a basis,  $\{d_k\}_{k=0}^K \in \mathbb{R}^P$ , of a  $K$  dimensional subspace, a signal  $y \in \mathbb{R}^P$  can be linearly approximated by projecting onto the subset of  $K$  basis elements. This can be written as

$$y \approx \sum_{k \in I_k} (d_k^T y) d_k \quad (7.2.0.1)$$

However, in the non-linear case, each signal is allowed to use a different set of dictionary atoms in order to achieve best approximation. Equation 7.2.0.1 can be re-written as

$$y \approx \sum_{k \in I_k(y)} \mathbf{D}_k x_k^T, \quad (7.2.0.2)$$

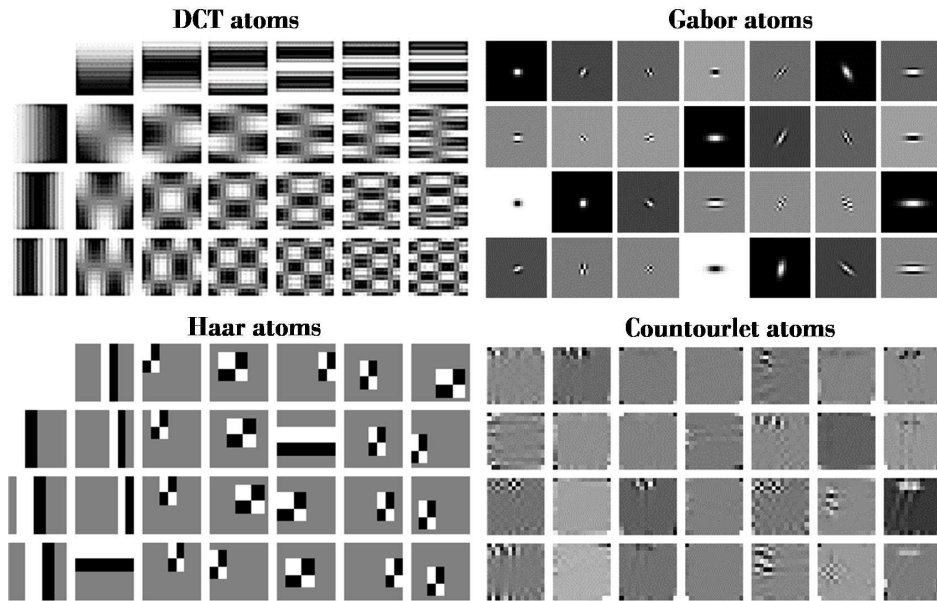
where,  $I_k(y)$ , is an index set adapted to each signal,  $y$ .

### DCT, Gabor and Haar Transforms

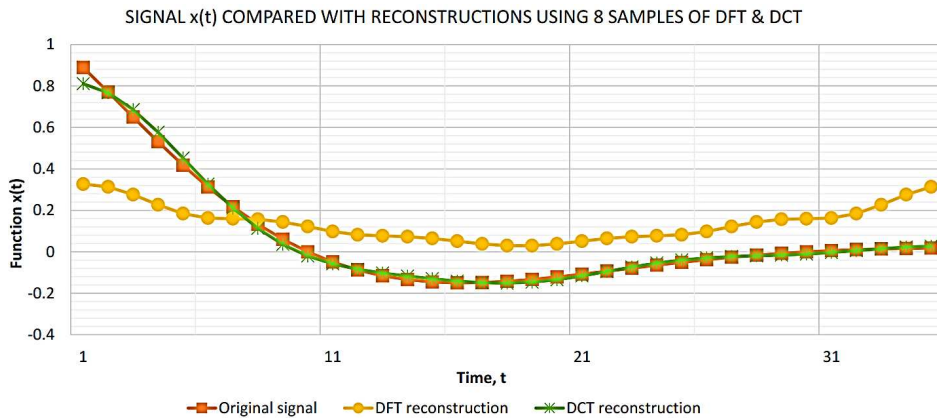
The most common of all linear transforms is the Fourier Transform [29]. The Fourier basis describes a signal in terms of its frequency content, a combination of orthogonal waveforms. It is also a method of expressing a periodic function in terms of several basis functions. Although, such a basis is an ideal choice for representing smooth and periodic functions, discontinuities cannot be well approximated due to the lack of localisation properties of Fourier basis functions.

**Discrete Cosine Transforms** In the discrete domain, the Fourier Transform have been extended in the complex and real domain as the discrete Fourier transform (DFT) [31] and the discrete cosine transform (DCT) [5], respectively. Figure 7.2a illustrates a subset of example DCT atoms of size  $14 \times 14$  used in the experiments. Due to the use





(a) DCT, Gabor, Haar and Contourlet atoms used in the experiments.



(b) Signal reconstruction using 8 regularly sampled coefficients of DFT and DCT.

Figure 7.2: a. Some of the dictionary atoms used in the experiments. The order of these atoms is optimised for every input signal. b. An illustration showing that the DCT approximation is more accurate using the same number of coefficients when compared with DFT.

of real numbers and the small number of basis vectors required to approximate an input signal, the DCT transforms have been extremely popular with the image processing and compression communities [112]. As illustrated in Figure 7.2b, for the same number of samples, the DCT approximates the signal with a higher accuracy. Unlike the Fourier Transform, the DCT does not assume a periodic extension of the signal, resulting in

smooth boundaries, a better approximation. A two-dimensional DCT can be written as:

$$F(u, v) = \frac{2}{\sqrt{NM}} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} \cos \left[ \frac{\pi u}{2N} (2i + 1) \right] \cos \left[ \frac{\pi v}{2M} (2j + 1) \right] \quad (7.2.0.3)$$

**Gabor Transform** A natural extension of the Fourier Transform is the Short Time Fourier Transform (STFT) [7], which provides better localisation and limits the effect of irregularities. The STFT was later generalised by Janssen et al. [94] which further developed into the Gabor Transform. The Gabor transform in the 2-dimensional case takes the form:

$$\mathcal{G}(\tilde{x}, \tilde{y}) = \exp \left[ -\frac{1}{2} \left[ \frac{\tilde{x}}{\sigma_x} \right]^2 + \left[ \frac{\tilde{y}}{\sigma_y} \right]^2 \right] \cos(2\pi f \tilde{x}),$$

$$\tilde{x} = x \cos \theta + y \sin \theta$$

$$\tilde{y} = y \cos \theta - x \sin \theta, \quad (7.2.0.4)$$

where  $\sigma_x$  and  $\sigma_y$  are the standard deviation of the Gaussian envelope in 2D. Using (7.2.0.4) a filter bank is created, where  $f$  controls the frequency resolutions of the transform and  $\theta$  controls the directionality, i.e. orientation. This is highly beneficial to represent spectral sonar signatures, illustrated in Figure 7.4. Figure 7.2a illustrates some examples of 2D Gabor atoms of different sizes and orientations.

**Haar Transform** The Haar transform has been extremely popular for image compression and sonar imagery is the Haar transform. Due to its discontinuous nature it can detect discontinuities and sudden transitions in signals. In [154], the authors have used this extensively in identifying sudden intensity changes between targets and their shadows

on the sea-bed. A Haar function takes the form

$$\Phi(y) = \begin{cases} 1 & 0 \leq x < 1/2 \\ -1 & 1/2 < x \leq 1, \text{ or more generally,} \\ 0, & \text{otherwise} \end{cases}$$

$$\Phi_{j,k}(y) = \Phi(2^j y - k), \quad (7.2.0.5)$$

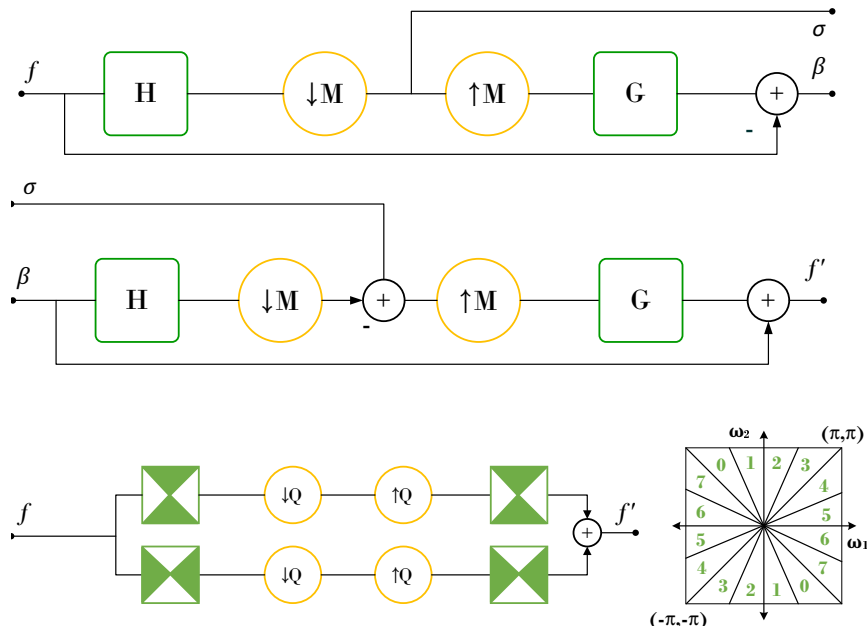
where  $j$  is non-negative integer and  $0 \leq k \leq 2^j - 1$ . Equation (7.2.0.5) is orthogonal in  $[0, 1]$ , with  $\int_0^1 \Phi_{j,k}(y) \Phi_{l,m}(y) dx = 0, \forall (i, j) \neq (l, m)$ . Examples of Haar like wavelet filters for different scales are shown in Figure 7.2a.

### Pyramidal Directional Filter Banks (PDFB) - Contourlets

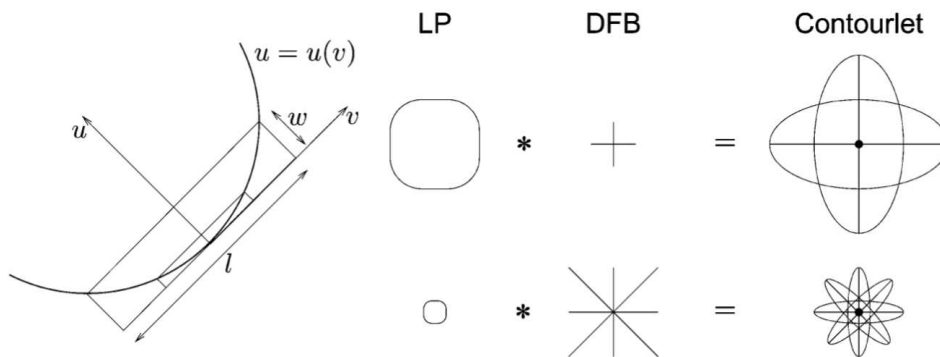
Wavelets, e.g. Gabor and Haar transforms, are efficient in capturing piecewise smooth signals and are well adapted to point singularities, especially in one dimension. This is shown in Section 7.4.2 where Haar and Gabor seem to provide good approximation and discrimination of targets which have dominant point singularities. However, in two-dimensions, interesting phenomena can be arranged along lines, curves, hyperplanes and other non-point like structures, for which wavelets are not well adapted. Wavelets may be less effective for images where the singularities dominate both scale or space, e.g. shape and directionality.

The intrinsic geometric or shape features of underwater mines in the two dimensional case require transforms that can capture not only directional details but also anisotropic and smooth curves. With regards to this, Ridgelets [37] and Contourlets were proposed [59] which are adapted to *straight* and *curved* singularities, respectively. In this work, contourlets were chosen to capture curve singularities and two dimensional Gabor and Haar transforms for straight singularities.

The pyramidal directional filter banks (PDFB) or Contourlets are implemented by decoupling the multi-scale and the directional decompositions using Laplacian Pyramid (LP) [35] and directional filter banks (DFB) [16], respectively. Such transforms are extremely useful in capturing high curve content in images, e.g. spherical or hemi-spherical shaped



(a) (Top) One-level LP decomposition and reconstruction using low-pass analysis,  $H$  and synthesis,  $G$ , filters, respectively.  $M$  is the sampling matrix,  $\sigma$  is the coarse approximation,  $\beta$  is the prediction error between  $f$  and  $f'$ . (Bottom) 2D spectrum partition using directional filter banks and a three level frequency partition, wedge shaped frequency bands; vertical directions: 0-3 and horizontal directions: 4-7.



(b) Curves' parabolic scaling relation which can be effectively approximated using contourlets. (Top) Parabolic scaling relation of curve fitting. (Bottom) Different support sizes with scale and orientation filters using a Laplacian Pyramid and directional filter banks (DFB), respectively.

Figure 7.3: Construction of the PDFB or contourlet transforms.

mines have high curvatures on edges. The basic idea behind LP is the following. A coarse approximation of the original signal is derived using a low-pass filter and then down-sampled. The approximation is then up-sampled to predict the original signal. Finally, a prediction error is derived by taking the difference between the approximated signal and the original signal. This is illustrated in Figure 7.3a. In order to solve the curve

singularity problem a set of DFB is used. The DFB is made up of two building blocks, a two-channel quincunx filter bank (QFB) [157],  $Q$ , and a shearing operator that handles pixel reordering. Finally, a three-level wedge-shaped frequency partition of contourlets is achieved by equally dividing the frequency spectrum into horizontal and vertical bands, as shown in Figure 7.3a.

Figure 7.3b illustrates the parabolic scaling of a curve and how LP and DFB combine to form the contourlet transform. The parametric representation of the discontinuity curve obeys

$$u(v) \approx \frac{kv^2}{2}, v \approx 0, \quad (7.2.0.6)$$

where  $k$  is the local curvature of the curve. The support sizes of the contourlet functions change according to the parabolic scaling. In order to fit a discontinuous curve at fine scales the width  $w$  and the length  $l$  of the basis function have to satisfy

$$w \approx \frac{kl^2}{8}, \quad (7.2.0.7)$$

Combining scale, the LP and direction, the DFB results in a contourlet function. The LP scheme allows capture of point discontinuities and the DFB's are used to solve curve-singularities which are highly effective in capturing curvature information in a 2D setting [59].

### Stage II - Dictionary Optimisation

Algorithm 2 can re-order, select a subset of dictionary atoms from a large over-complete dictionary and generates its corresponding coefficient vectors. These vectors are then used as feature vectors for object classification. The ordering and the coefficients of the dictionary atoms differ for each object class so a supervised dictionary learning approach is formulated using training samples and a cross-validation test is conducted.

**Stage III - Classification** Step II generates a large set of dictionary atoms based on a priori knowledge of the targets. Merging and re-ordering the atoms with respect to maximum discrimination can be achieved using Algorithm 2. Unlike previous works, a

single dictionary is learnt for the entire labelled training set. Using an optimal dictionary,  $\mathbf{D}_{opt}$ , a coefficient matrix  $\mathbf{X}_{test}$  is computed for all the test samples. The coefficient matrix  $\mathbf{X}_{test}$  is used as a feature matrix and a k-nearest neighbour (k-NN) classification is performed. Steps 5 - 7 compute a similarity matrix of  $\mathbf{X}_{test}$ , sorts the distances and finds a majority label for k-NN. The value for k was set to 3 for all the experiments. Finally, using the predicted labels,  $L_{predict}$  and the original labels,  $L_{org}$ , confusion matrices and misclassification error rates are computed.

### 7.3 | 1-D Sonar Echoes - Experiments & Results

Region-of-interest (ROI) selection based on automatic anomaly detection can be a highly useful tool when surveying large areas of sea-floor. In such scenarios, signal classification of traditional sonar returns is of huge importance. The analysis of backscattered sonar echoes can narrow down areas of interest, highlighting anomalous underwater regions by discriminating between seabed response and target echoes. Recently, Pailhas et al. [131] proposed to use bio-inspired adaptive chirps and presented echo discrimination of man-made materials from natural materials. Motivated by these results, as an on-going study, this work only presents approximation and discrimination results on 1-D sonar echoes. Whereas Pailhas et al. present results on sonar chirps, our approach is tested on three different sonar datasets, (i) Connectionist Sonar echo database, (ii) Synthetic side scan images and (iii) Real SAS images. The connectionist sonar set is a benchmark dataset

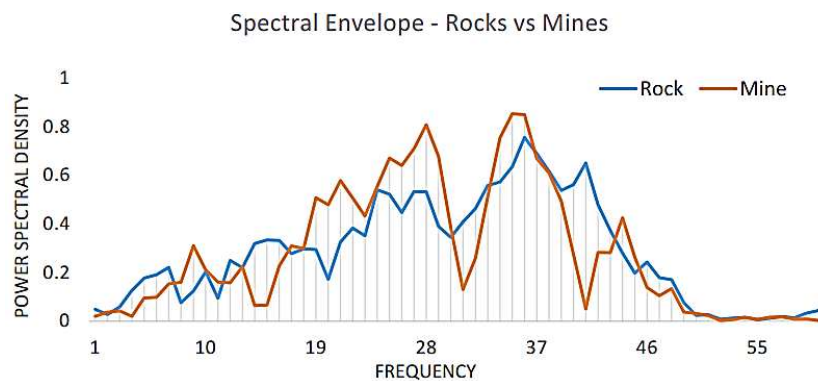


Figure 7.4: Spectral envelope of a mine and rock sonar return from the Connectionist Sonar Dataset.

Table 7.1: State-of-the-art and benchmark rate of misclassification results on the Connectionist dataset. Our approach performs best in both set of experiments.

Technique	Rate of Misclassification (%)	
	Angle Dependent	Angle Independent
Original Data + KNN	15.82	17.32
Neural Networks [74]	9.60	15.30
K-SVD + KNN [4]	16.75	18.75
$TS^3VM$ [117]	6.20	NA
Non-Linear PCA [75]	NA	17.71
<b>Our Approach</b>	<b>3.64</b>	<b>5.21</b>

from the UCI repository [74]. In total, there are 208 recordings, 111 backscattered sonar echoes from metal cylinders at various angles and 97 backscattered sonar echoes obtained from rocks. The backscattered signals are obtained from a variety of different aspect angles, spanning 180 degrees for different rocks and 90 degrees for metallic cylinders. The transmitted sonar pulse is a frequency-modulated up-chirp.

The targets were approximately 5ft in length and returns were collected at a range of 10m. Instead of using the raw data, authors in [74] extract a spectral envelope on the temporal signal by shifting a set of synthetic sampling aperture. This is carried out on the STFT spectrogram of the sonar return by integrating the area under each window. Figure 7.4 illustrates an example of a spectral envelope created for a single mine and rock sonar time series, respectively.

Working only on the spectral envelopes, the performance of the proposed algorithm is tested in terms of misclassification rate and reliability. State-of-the-art results and benchmarked results are also reported.

In the original work [74], the authors propose a neural network model to learn and classify these signals into two classes, rock and mine. They suggest two sets of experiments. In the “aspect-angle independent” experiment, the whole data set is used without controlling for aspect angle. In the “aspect-angle dependent” experiment the training and testing sets were carefully controlled to ensure that each set contained cases from different aspect angles in appropriate proportions. For both sets of experiments, the data is randomly

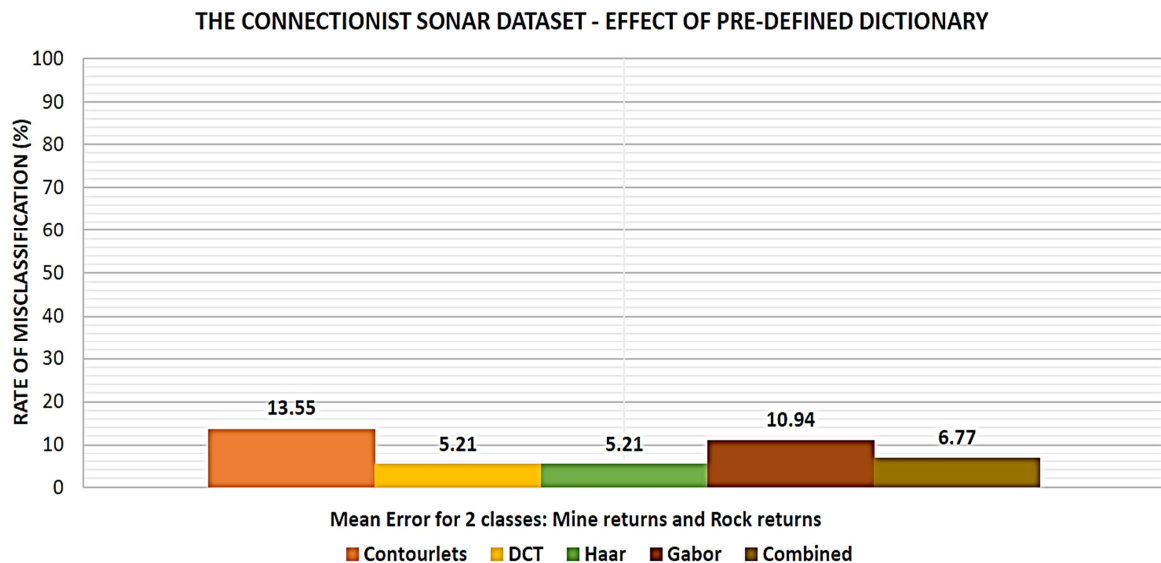


Figure 7.5: Effect of pre-defined dictionary on classification.

partitioned into 13 sets with 16 data points each. These sets were then used to create 12 training sets and 1 test set. Using a learnt neural network average classification rate of 84.7% and 90.4% was achieved for the aspect-angle independent and aspect-angle dependent experiment, respectively.

In [75], the authors propose a method, derived from the generative topographic mapping (GTM) [22], a non-linear generalisation of principal component analysis (PCA). The GTM method explicitly constructs a non-linear mapping from the  $L$ -dimensional latent space  $[0, 1]^L$  into a  $D$ -dimensional data space  $\mathbb{R}^D$ . They take a different approach to avoid over-fitting and carry out 30 test runs with an average classification accuracy of 71.2%. By careful parameter selection they report a classification accuracy of 82.3%. Classification results using a simple  $k$ -nearest neighbour (K-NN) algorithm, treating each spectral signature as a feature vector, are also reported in Table 7.1.

Recently, Malchiodi et. al [117] proposed a modified support vector (SV) based approach to a learning algorithm for partially labelled data. They report state-of-the-art results on the aspect-angle-dependent experiment with classification accuracy of 93.8%. Table 7.1 also illustrates classification results using a dictionary learning algorithm [4], where dictionary atoms are labelled exemplars chosen from the training dataset.



In contrast to other works, our approach achieves accuracies of 96.36% and 94.79% on the *angle dependent* and the *angle independent* datasets, respectively. The dictionary selection and merging technique proposed in this paper outperforms the current state-of-the-art results. This is evident in Table 7.1, especially for *angle independent* experiment where reduction in the error rate is 59% compared to the previous best approach [74].

The above experiments were repeated by fixing individual dictionary categories. The effect of the pre-defined dictionary on the misclassification error is evident in Figure 7.5. The misclassification error reported here is a mean error for 2 different classes: sonar returns from mines and rock, respectively. Due to discontinuities in the data it can be seen that the DCT and Haar dictionary bases perform well individually when compared to the contourlets and the Gabor bases. A combination of all the bases does not generate the sparsest solution and may not be the best strategy in this case. Signals with dominant point singularity features can be well approximated with the DCT compared to computationally expensive Contourlets or PDFB filters. Although this seems to be true for 1-D sonar signals, we show in Section 7.4 that this is not always true, especially for 2-D images.

## 7.4 | Sonar Images - Experiments & Results

New SAS sensors offer a higher resolution which can match classical optical images. For example, the CMRE, MUSCLE SAS, system can achieve up to 2cm pixel resolution. Thanks to this resolution, direct analysis on the highlights of the target rather than on its shadow is possible. In this work the experiments were carried out on both synthetic and real SAS imagery.

### 7.4.1 | Simulated Side-scan Imagery

Before working on the real dataset, a simulated side-scan image set was created using an in-house simulator previously reported in [132]. The dataset is divided into two sets,

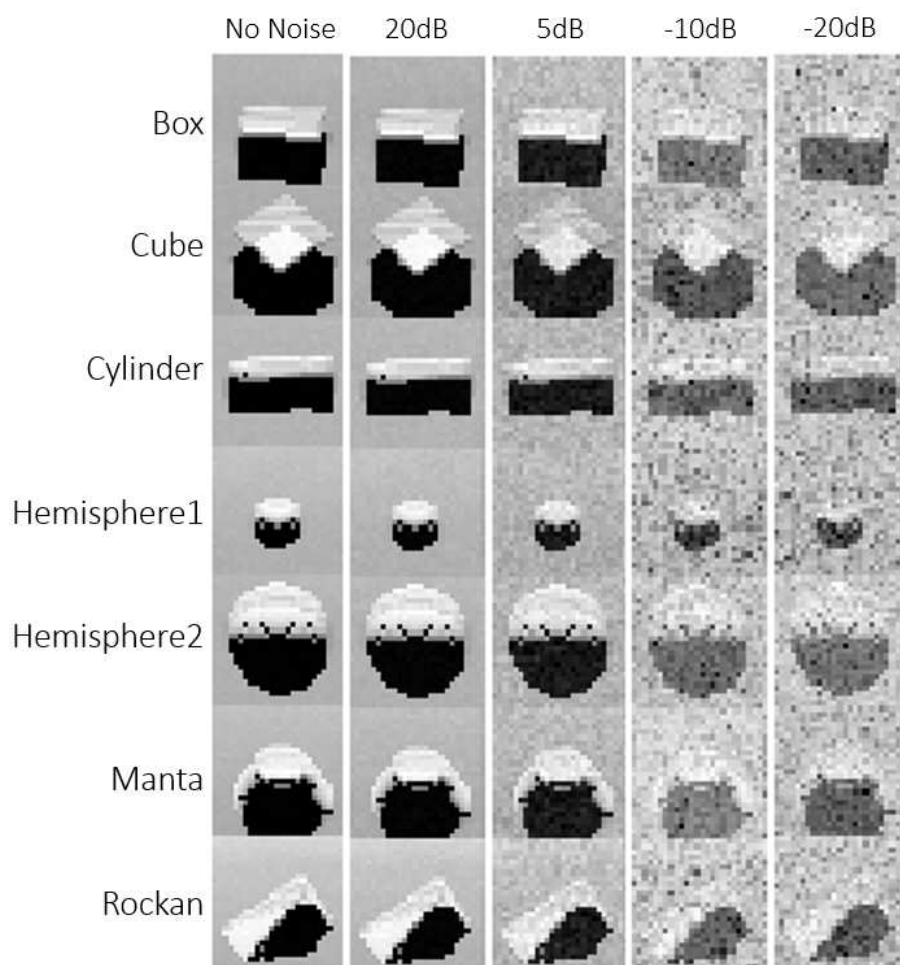


Figure 7.6: Example side-scan images from *Dataset02*, created using a real-time side-scan simulator [132]. Images shown here have a  $5\text{cm}$  pixel resolution and are on a plain featureless sea-bed.

Table 7.2: *Dataset01* confusion matrix, using Eigenfaces [132]. Image pixel resolution was set to 5cm and no noise was added.

	Cylinder	Cube	Manta	Rockan
Cylinder	<b>0.7272</b>	0.2605	0	0.0123
Cube	0.2099	<b>0.7716</b>	0	0.0185
Manta	0	0	<b>0.9815</b>	0.0185
Rockan	0.0259	0.0432	0	<b>0.9309</b>

Table 7.3: *Dataset01* confusion matrix, using our approach. Image pixel resolution was set to 5cm and no noise was added.

	Cylinder	Cube	Manta	Rockan
Cylinder	<b>0.9185</b>	0.0815	0	0
Cube	0.0790	<b>0.9185</b>	0	0.0025
Manta	0	0	<b>1.0000</b>	0
Rockan	0.0543	0.0840	0.0062	<b>0.8556</b>

*Dataset01* and *Dataset02*. The *Dataset01* dataset is made up of 4 targets: (i) cube, (ii) cylinder, (iii) manta, and (iv) rockan. *Dataset02* is made up of 7 different mine like objects: (i) box (ii) cube, (iii) cylinder, (iv) hemisphere type 1, (v) hemisphere type 2, (vi) manta, and (vii) rockan sensed under different sea-bed conditions. *Dataset01* is a subset selected from *Dataset02*. Another motivation behind creating a synthetic dataset is the lack of benchmarked labelled side-scan data, making it difficult to compare different classification techniques.

**Experiments on Dataset01:** Unlike *Dataset02*, shown in Figure 7.6, *Dataset01* is a difficult set to classify due to visual similarities between the cylinder and cube target images. In [132], an Eigenfaces approach, motivated by the PCA technique for dimensionality reduction was used to learn appearance based features and tested on the same datasets, *Dataset01* and *Dataset02*. Our approach demonstrates superior classification accuracy when compared to the Eigenfaces approach [132].

The confusion matrix of *Dataset01*, when classified using our proposed technique is listed in Table 7.3. In order to compare against the technique reported in [132], these experiments were carried out at 5cm pixel resolution and no noise was added. Under such

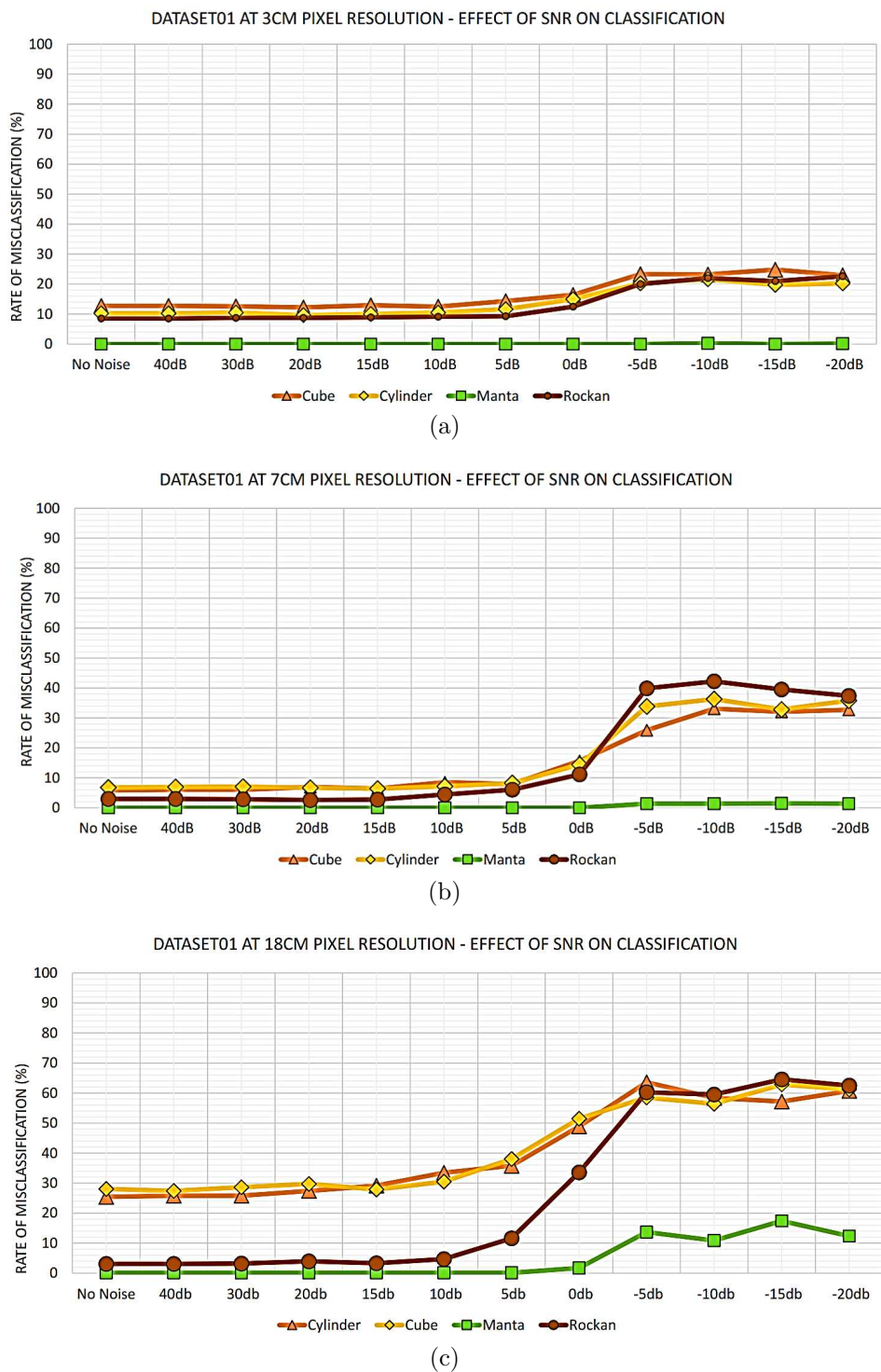
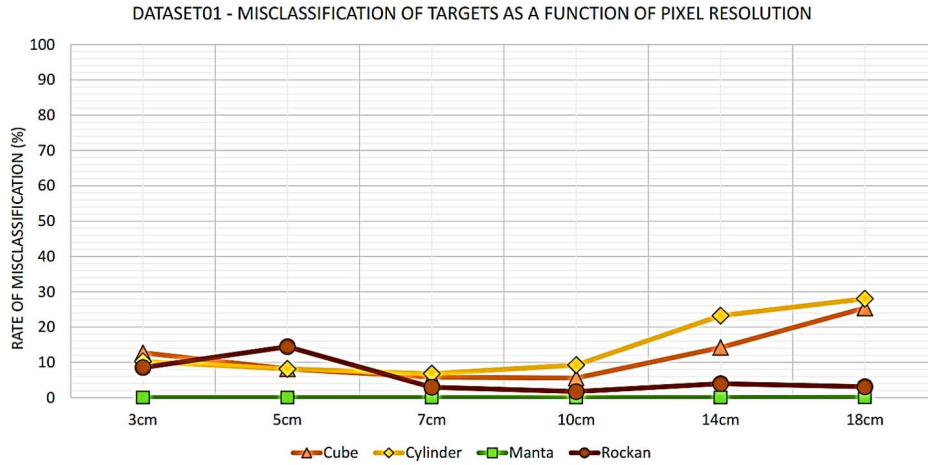
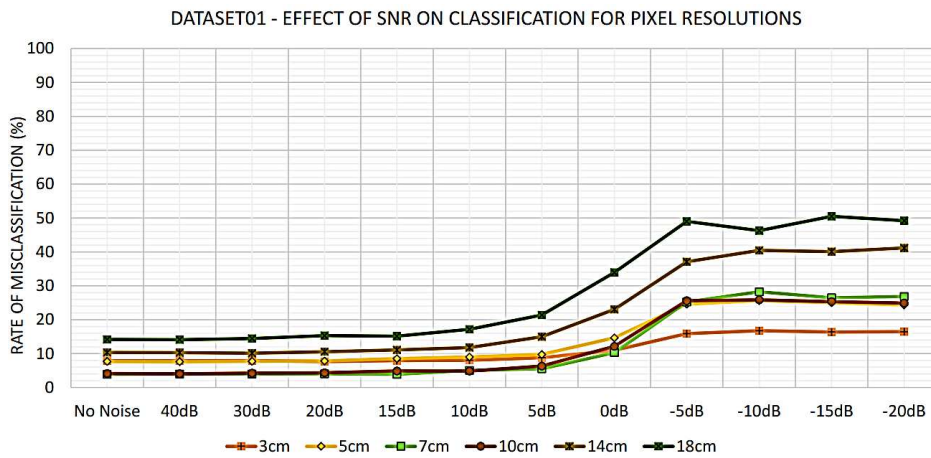


Figure 7.7: Misclassification rate for *Dataset01* illustrated as a function of SNR: (a) at 3cm, (b) at 7cm and (c) at 18cm pixel resolution.



(a) The proposed approach gives lowest misclassification errors at 7cm and 10cm pixel resolutions.



(b) With pixel resolution as low as 10cm pixel resolution, the rate of misclassification stays below 25% at -5dB SNR.

Figure 7.8: The effect of pixel resolution on misclassification error for *Dataset01* using the proposed approach. (a) Misclassification error rate for individual targets illustrated as a function of pixel resolution. (b) Mean misclassification rate of different target at different pixel resolutions is illustrated as a function of Rayleigh noise.

Table 7.4: *Dataset02* confusion matrix, using the proposed approach. Image pixel resolution was set to  $5cm$  and no noise was added.

	<b>Box</b>	<b>Cube</b>	<b>Cylinder</b>	<b>Hem'1</b>	<b>Hem'2</b>	<b>Manta</b>	<b>Rockan</b>
<b>Box</b>	<b>0.9840</b>	0.0012	0.0099	0.0025	0	0	0.0025
<b>Cube</b>	0.0136	<b>0.9840</b>	0.0012	0	0.0012	0	0
<b>Cylinder</b>	0.0049	0	<b>0.9901</b>	0.0025	0	0	0.0025
<b>Hem'1</b>	0	0	0	<b>1.00</b>	0	0	0
<b>Hem'2</b>	0	0	0	0.0012	<b>0.99</b>	0	0
<b>Manta</b>	0	0	0	0	0	<b>1.00</b>	0
<b>Rockan</b>	0.0284	0.0025	0.0457	0.0235	0	0	<b>0.90</b>

conditions, it can be seen that the proposed technique outperforms the Eigenfaces technique on all except the Rockan target class.

To prove the robustness of the proposed approach experiments under deteriorating pixel resolution and SNR were also carried out. Figure 7.7 illustrates misclassification curves as a function of Rayleigh noise for four different MLOs. Under the influence of noise the proposed technique produces low misclassification rates compared to previously reported results in [132]. Even at  $18cm$  pixel resolution, the mean misclassification stays below 30%. This is evident in Figure 7.8a. Figure 7.8b shows improved classification accuracy when the signal resolution is degraded and the noise levels are increased. It can be seen that with even with a  $10cm$  pixel resolution the rate of misclassification error stays below 25% even at  $-5dB$  SNR.

**Experiments on Dataset02:** The misclassification error curves as a function of pixel resolution and Rayleigh noise are presented in Figure 7.9. Under the influence of noise and the resolution as low as  $18cm$  our method produces low misclassification rates. This is evident in Figure 7.10 as the misclassification rate stays below 10% for different pixel resolutions. The confusion matrix for *Dataset02* was generated at  $5cm$  pixel resolution and no noise was added. Only *Cube*, *Cylinder*, *Manta* and *Rockan* classes were chosen for the Eigenfaces experiment as reported in [132], Table 7.5. The confusion matrix under similar experimental conditions using the proposed approach is presented in Table 7.4. Improved classification rates can be seen for the *Cube* and *Manta* class. To prove the

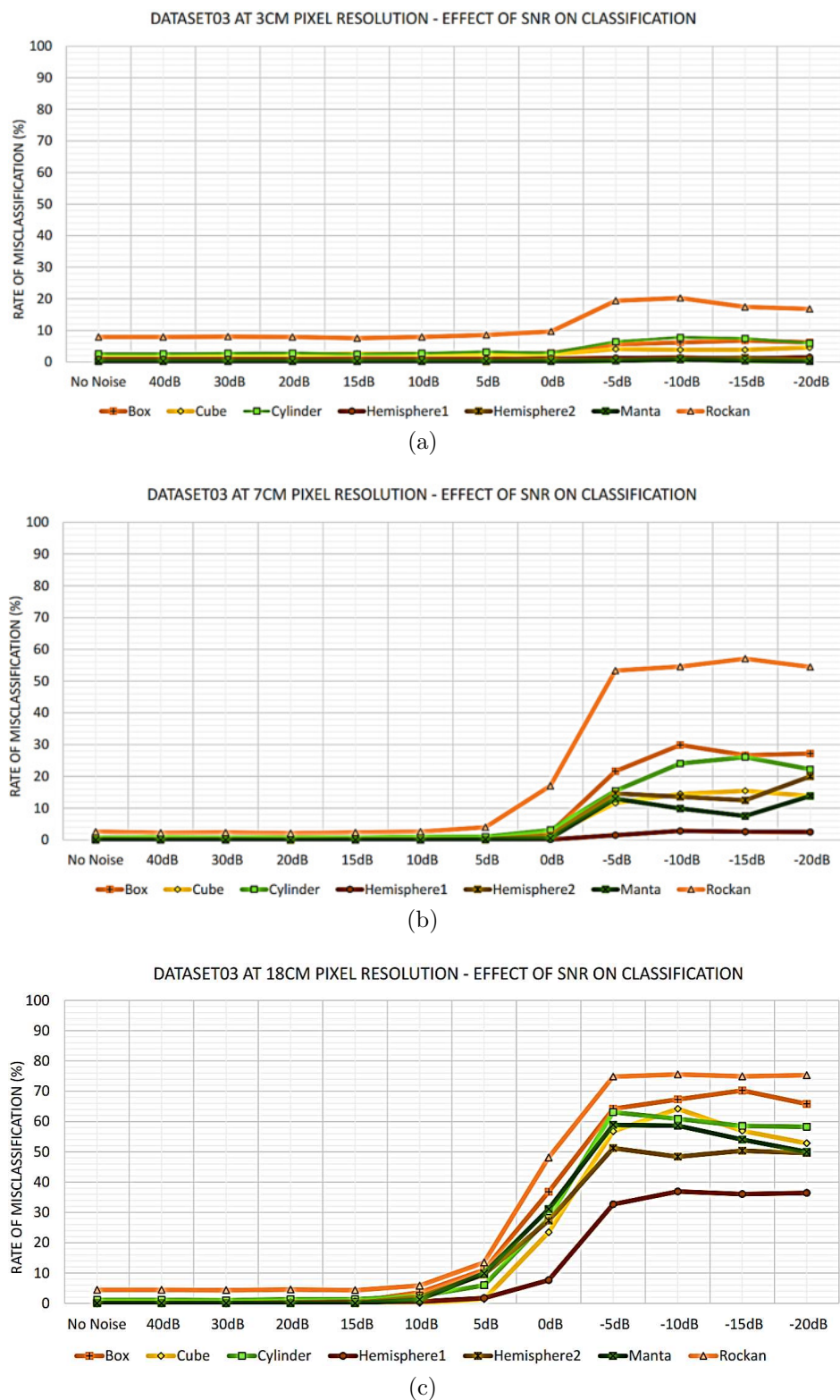


Figure 7.9: Misclassification rate for *Dataset02* illustrated as a function of SNR at (a) at 3cm, (b) at 7cm and (c) at 18cm pixel resolution.

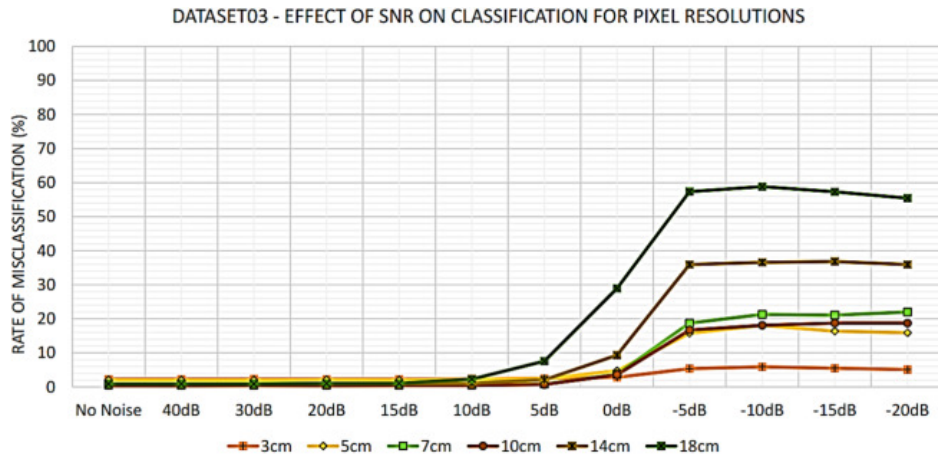


Figure 7.10: Misclassification rate for different pixel resolutions of *Dataset02* illustrated as a function of Rayleigh noise. Even at 10cm pixel resolution, the rate of misclassification stays below 22% for  $-20\text{dB}$  SNR.

Table 7.5: *Dataset02* confusion matrix using Eigenfaces [132]. Image pixel resolution was set to 5cm and no noise was added.

	Cylinder	Cube	Manta	Rockan
Cylinder	<b>0.9938</b>	0	0.0025	0.0037
Cube	0	<b>0.9210</b>	0	0.0790
Manta	0.0025	0	<b>0.9963</b>	0.0037
Rockan	0.0025	0.0309	0.0012	<b>0.9654</b>

robustness of the proposed approach experiments under deteriorating pixel resolution and signal SNR were also carried out for *Dataset02*. Figure 7.9 illustrates the misclassification curves as a function of Rayleigh noise for seven different MLOs. Figure 7.10 highlights the robustness of the proposed technique, especially when images are under the influence

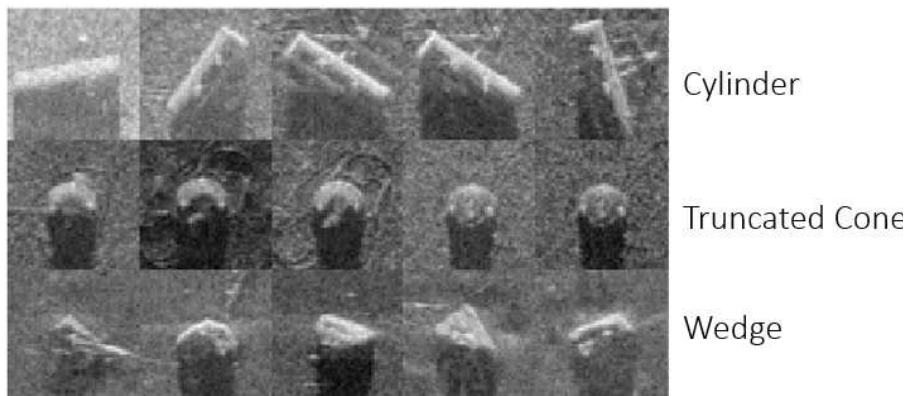


Figure 7.11: Real synthetic aperture sonar exemplar images of mine like objects from the MUSCLE SAS dataset.



Table 7.6: Confusion matrix, using our approach on the MUSCLE SAS image dataset.

	Cylinder	Truncated Cone	Wedge
Cylinder	<b>0.9402</b>	0.0513	0.0085
Truncated Cone	0	<b>0.9829</b>	0.0171
Wedge	0	0.1026	<b>0.8974</b>

of low pixel resolution and low SNR. At 10cm pixel resolution, the rate of misclassification stays below 22% when the SNR is as low as  $-20$ dB.

### 7.4.2 | Real SAS Imagery

Experiments on real synthetic aperture sonar images were also carried out. The SAS datasets used in this paper were gathered by NATO during the COLOSSUS 2 trials using the MUSCLE SAS system. A description can be found in [77]. Work by Blair et al. [23] focussed on detection and classification of mines on full-size images captured using MUSCLE SAS, unlike [132], where a neatly cropped region-of-interest (ROI) is centred on the targets and used for training and testing. Authors in [23] compare linear, non-linear support vector machines (SVM) and Gaussian process (GP) classifiers reporting classification accuracies with uncertainties. The focus of our approach is on classification and to learn an optimal and discriminatory dictionaries for different MLOs using a very small subset of this dataset with hand labelled exemplars. Figure 7.11 shows some example images of mine like objects imaged using the MUSCLE SAS sensor on the sea-bed. The dataset is divided into three classes: (i) Cylinder, (ii) Truncated Cone, and (iii) Wedge. Similar to the synthesised side-scan sonar data, the labelled dataset is divided into training and testing sets and a 10-fold cross-validation experiment is performed. The confusion matrix from these experiments is shown in Table 7.6. Noticeably, our approach correctly labels all the entries in the *cylinder* and *truncated cone*.

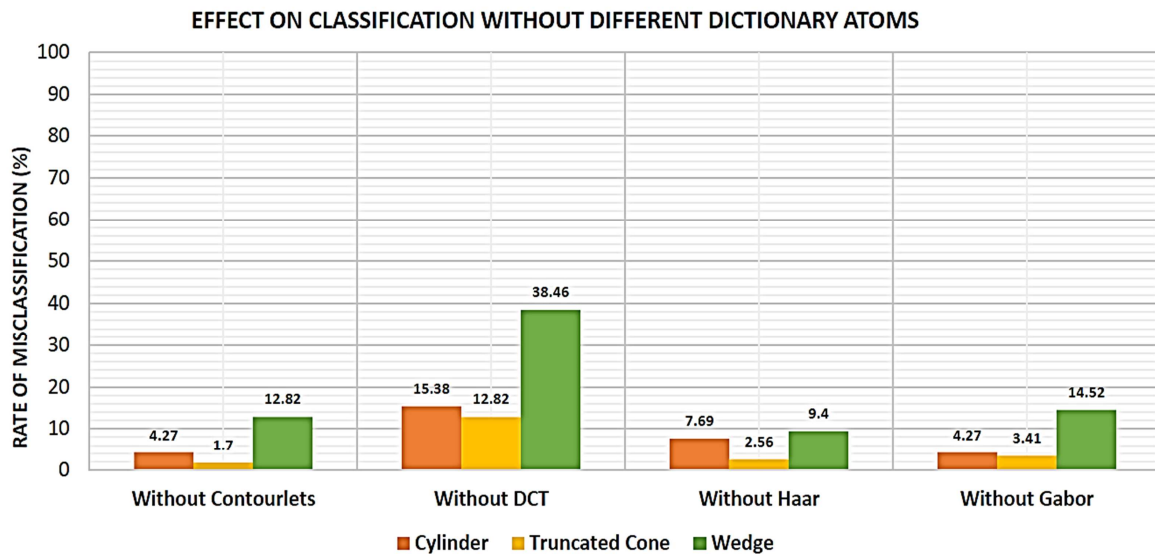


Figure 7.12: Impact of different dictionary categories on rate of misclassification for different MLOs.

## 7.5 | Summary

### 7.5.1 | What is an optimal dictionary?

Recent work in the field of sparsity based classification argue that learning an optimal dictionary that is adapted to the input data, for example using the K-SVD algorithm [4] produces state-of-the-art classification results. This work shows that it may not be true always, especially when the training data is not easily available or labelled. Experiments in this work show that relying on orthogonal dictionaries learnt from the training set is inefficient in classification tasks. This can be seen in Table 7.1, where the proposed approach in this chapter is also compared to a dictionary learning approach which uses input data as training samples [4].

In order to study the impact each set of orthogonal basis has on the classification task an experiment was conducted on the CMRE MUSCLE SAS dataset. Similar to experiments in section 7.4, for each experiment, the SAD optimisation is performed excluding each of the basis set, e.g. DCT, Haar, Gabor and Contourlets. Excluding 2-D DCT coefficients

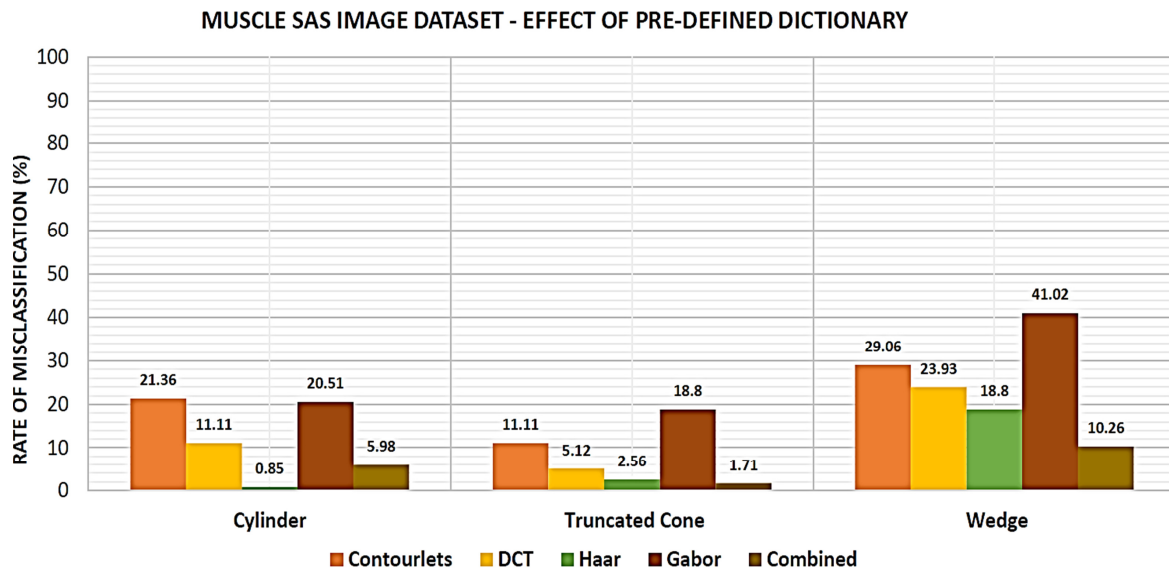


Figure 7.13: Effect of pre-defined dictionary on target classification.

results in higher misclassification error for all three MLOs. This can be seen in Figure 7.12. Leaving out Contourlet coefficients results in higher misclassification error rates for wedge shaped MLOs which have dominant curved singularities. This is evident in Figure 7.13 where Haar transforms provide better approximation and discrimination for cylindrical MLOs with dominant linear point singularities; and, Contourlets improve classification accuracies when compared directly to Gabor transforms for truncated-cone MLOs with dominant curve singularities.

Figure 7.13 illustrates classification results for three different MLO classes. The experiment where a combined dictionary is selected before the *SAD* optimisation is applied, see Section 3.1.1, is illustrated as *Combined* in Figure 7.13. In contrast to Figure 7.5, a combined pool of different pre-defined bases produces the least misclassification errors. Least misclassification errors is also reported on *Dataset01* and *Dataset02*, Figure 7.9 and Figure 7.7, respectively.

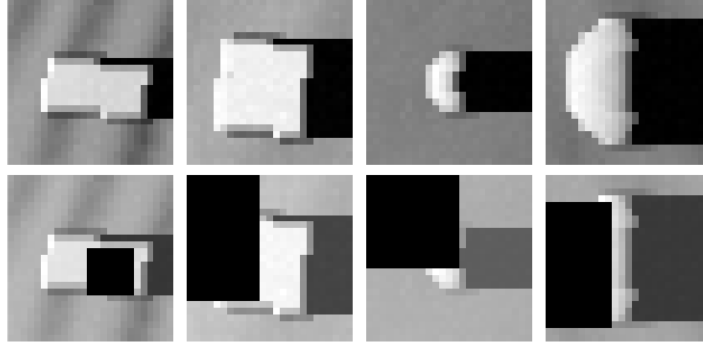


Figure 7.14: (Top) Example images of 4 different targets. (Bottom) Targets under 5%, 16%, 25% and 50% occlusion.

### 7.5.2 | Importance of SNR & Pixel Resolution

Signal to noise ratio (SNR) in SAS images limits the efficiency of detection and classification algorithms. In this work, the effect of noise on the classification process is also tested. Sonar noise is modelled by a coherent Rayleigh noise for the synthetic datasets, *Dataset02* and *Dataset01*. Both datasets are tested for different SNR values, high SNR, 40dB to low SNR, -20dB. Snapshots of different targets under noise are shown in Figure 7.6. The proposed algorithm have been run for different SNR values, 40dB to -20dB and the results show that the combined mean percentage of misclassification for all targets stays below 26% and below 11% for *Dataset01* and *Dataset02*, respectively.

Similar experiments were run for different pixel resolutions, 3cm to 18cm. Figure 7.8a illustrate misclassification errors on simulated SAS datasets. The minimum percentage of misclassification error is achieved at 7cm pixel resolution and 10cm for *Dataset01* and *Dataset02*, respectively. Experiments also suggest that it is not always true that adapting dictionary atoms to input data leads to least classification errors. Section 7.3 supports this argument with state-of-the-art results on the connectionist sonar dataset.

### 7.5.3 | Effect of occlusion on classification accuracy?

Occlusions are not always due to obscuring objects but may also be due to faulty sensors or a soiled field of view due to debris on the sensor lens, for example. Proposing classification and detection techniques and testing them against such occlusions are crucial for systems

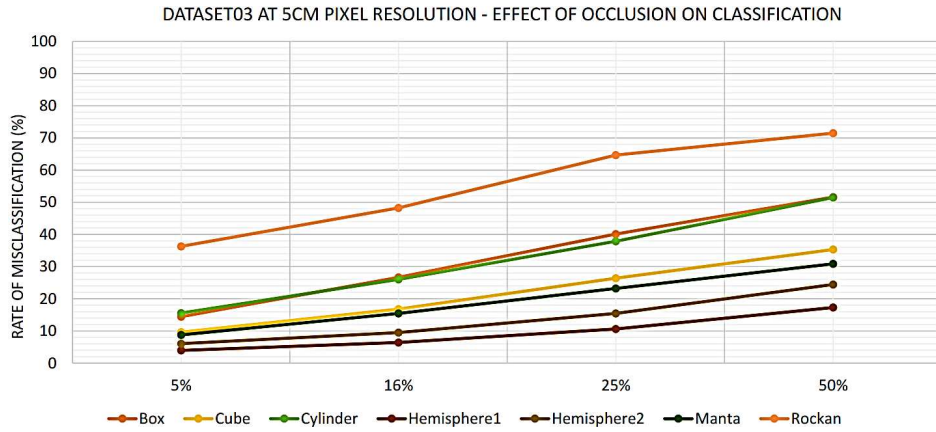


Figure 7.15: Rate of misclassification for 7 different targets achieved using the proposed approach on *Dataset03* by varying the percentage of occluded pixels.

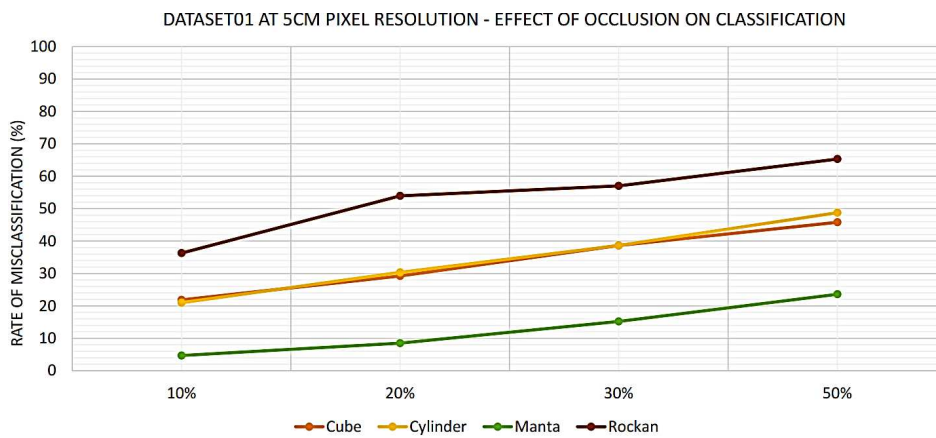


Figure 7.16: Rate of misclassification for 4 different targets achieved using the proposed approach on *Dataset01* by varying the percentage of occluded pixels.

engineering and decision making in underwater missions. Figure 7.14 illustrates target examples before and after occlusion. The amount of occlusion varies between 5% – 50%. Similar to previous experiments on simulated data, studying the effects of SNR and pixel resolution, *Dataset01* and *Dataset03* are chosen for occlusion experiments. The rate of misclassification is presented in Figure 7.15 and Figure 7.16 corresponds to *Dataset01* and *Dataset03*. It can be seen that the rate of misclassification for majority of the targets, except Rockan, box and cylinder, stays below 50%, doing better than random guess.

Future work will focus on two fronts: (i) signal classification of backscattered sonar returns using bio-inspired chirps, and, (ii) adapting and applying the proposed technique to raw full-waveform multi-spectral LiDAR data underwater, studying the effects of foliage and different materials.

# Chapter 8

## Conclusion

In this work, we present several signal processing pipelines with novel sparse approximation and dictionary learning algorithms for 1D, 2D and 3D datasets. We exploit the concept of sparsity in signal reconstruction using an over-complete and orthogonal set of basis or dictionaries, which can be fixed or learnt from the input data.

We show that intelligent encoding (Chapter 3.2) can be more practical and efficient in comparison to robust signal modelling techniques, especially when processing full-waveform LiDAR signals. The idea of compressed representation of time-series FW-LiDAR data (Chapter 4 and 6), geometric (shape) properties combined with spectral signatures (Chapter 6), can detect spectral anomalies (Chapter 4) and classify objects in LiDAR and sonar data (Chapter 7).

The non-linear dictionary learning algorithm (K-SAD Algorithm 3) addresses on two central problems: i) handling non-linearity in the input space by improving classification accuracy on existing publicly available datasets; ii) further reducing the learning and classification algorithm run-time through kernel matrix approximation. This work combines an orthogonal incoherent discriminatory dictionary learning method in the non-linear space with an efficient approximation of a kernel matrix.

We complement the small runtime required by the orthogonal DL step with a fast kernel approximation stage. We report state-of-the-art results on large-scale high-dimensional

datasets and report an average classification accuracy of 97% on 8-bit digits, face and RGB-D images. Unlike most sparsity based classifiers our approach uses the coefficients as target signatures.

Chapter 4 shows how the methodology can be applied to anomaly detection from aerial full-waveform LiDAR signals. We show that semantic relationships across layers of the tree and on the ground can be learnt using the sparse coefficients. The *concept of normality* in FW-LiDAR signals is represented using the *sparse discriminative subspace* algorithm. We learn sparse coefficients across multiple wavelengths and tree layers for various materials, e.g. bark, needle and metal. The anomaly detection approach is tested on simulated full-waveform data.

In chapter 5, we address the problem of simultaneously modelling and classifying FW-LiDAR signals. In our experiments, we model the instrumental response and its corresponding backscatter using a family of generalised Gaussian functions. Our proposed approach, SPeED (Chapter 4), is significantly different. We suppose that an individual peak is composed of elementary shape functions (mathematically modelled or learnt from the data) and its coefficient or contribution vector is a significant parameter in class separation. The elementary shapes can be fixed (e.g. generalised Gaussian) or can be learned from the data (Algorithm 5). The SPeED algorithm learns peak parameters, e.g. location, amplitude and shape, from the data. We show that these parameters can be used in unsupervised and supervised classification.

We compare our approach with the RJMCMC approach and report results using several metrics. The improvement on the average number of peaks extracted using the SPeED algorithm when compared against the RJMCMC approach [85] is 2.3 times and reducing the time complexity by 234 times. This makes our approach highly suitable in practice.

The proposed algorithms are further tested on underwater time correlated single photon counting data for mine countermeasures. A novel spectral-depth representation is presented that is highly discriminative in characterising different target signatures underwater. Several custom-made realistically scaled exemplars with known and unknown

targets have been investigated using a multi-spectral single photon counting LiDAR system. Multi-spectral measurements were made on targets with different shapes and materials. Our approach combines shape and spectral representations using sparse codes. These are optimised for maximum discrimination. Combining depth with spectral data, the approach is very efficient in classifying targets of different shapes. This work is the first to report the analysis and discrimination of multi-spectral underwater, single photon counting LiDAR signals as an alternative to acoustic MCM.

In Chapter 7, we have shown that a union of discriminatory dictionaries can be selected along with their respective coefficients for mine classification tasks (1D and 2D Sonar datasets). Our approach has been tested on traditional 1-D sonar returns with an accuracy of 96%, 2-D realistic side-scan sonar imagery with an accuracy of 95% and 2-D SAS imagery with an accuracy of 94%. With a 23.24% decrease in the mis-classification error rate on the CMRE MUSCLE SAS imagery, we have studied the impact of different orthogonal basis sets on target mis-classification rates. An optimal union of such dictionaries is then selected using an optimisation routine which maximises sparsity and dictionary discrimination.

Our approach mathematically models target signals and uses such models a-priori for dictionary design. We have also tried to answer the following questions: (i) Can a combined dictionary optimised for discriminating different MLO classes and natural rocks be found and learnt? (ii) What influence does noise and pixel resolution have on accuracy? We show, for mine counter-measures, when classifying 2D images, a combined pool of pre-defined dictionary atoms produces the least misclassification error. This was reported on both side scan and synthetic aperture sonar images. Similar experiments were run for different pixel resolutions, 3cm to 18cm. Our experiments shows a minimum mis-classification error is achieved at 7cm pixel resolution and 10cm for Dataset01 and Dataset02, respectively.



# Appendix A

## Proposition Proof

**Proof for proposition 1** We apply the majorisation-minimisation method to solve our non-linear cost function  $J(\mathbf{X})$  (3.2.1.1) using a surrogate function  $M(\mathbf{X}, \mathbf{X}_t) = \frac{1}{2}(\mathbf{X} - \mathbf{X}_t)^T (\alpha \mathbf{I} - \mathbf{D}^T \mathbf{D})(\mathbf{X} - \mathbf{X}_t)$ . Hence by design  $\hat{J}(\mathbf{X}) = J(\mathbf{X}) + Q(\mathbf{X}, \mathbf{X}_t)$  coincides with  $J(\mathbf{X})$  at  $\mathbf{X}_t$ . Solving the modified cost function leads to

$$\begin{aligned} \hat{J}(\mathbf{X}) = & \Phi_{train}^T \Phi_{train} - 2\Phi_{train}^T \mathbf{D}\mathbf{X} + \\ & \mathbf{X}^T \mathbf{D}^T \mathbf{D}\mathbf{X} + (\mathbf{X} - \mathbf{X}_t)^T (\alpha \mathbf{I} - \mathbf{D}^T \mathbf{D})(\mathbf{X} - \mathbf{X}_t) \end{aligned} \quad (\text{A.0.0.1})$$

$$\begin{aligned} \frac{\partial J(\mathbf{X})}{\partial \mathbf{X}} = & -2\mathbf{D}^T \Phi_{train} - 2(\alpha \mathbf{I} - \mathbf{D}^T \mathbf{D})\mathbf{X}_t + \\ 2\beta_1 \mathbf{X} = 0 \Rightarrow \mathbf{X}^* = & \text{T}_{\beta_1} \left( \mathbf{X}_t + \frac{1}{\alpha} \mathbf{D}^T \Phi_{train} - \right. \\ & \left. \mathbf{Z}\mathbf{X}_t + G(\mathbf{X}_t) \right), \frac{\beta_1}{2\alpha} \end{aligned} \quad (\text{A.0.0.2})$$

**Proof for proposition 2** The reduced rank procrustes rotation (Theorem 4 in [195]) shows that for the minimisation problem for  $\mathbf{D}$  in

$$\min_{\mathbf{D}} \|\Phi_{train} - \mathbf{D}\mathbf{X}^T\|^2 \quad (\text{A.0.0.3})$$

subject to  $\mathbf{D}^T \mathbf{D} = \mathbf{I}$ , has a unique solution  $\mathbf{D}^* = \mathbf{U}\mathbf{V}^T$ , where  $\Phi_{train} \mathbf{X}^T = \mathbf{U}\Sigma\mathbf{V}^T$ . In

contrast to this work, where the dictionary learnt is a kernel dictionary, the proof in [195] is in the linear domain.

# Bibliography

- [1] N. Aeronautics and H. R. C. Space Administration (NASA): HQ. *Apollo 15 Press Kit*. Bibliogov, 2012.
- [2] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(11):1475–1490, 2004.
- [3] M. Agyemang, K. Barker, and R. Alhajj. A comprehensive survey of numeric and symbolic outlier mining techniques. *Intelligent Data Analysis*, 10(6):521–538, 2006.
- [4] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing over-complete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322, 2006.
- [5] N. Ahmed, T. Natarajan, and K. R. Rao. Discrete cosine transform. *Computers, IEEE Transactions on*, 100(1):90–93, 1974.
- [6] M. Alexa and A. Adamson. On normals and projection operators for surfaces defined by point sets. In *Proceedings of the First Eurographics conference on Point-Based Graphics*, pages 149–155. Eurographics Association, 2004.
- [7] J. B. Allen and L. R. Rabiner. A unified approach to short-time fourier analysis and synthesis. *Proceedings of the IEEE*, 65(11):1558–1564, 1977.
- [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 Transactions on Computational Imaging*, 2(4):456–467, 2016.

- [9] M.-C. Amann, T. Bosch, R. Myllyla, M. Rioux, and M. Lescure. Laser ranging: a critical review of usual techniques for distance measurement. *Optical Engineering*, 40(1):10–19, 2001.
- [10] W. Armbruster. Bayesian hypothesis generation and verification. *Pattern Recognition and Image Analysis*, 18(2):269–274, 2008.
- [11] W. Armbruster. Exploiting range imagery: techniques and applications. In *International Symposium on Photoelectronic Detection and Imaging 2009*, pages 738203–738203. International Society for Optics and Photonics, 2009.
- [12] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.
- [13] M. Attene, S. Marini, M. Spagnuolo, and B. Falcidieno. Part-in-whole 3d shape matching and docking. *The Visual Computer*, 27(11):991–1004, 2011.
- [14] S. D. Babacan, R. Molina, and A. K. Katsaggelos. Bayesian compressive sensing using laplace priors. *IEEE Transactions on Image Processing*, 19(1):53–63, 2010.
- [15] A. Baldrige, S. Hook, C. Grove, and G. Rivera. The aster spectral library version 2.0. *Remote Sensing of Environment*, 113(4):711–715, 2009.
- [16] R. H. Bamberger and M. J. Smith. A filter bank for the directional decomposition of images: theory and design. *Signal Processing, IEEE Transactions on*, 40(4):882–893, 1992.
- [17] D. Baron, S. Sarvotham, and R. G. Baraniuk. Bayesian compressive sensing via belief propagation. *IEEE Transactions on Signal Processing*, 58(1):269–280, 2010.
- [18] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.
- [19] J. Behley, V. Steinhage, and A. B. Cremers. Laser-based segment classification using a mixture of bag-of-words. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

- [20] A. Bhandari, A. M. Wallace, and R. Raskar. Super-resolved time-of-flight sensing via fri sampling theory. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 4009–4013. IEEE, 2016.
- [21] A. D. Bimbo and P. Pala. Content-based retrieval of 3d models. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):20–43, Feb. 2006.
- [22] C. M. Bishop, M. Svensén, and C. K. Williams. Gtm: The generative topographic mapping. *Neural computation*, 10(1):215–234, 1998.
- [23] C. Blair, J. Thompson, and N. M. Robertson. Identifying anomalous objects in SAS imagery using uncertainty. In *Information Fusion (Fusion), 2015 18th International Conference on*, pages 1410–1416. IEEE, 2015.
- [24] M. Blum, J. T. Springenberg, J. Wülfing, and M. Riedmiller. A learned feature descriptor for object recognition in rgb-d data. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1298–1303. IEEE, 2012.
- [25] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and computational harmonic analysis*, 27(3):265–274, 2009.
- [26] L. Bo, X. Ren, and D. Fox. Depth kernel descriptors for object recognition. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 821–826. IEEE, 2011.
- [27] L. Bo, X. Ren, and D. Fox. Unsupervised feature learning for rgb-d based object recognition. In *Experimental Robotics*, pages 387–402. Springer, 2013.
- [28] D. Borrmann, J. Elseberg, K. Lingemann, and A. Nüchter. The 3d hough transform for plane detection in point clouds: A review and a new accumulator design. *3D Research*, 2(2):1–13, 2011.
- [29] R. Bracewell. The Fourier transform and its applications. *New York*, 5, 1965.
- [30] S. Briechle, A. Sizov, O. Tretyak, V. Antropov, N. Molitor, and P. Krzystek. Uav-based detection of unknown radioactive biomass deposits in chernobyl’s exclusion

- zone. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42(2), 2018.
- [31] E. O. Brigham and E. O. Brigham. *The fast Fourier transform*, volume 7. Prentice-Hall Englewood Cliffs, NJ, 1974.
- [32] N. Brodu and D. Lague. 3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology. *ISPRS Journal of Photogrammetry and Remote Sensing*, 68:121–134, 2012.
- [33] O. Bryt and M. Elad. Compression of facial images using the k-svd algorithm. *Journal of Visual Communication and Image Representation*, 19(4):270–282, 2008.
- [34] G. Burel and H. Hénocq. Three-dimensional invariants and their application to object recognition. *Signal Processing*, 45(1):1 – 22, 1995.
- [35] P. J. Burt and E. H. Adelson. The Laplacian pyramid as a compact image code. *Communications, IEEE Transactions on*, 31(4):532–540, 1983.
- [36] R. J. Campbell and P. J. Flynn. Eigenshapes for 3d object recognition in range data. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE, 1999.
- [37] E. J. Candès and D. L. Donoho. Ridgelets: A key to higher-dimensional intermittency? *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 357(1760):2495–2509, 1999.
- [38] E. J. Candès and M. B. Wakin. An introduction to compressive sampling. *Signal Processing Magazine, IEEE*, 25(2):21–30, 2008.
- [39] L. Cao, N. C. Coops, J. L. Innes, J. Dai, H. Ruan, and G. She. Tree species classification in subtropical forests using small-footprint full-waveform lidar data. *International Journal of Applied Earth Observation and Geoinformation*, 49:39–51, 2016.

- [40] L. Cao, D. Yang, Q. Wang, Y. Yu, J. Wang, and E. A. Rundensteiner. Scalable distance-based outlier detection over high-volume data streams. In *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*, pages 76–87. IEEE, 2014.
- [41] J. B. Case. Automation in photogrammetry. Technical report, DEFENSE MAPPING AGENCY WASHINGTON DC, 1980.
- [42] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15, 2009.
- [43] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection for discrete sequences: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 24(5):823–839, 2012.
- [44] A. P. Charaniya, R. Manduchi, and S. K. Lodha. Supervised parametric classification of aerial lidar data. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, pages 30–30. IEEE.
- [45] A. Chauve, C. Mallet, F. Bretar, S. Durrieu, M. P. Deseilligny, W. Puech, et al. Processing full-waveform lidar data: modelling raw signals. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 2007*, pages 102–107, 2007.
- [46] A. Chauve, C. Mallet, F. Bretar, S. Durrieu, M. P. Deseilligny, W. Puech, et al. Processing full-waveform lidar data: modelling raw signals. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 2007*, pages 102–107, 2007.
- [47] N. Chehata, L. Guo, and C. Mallet. Contribution of airborne full-waveform lidar and image data for urban scene classification. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 1669–1672. IEEE, 2009.

- [48] N. Chehata, L. Guo, and C. Mallet. Contribution of airborne full-waveform lidar and image data for urban scene classification. In *16<sup>th</sup> IEEE International Conference on Image Processing (ICIP), 2009*, pages 1669–1672. IEEE, 2009.
- [49] C. Chen, B. Yang, S. Song, X. Peng, and R. Huang. Automatic clearance anomaly detection for transmission line corridors utilizing uav-borne lidar data. *Remote Sensing*, 10(4):613, 2018.
- [50] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.
- [51] Y. Cheng, R. Cai, C. Zhang, Z. Li, X. Zhao, K. Huang, and Y. Rui. Query adaptive similarity measure for rgb-d object recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 145–153, 2015.
- [52] P. S. Chhabra, A. Maccarone, A. McCarthy, A. M. Wallace, and G. S. Buller. Discriminating underwater lidar target signatures using sparse multi-spectral depth codes. In *Sensor Signal Processing for Defence (SSPD)*. IEEE, 2016.
- [53] P. S. Chhabra, A. M. Wallace, and J. R. Hopgood. Anomaly detection in clutter using spectrally enhanced ladar. In *SPIE Defense+ Security*, pages 946508–946508. International Society for Optics and Photonics, 2015.
- [54] A. G. Cottin, D. L. Forbes, and B. F. Long. Shallow seabed mapping and classification using waveform analysis and bathymetry from shoals lidar data. *Canadian Journal of Remote Sensing*, 35(5):422–434, 2009.
- [55] P. Csákány and A. M. Wallace. Representation and classification of 3-d objects. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 33(4):638–647, 2003.
- [56] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. R. Bradski. Self-supervised monocular road detection in desert terrain. In *Robotics: science and systems*, volume 38. Philadelphia, 2006.



- [57] T. V. de Guélis, H. Chepfer, R. Guzman, M. Bonazzola, D. M. Winker, and V. Noel. Space lidar observations constrain longwave cloud feedback. *Scientific reports*, 8, 2018.
- [58] T. K. Dey, K. Li, C. Luo, P. Ranjan, I. Safa, and Y. Wang. Persistent heat signature for pose-oblivious matching of incomplete models. In *Computer Graphics Forum*, volume 29, pages 1545–1554. Wiley Online Library, 2010.
- [59] M. N. Do and M. Vetterli. The contourlet transform: an efficient directional multiresolution image representation. *Image Processing, IEEE Transactions on*, 14(12):2091–2106, 2005.
- [60] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck. Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 58(2):1094–1121, 2012.
- [61] M.-P. Dubuisson and A. K. Jain. A modified hausdorff distance for object matching. In *Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, volume 1, pages 566–568. IEEE, 1994.
- [62] R. O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972.
- [63] E. Fakiris, G. Papatheodorou, M. Geraga, and G. Ferentinos. An automatic target detection algorithm for swath sonar backscatter imagery, using image texture and independent component analysis. *Remote Sensing*, 8(5):373, 2016.
- [64] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.
- [65] J. C. Fernandez-Diaz, W. E. Carter, R. L. Shrestha, and C. L. Glennie. Now you see it... now you don't: Understanding airborne mapping lidar collection and data

- product generation for archaeological research in mesoamerica. *Remote Sensing*, 6(10):9951–10001, 2014.
- [66] M. A. Figueiredo, R. D. Nowak, and S. J. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE Journal of selected topics in signal processing*, 1(4):586–597, 2007.
- [67] P. Gagnon, J. Agnard, C. Nolette, M. Boulianne, et al. A microcomputer-based general photogrammetric system. *Photogrammetric Engineering and Remote Sensing*, 56(5):623–625, 1990.
- [68] M. J. Gangeh, A. Ghodsi, and M. S. Kamel. Kernelized supervised dictionary learning. *IEEE Transactions on Signal Processing*, 61(19):4753–4767, 2013.
- [69] R. Garg and R. Khandekar. Gradient descent with sparsification: an iterative algorithm for sparse recovery with restricted isometry property. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 337–344. ACM, 2009.
- [70] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In *Symposium on geometry processing*, volume 2, page 5, 2005.
- [71] N. Gold, M. G. Frasca, C. L. Herry, B. S. Richardson, and X. Wang. A doubly stochastic change point detection algorithm for noisy biological signals. *Frontiers in physiology*, 8:1112, 2018.
- [72] A. Golts and M. Elad. Linearized kernel dictionary learning. *IEEE Journal of Selected Topics in Signal Processing*, 10(4):726–739, 2016.
- [73] J. Gomez and D. Dasgupta. Evolving fuzzy classifiers for intrusion detection. In *Proceedings of the 2002 IEEE Workshop on Information Assurance*, volume 6, pages 321–323. New York: IEEE Computer Press, 2002.
- [74] R. P. Gorman and T. J. Sejnowski. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural networks*, 1(1):75–89, 1988.

- [75] M. Griebel and A. Hullmann. Dimensionality reduction of high-dimensional data with a nonlinear principal component aligned generative topographic mapping. *SIAM Journal on Scientific Computing*, 36(3):A1027–A1047, 2014.
- [76] J. Groen, E. Coiras, J. D. R. Vera, and B. Evans. Model-based sea mine classification with synthetic aperture sonar. *IET radar, sonar & navigation*, 4(1):62–73, 2010.
- [77] J. Groen, E. Coiras, and D. Williams. Detection rate statistics in synthetic aperture sonar images. In *Proc Intl Conf & Exh underwater acoustic measurements*, pages 367–374, 2009.
- [78] C. Grönwall, T. Chevalier, G. Tolt, and P. Andersson. An approach to target detection in forested scenes. In *SPIE Defense and Security Symposium*, pages 69500S–69500S. International Society for Optics and Photonics, 2008.
- [79] C. Grönwall, G. Tolt, T. Chevalier, and H. Larsson. Spatial filtering for detection of partly occluded targets. *Optical engineering*, 50(4):047201–047201, 2011.
- [80] C. Grönwall, G. Tolt, T. Chevalier, and H. Larsson. Spatial filtering for detection of partly occluded targets. *Optical engineering*, 50(4):047201–047201, 2011.
- [81] L. Guo, N. Chehata, C. Mallet, and S. Boukir. Relevance of airborne lidar and multispectral image data for urban scene classification using random forests. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(1):56–66, 2011.
- [82] N. Halko, P.-G. Martinsson, Y. Shkolnisky, and M. Tygert. An algorithm for the principal component analysis of large data sets. *SIAM Journal on Scientific computing*, 33(5):2580–2594, 2011.
- [83] J. C. Harsanyi and C.-I. Chang. Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach. *Geoscience and Remote Sensing, IEEE Transactions on*, 32(4):779–785, 1994.

- [84] S. Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.
- [85] S. Hernandez-Marin, A. M. Wallace, and G. J. Gibson. Bayesian analysis of lidar signals with multiple returns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(12):2170–2180, 2007.
- [86] S. Hernandez-Marin, A. M. Wallace, and G. J. Gibson. Bayesian analysis of lidar signals with multiple returns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(12):2170–2180, 2007.
- [87] R. Hoff. Vertical structure of arctic haze observed by lidar. *Journal of Applied Meteorology*, 27(2):125–139, 1988.
- [88] M. A. Hofton, J. B. Minster, and J. B. Blair. Decomposition of laser altimeter waveforms. *Geoscience and Remote Sensing, IEEE Transactions on*, 38(4):1989–1996, 2000.
- [89] M. A. Hofton, J. B. Minster, and J. B. Blair. Decomposition of laser altimeter waveforms. *Geoscience and Remote Sensing, IEEE Transactions on*, 38(4):1989–1996, 2000.
- [90] B. K. P. Horn. Extended gaussian images. *Proceedings of the IEEE*, 72(12):1671–1686, 1984.
- [91] C. Hug, A. Ullrich, and A. Grimm. Litemapper-5600-a waveform-digitizing lidar terrain and vegetation mapping system. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(Part 8):W2, 2004.
- [92] O. Ibidunmoye, A.-R. Rezaie, and E. Elmroth. Adaptive anomaly detection in performance metric streams. *IEEE Transactions on Network and Service Management*, 15(1):217–231, 2018.
- [93] M. Isenburg. The pulsewaves format, Dec. 2015.

- [94] A. Janssen. Gabor representation of generalized functions. *Journal of Mathematical Analysis and Applications*, 83(2):377–394, 1981.
- [95] S. Ji, Y. Xue, L. Carin, et al. Bayesian compressive sensing. *IEEE Transactions on Signal Processing*, 56(6):2346, 2008.
- [96] Z. Jiang, Z. Lin, and L. S. Davis. Label consistent k-svd: Learning a discriminative dictionary for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2651–2664, 2013.
- [97] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 21(5):433–449, 1999.
- [98] B. Jutzi and U. Stilla. Range determination with waveform recording laser systems using a wiener filter. *{ISPRS} Journal of Photogrammetry and Remote Sensing*, 61(2):95 – 107, 2006.
- [99] B. Jutzi and U. Stilla. Range determination with waveform recording laser systems using a wiener filter. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61(2):95–107, 2006.
- [100] B. Jutzi and U. Stilla. Range determination with waveform recording laser systems using a Wiener filter. *Journal of Photogrammetry and Remote Sensing*, 61(2):95–107, 2006.
- [101] N. B. Karahanoglu and H. Erdogan. A\* orthogonal matching pursuit: Best-first search for compressed sensing signal recovery. *Digital Signal Processing*, 22(4):555–568, 2012.
- [102] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Rotation invariant spherical harmonic representation of 3d shape descriptors. In *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 156–164. Eurographics Association, 2003.

- [103] A. M. Kim. *Simulating full-waveform LIDAR*. PhD thesis, Monterey, California. Naval Postgraduate School, 2009.
- [104] M. Kirchhof, B. Jutzi, and U. Stilla. Iterative processing of laser scanning data by full waveform analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(1):99–114, 2008.
- [105] K. Koh, S.-J. Kim, and S. Boyd. An interior-point method for large-scale  $l_1$ -regularized logistic regression. *Journal of Machine learning research*, 8(Jul):1519–1555, 2007.
- [106] H.-P. Kriegel, P. Kröger, and A. Zimek. Outlier detection techniques. *Tutorial at KDD*, 10, 2010.
- [107] J.-Y. Kwok and I.-H. Tsang. The pre-image problem in kernel methods. *IEEE transactions on neural networks*, 15(6):1517–1525, 2004.
- [108] K. Lai, L. Bo, X. Ren, and D. Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1817–1824. IEEE, 2011.
- [109] K. Lai, L. Bo, X. Ren, and D. Fox. Sparse distance learning for object recognition combining rgb and depth information. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 4007–4013. IEEE, 2011.
- [110] K.-C. Lee, J. Ho, and D. J. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on pattern analysis and machine intelligence*, 27(5):684–698, 2005.
- [111] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya. Learning unions of orthonormal bases with thresholded singular value decomposition. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 5, pages v–293. IEEE, 2005.

- [112] A. S. Lewis and G. Knowles. Image compression using the 2-d wavelet transform. *Image Processing, IEEE Transactions on*, 1(2):244–250, 1992.
- [113] D. Ludwig, A. Kongable, S. Krywick, H. T. Albrecht, G. Kamrath, J. Milam, D. Brown, G. J. Fetzer, and K. Hanna. Identifying targets under trees: jigsaw 3d lidar test results. volume 5086, pages 16–26, 2003.
- [114] 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. *Optics Express*, 23(26):33911–33926, 2015.
- [115] L. A. Magruder and A. L. Neuenschwander. Lidar full-waveform data analysis for detection of faint returns through obscurants. In *SPIE Defense, Security, and Sensing*, pages 73230L–73230L. International Society for Optics and Photonics, 2009.
- [116] J. Mairal, F. Bach, and J. Ponce. Task-driven dictionary learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):791–804, 2012.
- [117] D. Malchiodi and T. Legnani. Avoiding the cluster hypothesis in sv classification of partially labeled data. In *Recent Advances of Neural Network Models and Applications*, pages 33–40. Springer, 2014.
- [118] C. Mallet and F. Bretar. Full-waveform topographic lidar: State-of-the-art. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(1):1–16, 2009.
- [119] C. Mallet, F. Bretar, M. Roux, U. Soergel, and C. Heipke. Relevance assessment of full-waveform lidar data for urban area classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6):S71–S84, 2011.
- [120] C. Mallet, F. Lafarge, M. Roux, U. Soergel, F. Bretar, and C. Heipke. A marked point process for modeling lidar waveforms. *IEEE transactions on image processing*, 19(12):3204–3221, 2010.

- [121] B. C. Matei, Y. Tan, H. S. Sawhney, and R. Kumar. Rapid and scalable 3d object recognition using lidar data. In *Defense and Security Symposium*, pages 623401–623401. International Society for Optics and Photonics, 2006.
- [122] P. McCool, Y. Altmann, A. Perperidis, and S. McLaughlin. Robust markov random field outlier detection and removal in subsampled images. In *Statistical Signal Processing Workshop (SSP), 2016 IEEE*, pages 1–5. IEEE, 2016.
- [123] G. Medioni, C.-K. Tang, and M.-S. Lee. Tensor voting: Theory and applications.
- [124] L. Mora-López and J. Mora. An adaptive algorithm for clustering cumulative probability distribution functions using the kolmogorov–smirnov two-sample test. *Expert Systems with Applications*, 42(8):4016–4021, 2015.
- [125] E. Næsset. Practical large-scale forest stand inventory using a small-footprint airborne scanning laser. *Scandinavian Journal of Forest Research*, 19(2):164–179, 2004.
- [126] Neonscience. Reflections on remote sensing, ecology and the neon aop, 2012.
- [127] J. Neulist and W. Armbruster. Segmentation, classification, and pose estimation of military vehicles in low resolution laser radar images. In *Defense and Security*, pages 218–225. International Society for Optics and Photonics, 2005.
- [128] M. of Defence. Future character of conflict. Technical report, Ministry of Defence, 2010.
- [129] M. Ovsjanikov, Q. Mérigot, F. Mémoli, and L. Guibas. One point isometric matching with the heat kernel. In *Computer Graphics Forum*, volume 29, pages 1555–1564. Wiley Online Library, 2010.
- [130] M. Ovsjanikov, Q. Mérigot, F. Mémoli, and L. Guibas. One point isometric matching with the heat kernel. In *Computer Graphics Forum*, volume 29, pages 1555–1564. Wiley Online Library, 2010.



- [131] Y. Pailhas, C. Capus, K. Brown, and Y. Petillot. *BioSonar: a bio-mimetic approach to sonar systems concepts and applications*. INTECH Open Access Publisher, 2011.
- [132] Y. Pailhas, Y. Petillot, and C. Capus. High-resolution sonars: what resolution do we need for target recognition? *EURASIP Journal on Advances in Signal Processing*, 2010:1–13, 2010.
- [133] H. C. Palm, T. V. Haavardsholm, H. Ajer, and C. V. Jensen. Extraction and classification of vehicles in ladar imagery. In *SPIE Defense, Security, and Sensing*, pages 873102–873102. International Society for Optics and Photonics, 2013.
- [134] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa. Visual domain adaptation: A survey of recent advances. *IEEE signal processing magazine*, 32(3):53–69, 2015.
- [135] Y. C. Pati, R. Rezaifar, and P. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993.
- [136] A. Patterson IV, P. Mordohai, and K. Daniilidis. Object detection from large-scale 3d datasets using bottom-up and top-down descriptors. In *Computer Vision–ECCV 2008*, pages 553–566. Springer, 2008.
- [137] Å. Persson, U. Söderman, J. Töpel, and S. Ahlberg. Visualization and analysis of full-waveform airborne laser scanner data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(3/W19):103–108, 2005.
- [138] Å. Persson, U. Söderman, J. Töpel, and S. Ahlberg. Visualization and analysis of full-waveform airborne laser scanner data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(3/W19):103–108, 2005.
- [139] B. Peterson, W. Ni, B. Blair, M. Hofton, P. Hyde, and R. Dubayah. Modeling lidar waveforms using a radiative transfer model. *INTERNATIONAL ARCHIVES OF PHOTOGRAMMETRY REMOTE SENSING AND SPATIAL INFORMATION SCIENCES*, 34(3/W4):121–124, 2001.

- [140] J. Reitberger, P. Krzystek, and U. Stilla. Analysis of full waveform lidar data for tree species classification. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(Part 3):228–233, 2006.
- [141] J. Reitberger, P. Krzystek, and U. Stilla. Analysis of full waveform lidar data for tree species classification. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(Part 3):228–233, 2006.
- [142] J. Reitberger, P. Krzystek, and U. Stilla. 3d segmentation and classification of single trees with full waveform LiDAR data. In *Proceedings of SilviLaser 2008, 8<sup>th</sup> International Conference on LiDAR Applications in Forest Assessment and Inventory*, pages 216–226, 2008.
- [143] J. Reitberger, P. Krzystek, and U. Stilla. 3d segmentation and classification of single trees with full waveform lidar data. In *Proceedings of SilviLaser 2008, 8<sup>th</sup> International Conference on LiDAR Applications in Forest Assessment and Inventory*, pages 216–226, 2008.
- [144] Riegl. <http://www.riegl.com/nc/products/airborne-scanning/produktdetail/product/scanner/34/>, Aug. 2017.
- [145] Riegl. <http://www.riegl.com/nc/products/airborne-scanning/produktdetail/product/scanner/38/>, Aug. 2017.
- [146] Riegl. <http://www.riegl.com/nc/products/airborne-scanning/produktdetail/product/scanner/46/>, Aug. 2017.
- [147] R. Rubinstein and M. Elad. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, 98(6), 2010.
- [148] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu. Fast 3d recognition and pose using the viewpoint feature histogram. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 2155–2162. IEEE, 2010.

- [149] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz. Learning informative point classes for the acquisition of object model maps. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, pages 643–650. IEEE, 2008.
- [150] F. A. Sadjadi and E. L. Hall. Three-dimensional moment invariants. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (2):127–136, 1980.
- [151] M. Salehi and L. Rashidi. A survey on anomaly detection in evolving data:[with application to forest fire risk prediction]. *ACM SIGKDD Explorations Newsletter*, 20(1):13–23, 2018.
- [152] F. S. Samaria and A. C. Harter. Parameterisation of a stochastic model for human face identification. In *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, pages 138–142. IEEE, 1994.
- [153] A. Samberg. An implementation of the asprs las standard. In *ISPRS Workshop on Laser Scanning and SilviLaser*, pages 363–372, 2007.
- [154] J. Sawas and Y. Petillot. Cascade of boosted classifiers for automatic target recognition in synthetic aperture sonar imagery. In *Proceedings of Meetings on Acoustics*, volume 17, page 070074. Acoustical Society of America, 2013.
- [155] B. Scholkopf and K.-R. Mullert. Fisher discriminant analysis with kernels. *Neural networks for signal processing IX*, 1(1):1, 1999.
- [156] B. Scotland. Wat at sea: Scotland’s story, June 2015.
- [157] J. M. Shapiro. Adaptive Mcclellan transformations for quincunx filter banks. *Signal Processing, IEEE Transactions on*, 42(3):642–648, 1994.
- [158] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng. Convolutional-recursive deep learning for 3d object classification. In *Advances in Neural Information Processing Systems*, pages 665–673, 2012.

- [159] O. Steinvall. Laser radar work at foi. In *Proc. SPIE*, volume 7382, pages 738202–1, 2009.
- [160] U. Stilla, W. Yao, and B. Jutzi. Detection of weak laser pulses by full waveform stacking. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(Part 3):W49A, 2007.
- [161] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer Graphics Forum*, volume 28, pages 1383–1392. Wiley Online Library, 2009.
- [162] Z.-h. Sun, J.-h. Deng, and X.-w. Yan. Target detection and recognition techniques of line imaging lidar sensor. In *International Symposium on Photoelectronic Detection and Imaging 2009*, pages 73824C–73824C. International Society for Optics and Photonics, 2009.
- [163] J. L. Tackett, D. M. Winker, B. J. Getzewich, M. A. Vaughan, S. A. Young, and J. Kar. Calipso lidar level 3 aerosol profile product: version 3 algorithm design. *Atmospheric Measurement Techniques*, 11(7):4129–4152, 2018.
- [164] J.-i. Takeuchi and K. Yamanishi. A unifying framework for detecting outliers and change points from time series. *IEEE transactions on Knowledge and Data Engineering*, 18(4):482–492, 2006.
- [165] K. Thiel and A. Wehr. Performance capabilities of laser-scanners-an overview and measurement principle analysis. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(Part 8):W2, 2004.
- [166] J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on information theory*, 53(12):4655–4666, 2007.
- [167] J. A. Tropp, A. C. Gilbert, and M. J. Strauss. Simultaneous sparse approximation via greedy pursuit. In *Proceedings.(ICASSP’05). IEEE International Conference*

- on Acoustics, Speech, and Signal Processing, 2005.*, volume 5, pages v–721. IEEE, 2005.
- [168] J. A. Tropp, A. C. Gilbert, and M. J. Strauss. Algorithms for simultaneous sparse approximation. part i: Greedy pursuit. *Signal Processing*, 86(3):572–588, 2006.
- [169] C.-W. Tsai, C.-F. Lai, H.-C. Chao, and A. V. Vasilakos. Big data analytics: a survey. *Journal of Big data*, 2(1):21, 2015.
- [170] B. L. Ulich. Imaging lidar system, Aug. 29 1989. US Patent 4,862,257.
- [171] M. Vaidyanathan, S. Blask, T. Higgins, W. Clifton, D. Davidsohn, R. Carson, V. Reynolds, J. Pfannenstiel, R. Cannata, R. Marino, J. Drover, R. Hatch, D. Schue, R. Freehart, G. Rowe, J. Mooney, C. Hart, B. Stanley, J. McLaughlin, E.-I. Lee, J. Berenholtz, B. Aull, J. Zayhowski, A. Vasile, P. Ramaswami, K. Ingersoll, T. Amoruso, I. Khan, W. Davis, and R. Heinrichs. Jigsaw phase iii: a miniaturized airborne 3-d imaging laser radar with photon-counting sensitivity for foliage penetration. volume 6550, pages 65500N–65500N–12, 2007.
- [172] M. Vaidyanathan, S. Blask, T. Higgins, W. Clifton, D. Davidsohn, R. Carson, V. Reynolds, J. Pfannenstiel, R. Cannata, R. Marino, et al. Jigsaw phase iii: a miniaturized airborne 3-d imaging laser radar with photon-counting sensitivity for foliage penetration. In *Defense and Security Symposium*, volume 6550, pages 65500N–65500N. International Society for Optics and Photonics, 2007.
- [173] H. Van Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa. Kernel dictionary learning. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2021–2024. IEEE, 2012.
- [174] J.-P. Vandeborre, V. Couillet, and M. Daoudi. A practical approach for 3d model indexing by combining local and global invariants. In *3D Data Processing Visualization and Transmission, 2002. Proceedings. First International Symposium on*, pages 644–647. IEEE, 2002.

- [175] J. Vauhkonen, T. Hakala, J. Suomalainen, S. Kaasalainen, O. Nevalainen, M. Vastaranta, M. Holopainen, and J. Hyypä. Classification of spruce and pine trees using active hyperspectral lidar. *IEEE Geoscience and Remote Sensing Letters*, 10(5):1138–1141, 2013.
- [176] W. Wagner, A. Ullrich, V. Ducic, T. Melzer, and N. Studnicka. Gaussian decomposition and calibration of a novel small-footprint full-waveform digitising airborne laser scanner. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60(2):100–112, 2006.
- [177] W. Wagner, A. Ullrich, V. Ducic, T. Melzer, and N. Studnicka. Gaussian decomposition and calibration of a novel small-footprint full-waveform digitising airborne laser scanner. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60(2):100–112, 2006.
- [178] 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. 2014.
- [179] A. M. Wallace, J. Ye, N. J. Krichel, A. McCarthy, R. J. Collins, and G. S. Buller. Full wave form analysis for long-range 3d imaging laser radar. *EURASIP Journal on Advances in Signal Processing*, 2010:33, 2010.
- [180] A. M. Wallace, J. Ye, N. J. Krichel, A. McCarthy, R. J. Collins, and G. S. Buller. Full wave form analysis for long-range 3D imaging laser radar. *EURASIP Journal on Advances in Signal Processing*, 2010:33, 2010.
- [181] G. Walton, M. S. Diederichs, K. Weinhardt, D. Delaloye, M. J. Lato, and A. Punkkinen. Change detection in drill and blast tunnels from point cloud data. *International Journal of Rock Mechanics and Mining Sciences*, 105:172–181, 2018.
- [182] J. Wang, S. Kwon, and B. Shim. Generalized orthogonal matching pursuit. *IEEE Transactions on signal processing*, 60(12):6202, 2012.

- [183] K. F. West, B. N. Webb, J. R. Lersch, S. Pothier, J. M. Triscari, and A. E. Iverson. Context-driven automated target detection in 3d data. In *Defense and Security*, pages 133–143. International Society for Optics and Photonics, 2004.
- [184] C. Williams and M. Seeger. Using the nyström method to speed up kernel machines. In *Proceedings of the 14th annual conference on neural information processing systems*, number EPFL-CONF-161322, pages 682–688, 2001.
- [185] O. J. Woodford, M.-T. Pham, A. Maki, F. Perbet, and B. Stenger. Demisting the hough transform for 3d shape recognition and registration. *International Journal of Computer Vision*, 106(3):332–341, 2014.
- [186] D. Wu, M. Yaghoobi, S. Kelly, M. Davies, and R. Clewes. A sparse regularized model for raman spectral analysis. In *Sensor Signal Processing for Defence (SSPD), 2014*, pages 1–5. IEEE, 2014.
- [187] J. Wu, J. Van Aardt, and G. P. Asner. A comparison of signal deconvolution algorithms based on small-footprint lidar waveform simulation. *IEEE Transactions on Geoscience and Remote Sensing*, 49(6):2402–2414, 2011.
- [188] M. Yaghoobi, T. Blumensath, and M. E. Davies. Dictionary learning for sparse approximations with the majorization method. *Signal Processing, IEEE Transactions on*, 57(6):2178–2191, 2009.
- [189] M. Yang, L. Zhang, X. Feng, and D. Zhang. Fisher discrimination dictionary learning for sparse representation. In *2011 International Conference on Computer Vision*, pages 543–550. IEEE, 2011.
- [190] J. Ye, A. M. Wallace, A. Al Zain, and J. Thompson. Parallel bayesian inference of range and reflectance from ladar profiles. *Journal of Parallel and Distributed Computing*, 73(4):383–399, 2013.
- [191] Q. Zhang and B. Li. Discriminative k-svd for dictionary learning in face recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2691–2698. IEEE, 2010.

- [192] T. Zhang, R. Ramakrishnan, and M. Livny. Birch: an efficient data clustering method for very large databases. In *ACM Sigmod Record*, volume 25, pages 103–114. ACM, 1996.
- [193] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang. A survey of sparse representation: algorithms and applications. *IEEE Access*, 3:490–530, 2015.
- [194] N. Zhao, Y. Hu, and M. He. Target detection method based on the single laser return waveform. In *International Symposium on Photoelectronic Detection and Imaging 2009*, pages 73822A–73822A. International Society for Optics and Photonics, 2009.
- [195] H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006.