Joint modeling and learning approaches for hyperspectral imaging and changepoint detection

Xiuheng Wang

Approches conjointes de modélisation et d'apprentissage pour l'imagerie hyperspectrale et la détection de changements

Xiuheng Wang

Laboratoire J.-L. Lagrange

Présentée en vue de l'obtention du grade de docteur en Sciences pour l'ingénieur
d'Université Côte d'Azur
Dirigée par : Cédric Richard
Soutenue le : 27 / 06 / 2024

Devant le jury, composé de :
Jean-Yves Tourneret, Professeur, Toulouse INP, France
Arnaud Breloy, Professeur, CNAM, France
Geert Leus, Professor, TU Delft, Netherlands
Gersende Fort, Directrice de recherche, IMT, France
Ricardo Borsoi, Chargé de recherche,
Université de Lorraine, France

Joint Modeling and Learning Approaches for Hyperspectral Imaging and Changepoint Detection

Jury:

Rapporteurs
Jean-Yves Tourneret, Professeur, Toulouse INP, France
Arnaud Breloy, Professeur, Conservatoire National des Arts et Métiers, France

Examinateurs
Geert Leus, Professor, Delft University of Technology, Netherlands
Gersende Fort, Directrice de Recherche, Institut de Mathématiques de Toulouse, France
Ricardo Borsoi, Chargé de recherche, Université de Lorraine, France

Encadrant
Cédric Richard, Professeur, Université Côte d'Azur, France

Joint Modeling and Learning Approaches for Hyperspectral Imaging and Changepoint Detection

Abstract

In the era of artificial intelligence, there has been a growing consensus that solutions to complex science and engineering problems require novel methodologies that can integrate interpretable physics-based modeling approaches with machine learning techniques, from stochastic optimization to deep neural networks. This thesis aims to develop new methodological and applied frameworks for combining the advantages of physics-based modeling and machine learning, with special attention to two important signal processing tasks: solving inverse problems in hyperspectral imaging and detecting changepoints in time series. The first part of the thesis addresses learning priors in model-based optimization for solving inverse problems in hyperspectral imaging systems. First, we introduce a tuning-free Plug-and-Play algorithm for hyperspectral image deconvolution (HID). Specifically, we decompose the optimization problem into two iterative sub-problems, learn deep priors to solve the blind denoising sub-problem with neural networks, and estimate hyperparameters with a measure of the statistical whiteness of the residual. Second, we introduce an original hyperspectral and multispectral image fusion (HMIF) method. It leverages neural networks to learn image priors from data to solve the optimization problem accounting for inter-image variability. We also propose a zero-shot strategy to learn the image-specific priors in an unsupervised manner. The second part of the thesis focuses on modeling changes in data distribution and learning knowledge of time series signals to detect changepoints. First, we propose a changepoint detection (CPD) method using an online approach based on neural networks and continual learning to directly estimate the density ratio between current and reference windows of the data stream. Second, we introduce a non-parametric algorithm for online CPD in manifold-valued data and provide theoretical bounds on the detection and false alarm rate performances using a new result on the non-asymptotic convergence of the stochastic Riemannian gradient descent. Finally, we extend this algorithm to distributed CPD in streaming manifold-valued signals over graphs with a parallel implementation of a graph filter. This significantly improves the detection of changepoints in unknown communities of networks.

Keywords: physics-based modeling, machine learning, hyperspectral images, inverse problems, changepoint detection, Riemannian manifolds.

Approches conjointes de modélisation et d'apprentissage pour l'imagerie hyperspectrale et la détection de changements

Résumé

À l'ère de l'intelligence artificielle, on observe un consensus croissant sur le fait que des solutions à des problèmes complexes de science et d'ingénierie nécessitent de nouvelles méthodologies qui peuvent associer des méthodes de modélisation physique à des techniques d'apprentissage automatique, en recourant à l'optimisation stochastique et aux réseaux neuronaux profonds. Cette thèse vise à étudier des méthodes permettant de combiner les avantages de la modélisation basée sur la physique et de l'apprentissage automatique, en accordant une attention particulière à deux questions importantes de traitement du signal : la résolution de problèmes inverses en imagerie hyperspectrale et la détection de changements dans des séries temporelles. La première partie de la thèse traite de l'apprentissage d'informations a priori dans une démarche orientée modèles pour la résolution de problèmes inverses en imagerie hyperspectrale. Tout d'abord, nous introduisons un algorithme Plug-and-Play pour la déconvolution des images hyperspectrales (HID). Plus précisément, nous décomposons le problème d'optimisation correspondant en deux sous-problèmes résolus itérativement, et apprenons les informations a priori profondes pour résoudre le sous-problème de débruitage aveugle à l'aide de réseaux neuronaux, et estimons les hyperparamètres à l'aide d'une mesure de la blancheur résiduelle. Dans un second temps, nous introduisons une méthode originale de fusion d'images hyperspectrales et multispectrales (HMIF). Elle s'appuie sur des réseaux neuronaux pour apprendre des informations a priori sur les images, à partir de données, dans le but de résoudre le problème d'optimisation en tenant compte de la variabilité entre les images. Nous proposons également une stratégie "zero-shot" pour apprendre l'a priori spécifique à l'image de manière non supervisée. La deuxième partie de la thèse se concentre sur l'apprentissage de séries temporelles en vue de détection de changements. Tout d'abord, nous proposons une méthode de détection de changements (CPD) utilisant une approche en ligne reposant sur des réseaux neuronaux et leur apprentissage en continu pour estimer directement le rapport des fonctions de densité des observations entre deux fenêtres temporelles glissantes. Ensuite, nous introduisons un algorithme non paramétrique pour la CPD en ligne dans le cas de signaux multi-variés. Nous fournissons des limites théoriques sur les taux de détection et de fausse alarme à l'aide d'un nouveau résultat sur la convergence non asymptotique de l'algorithme de gradient stochastique sur des variétés riemanniennes. Enfin, nous étendons cet algorithme à la CPD distribuée en ligne sur des signaux multi-variés sur graphes, et proposons une implémentation parallèle sur graphe. Ceci améliore considérablement la détection de changements survenant dans des communautés fortement connectées, inconnues, des réseaux.

Mots clés : modélisation basée sur la physique, apprentissage automatique, images hyperspectrales, problèmes inverses, détection de changements, variétés riemanniennes.

Acknowledgment

This thesis and its successful defense represent a significant milestone in my life. I would like to express my deepest gratitude to everyone who has supported me on this journey. First and foremost, I am profoundly grateful to my supervisor, Prof. Cédric Richard, for his unwavering support throughout my doctorate program and his invaluable guidance in my study and research career. From him, I learned the importance of maintaining an innovative attitude toward impressive work, this continued my belief in staying in academia. He also provided me the opportunity to get in touch with distinguished researchers and participate in signal processing and machine learning conferences to present in front of peers, which spread my work and sparked potential collaborations.

I extend my heartfelt thanks to CNRS researcher Ricardo Borsoi, an academic rising star, for his valuable guidance and assistance in the past three years. Our countless discussions over these years are an integral part of most work in this thesis. For Prof. Jie Chen, my "academic enlightener", goes my deep gratitude, especially for his priceless advice during the early stages of my academic career.

I also deeply appreciate Prof. Jean-Yves Tourneret, Arnaud Breloy, Geert Leus, and CNRS senior researcher Gersende Fort for serving as thesis committee members. A special thank you to Prof. Tourneret and Breloy for reviewing my thesis manuscript and their detailed reports, which provided me with invaluable feedback.

My sincere thanks go to Prof. André Ferrari for his invaluable guidance and assistance on distributed change point detection over graphs. I am deeply grateful to Dr. Martijn van den Ende, from whom I gained extensive knowledge about earthquake detection and distributed acoustic sensing. My heartfelt thanks also go to Prof. Susanto Rahardja and Dr. Min Zhao for their priceless suggestions and collaboration on the review paper on hyperspectral unmixing. Their contributions have significantly enriched my work. I also deeply appreciate Dr. Céline Theys, Dr. Elena Lega, and Prof. Simon Prunet kindly serving as monitoring committee members during my doctorate program.

I have been fortunate to have a great group of friends, including but not limited to Yacine, Mary-Joe, Rahul, Mengfei, Joachim, Amel, Junjie, Yingbo, Nayeem, Sunwise, Mathieu, Priyam, Atakan, Ying, Keyu, Tristan, Dinil, Muhammed, Alohosty, and Parinaz. My spare time in France was made immensely enjoyable, in large part due to these friends. A special mention to Yacine Khacef, with whom I had the pleasure of sharing an office and many memorable moments. I also acknowledge the support of the administrative staff in both the laboratory and university, especially Julie Frisetti, whose helpfulness and warm-hearted nature made my journey smoother.

Lastly, and most importantly, I want to thank all my family and my dear Siyu Peng for their unwavering love and continued companionship throughout my doctorate journey. Their support has been my greatest source of strength, especially during the tough time of the coronavirus pandemic. I also extend my thanks to all my friends in China, in particular to the Toupifama group, for the strength that they gave me throughout these years.

Table of contents

List of figures

List of tables

List of symbols

General notation

a, A	lowercase and uppercase lightface letters denote scalars
$\boldsymbol{a}$	lowercase boldface letters denote column vectors in Euclidean spaces or points on Riemannian manifolds
$\boldsymbol{A}$	uppercase boldface letters denote matrices
$\propto$	proportional to
$\approx$	approximated to
$\boldsymbol{a} \sim F(\boldsymbol{a})$	$\boldsymbol{a}$ follows distribution F
$\sum$, $\sum_{i=1}^{N}$	summation
$\lvert a \rvert$	absolute value of a
$\lVert \boldsymbol{a} \rVert$	Euclidean norm of a vector $\boldsymbol{a}$
$\lVert \boldsymbol{a} \rVert_p$	ℓ_p-norm of a vector $\boldsymbol{a}$
$\mathbb{E}\boldsymbol{a}$	expectation of $\boldsymbol{a}$
$\lVert \boldsymbol{A} \rVert_F$	Frobenius norm of a matrix $\boldsymbol{A}$
$\lVert \boldsymbol{A} \rVert_p$	Entrywise L_p norm of a matrix $\boldsymbol{A}$
$\min\{a, b\}$	smaller value in a and b
$\inf(\cdot)$	infimum of a set

Sets and spaces

$\{i, \ldots, j\}$	set of integers between i and j
$\{\boldsymbol{a}_i, \ldots, \boldsymbol{a}_j\}$	set of vectors between $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$
$\mathbb{N}$	non-negative integer space
$\mathbb{N}_+$	positive integer space
$\mathbb{R}_+$	positive real number space
$\mathbb{R}^d$	d-dimensional Euclidean space
$\mathcal{M}$	Riemannian manifold
$T_x\mathcal{M}$	tagent space of $\mathcal{M}$ at x
$\mathcal{S}_p^{++}$	manifold of SPD matrices
$\mathcal{G}_p^k$	Grassmann manifold
$\mathcal{G}$	undirected graph
$\mathcal{C}$	community of an undirected graph

Matrix and vector

$[a_1, a_2, \ldots, a_N]$	$(1 \times N)$ vector with components a_i, $i = 1, \ldots, N$
$[\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_N]$	matrix with column $\boldsymbol{a}_i$, $i = 1, \ldots, N$
$[\boldsymbol{A}]_{ij}$	(i, j)th element of a matrix $\boldsymbol{A}$

$\boldsymbol{I}$	identity matrix
$\mathbf{1}$	vector of ones
$\mathbf{0}$	vector of zeros
$\boldsymbol{A}^\top$	transpose of a matrix $\boldsymbol{A}$
$\boldsymbol{A}^{-1}$	inverse of a matrix $\boldsymbol{A}$
$tr\{\boldsymbol{A}\}$	trace of a matrix $\boldsymbol{A}$
$\mathrm{Diag}\{\boldsymbol{A}\}$	vector composed by diagonal elements of a matrix $\boldsymbol{A}$
$\max(\boldsymbol{A})$	maximum element of a matrix $\boldsymbol{A}$
$\mathrm{mean}(\boldsymbol{A})$	mean value of all elements of a matrix $\boldsymbol{A}$
$\mathrm{vec}(\boldsymbol{A})$	Vector obtained by stacking the columns of $\boldsymbol{A}$ on top of each others
$\|\boldsymbol{A}\|_{p,p}$	sum of the column ℓ_p-norms of a matrix $\boldsymbol{A}$
$\boldsymbol{A} * \boldsymbol{B}$	Convolution of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$
$\boldsymbol{A} \star \boldsymbol{B}$	Correlation of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$
$\boldsymbol{A} \odot \boldsymbol{B}$	Hadamard product of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$

List of abbreviations

3DFTV	3D Fractional Total Variation
ADMM	Alternating Direction Method of Multipliers
ARL	Average Run Length
ARMA	Autoregressive Moving Average
B3DDN	Blind 3D Deep Denoising Network
BCE	Binary Cross Entropy
BCD	Block Coordinate Descent
BN	Batch Normalization
CAVE	Columbia Multispectral Database
CBC	Circulant-Block-Circulant
CNN	Convolutional Neural Network
CPD	Changepoint Detection
CUSUM	Cumulative sum
daGFSS	distributed adaptive Graph Fourier Scan Statistic
DFT	Discrete Fourier Transform
DIFIV	Deep Image Fusion with Inter-image Variability
DP	Discrepancy Principle
ERGAS	Erreur Relative Globale Adimensionnelle de Synthèse
EWMAs	Exponentially Weighted Moving Averages
FFT	Fast Fourier Transform
GCV	Generalized Cross-Validation
GFSS	Graph Fourier Scan Statistic
GLRT	Generalized Likelihood Ratio Test
GMM	Gaussian Mixture Model
HI	Hyperspectral Image
HID	Hyperspectral Image Deconvolution
HMIF	Hyperspectral and Multispectral Image Fusion
HLP	Hyper-Laplacian Priors
HRI	High-Resolution Image
HQS	Half Quadratic Splitting
i.i.d.	Independent and Identically Distributed

KL	Kullback-Leibler
KLIEP	Kullback-Leibler Divergence based Importance Estimation Procedure
kNN	k-Nearest Neighbors
LMM	Linear Mixing Model
MI	Multispectral Image
MRI	Magnetic Resonance Imaging
MDD	Mean Detection Delay
NLM	Non-Local Means
PCA	Principal Component Analysis
PDF	Probability Density Function
PSF	Point Spread Function
PSNR	Peak-Signal-to-Noise-Ratio
RED	Regularization by Denoising
RGB	Red, Green, Blue
ROC	Receiver Operating Characteristic
RMSE	Root Mean-Square Error
R-SGD	Riemannian Stochastic Gradient Descent
SGD	Stochastic Gradient Descent
SI	Subspace Identification
SSIM	Structural SIMilarity
SSP	Spatial and Spectral Priors
SURE	Stein's Unbiased Risk Estimate
SVD	Singular Value Decomposition
SRF	Spectral Response Functions
SSIM	Structural SIMilarity
STFT	Short Time Fourier Transform
TV	Total Variation
uLSIF	Unconstrained Least Squares Importance Fitting
VC	Virtual Classifier
VCL	Variational Continual Learning
WLRTR	Weighted Low-Rank Tensor Recovery
w.r.t.	with respect to

Chapter 1

Introduction

Contents

1.1 Joint Modeling and Learning Approaches

In the field of signal processing, physics-based modeling approaches and machine learning techniques are two important methodologies that are often investigated individually. Traditional physics-based approaches incorporate system, signal, and noise models constructed based on a representation of the physical mechanisms underlying the data generation process and admit a clear interpretation. Nevertheless, the physical processes underlying many real applications such as hyperspectral imaging can be extremely complex and are often not completely known. Thus, physics-based models defined a priori are often inaccurate, which limits their performance in downstream tasks. Data-driven methods using machine learning, from stochastic optimization [Bottou 2010, Bottou 2018] to deep neural networks [Krizhevsky 2012, LeCun 2015], have developed rapidly in recent years. Their performance in many problems has surpassed that of classical methods thanks to their using less assumptions on the data distribution and having superior capability in directly learning information from data. However, applying such machine learning techniques as black boxes to perform signal processing tasks may lead to low interpretability and poor generalization ability, especially when the amount of data available for training is small.

To bring together the best of two worlds, recent research efforts have focused on combining the advantages of physics-based and data-driven methods [Wen 2023, Kadambi 2023, Shlezinger 2023] to explore the continuum between domain-specific knowledge and machine learning. In this way, superior performance with clear interpretation can be achieved with a principled design of the models and integration methodologies to leverage available physical knowledge without restricting the representation capability of the model. This thesis aims to advance the state of the

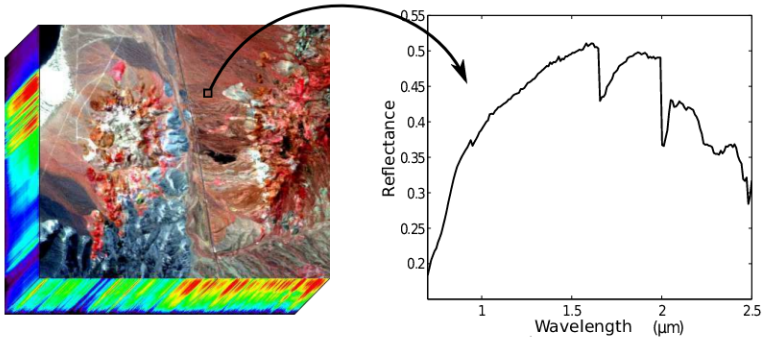

FIGURE 1.1 – Illustration of a pixel from a hyperspectral image [Borsoi 2021b].

art in both theoretical and algorithmic (applied) aspects of this topic, with special attention to two important signal processing tasks: solving inverse problems in hyperspectral imaging [Rasti 2021, Yokoya 2017a] and detecting changepoints in time series [Aminikhanghahi 2017, Truong 2020]. The former problem is mainly concerned with accurately reconstructing a signal while the latter consists in detecting shifts or anomalies in the behavior of signals over time, and one can find interesting applications which combine both of them including detecting changepoints in multi-temporal hyperspectral image sequences [Borsoi 2021c, Borsoi 2021f]. Now we shall give a brief introduction to these two tasks below.

1.1.1 Solving inverse problems in hyperspectral imaging

As illustrated in FIGURE 1.1, hyperspectral imaging systems simultaneously capture images of a scene over continuous narrow spectral bands ranging from the ultraviolet to the visible and infrared spectra. The high spectral resolution provided by hyperspectral images (HIs) enables us to conduct analyses that cannot be performed with conventional imaging techniques. Their rich spectral information has attracted interest in many applications such as remote sensing for mineral exploration, vegetation monitoring, and land cover analysis [Bioucas-Dias 2013] as well as computer vision for object detection [Yan 2021]. However, due to various physical and hardware limitations, the acquisition process induces various degradations. For example, observed HIs are usually blurred and corrupted by noise during the acquisition process. Moreover, the high spectral resolution of HIs and typical sensor-to-target distances limits their spatial resolution [Shaw 2003]. These degradations on the acquired images can reduce the accuracy of downstream processing applications of interest. Thus, it is desirable to restore images by modeling the inversion of the degradation process and consequently solving inverse problems beforehand.

In response to various forms of degradation, a range of specific tasks has been investigated, including but not limited to HI denoising, deconvolution, and super-resolution (fusion) [Rasti 2021, Yokoya 2017a]. Each of these tasks necessitates the solution to a corresponding inverse problem to enhance the image quality. Despite the diverse nature of degradations affecting the hyperspectral imaging process, each

giving rise to distinct hyperspectral image degradation phenomena, it is possible to unify these different phenomena mathematically by establishing a general model. This model allows for the representation of the diverse degradation types as the following linear model:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}, \tag{1.1}$$

where $\boldsymbol{y}$ and $\boldsymbol{x}$ denote observed and latent HIs, respectively, $\boldsymbol{H}$ is a degradation matrix, and $\boldsymbol{n}$ is independent and identically distributed (i.i.d.) Gaussian noise. Problem (1.1) with different meanings of $\boldsymbol{H}$ represents distinct hyperspectral inverse imaging problems:

— $\boldsymbol{H}$ is an identity matrix: HI denoising problem.
— $\boldsymbol{H}$ is a convolution matrix: HI deconvolution problem.
— $\boldsymbol{H}$ is a composite matrix of convolution and down-sampling matrices: HI super-resolution problem.

Note the HI super-resolution problem is often addressed with the fusion of multispectral images (MIs) of the same scene, which have higher spatial but lower spectral resolution. This problem is consequently called as HMIF problem. One challenge in solving (1.1) is that matrix $\boldsymbol{H}$ is typically ill-conditioned, making the process of solving for $\boldsymbol{x}$ involving the inversion of $\boldsymbol{H}$ ill-posed and highly unstable. To tackle this issue, regularization strategies can be employed to use additional prior information to constrain the solution space, leading to a stable solution to (1.1). To provide a statistical interpretation for these strategies, we will now formulate the optimization models of solving (1.1) from a Bayesian inference viewpoint.

The distribution of $\boldsymbol{y}$ conditioned on $\boldsymbol{x}$ can be determined by the noise distribution:

$$p(\boldsymbol{y}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{H}\boldsymbol{x}, \sigma^2\boldsymbol{I}), \tag{1.2}$$

where $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ represents Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. From Bayes rule, we can compute the posterior distribution of $\boldsymbol{x}$ as

$$p(\boldsymbol{x}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x}), \tag{1.3}$$

where $p(\boldsymbol{x})$ denotes the prior distribution of $\boldsymbol{x}$ and $\propto$ means "proportional to". Finally, the log-posterior distribution can be written as

$$-\log p(\boldsymbol{x}|\boldsymbol{y}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2 + \log p(\boldsymbol{x}) + K\,, \tag{1.4}$$

where K is a constant. Different prior distributions of $p(\boldsymbol{x})$ can be considered in the literature to solve ill-posed inverse problems. In the sense of the maximum a posteriori (MAP) principle, we can estimate $\boldsymbol{x}$ by seeking the minimum of the following degradation model-based optimization problem:

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2 + \eta\Phi(\boldsymbol{x}), \tag{1.5}$$

where $\frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2$ represents the data fidelity term and $\Phi(\boldsymbol{x})$ is the regularization term (which is related to the choice of prior) that enforces desirable properties of

the solution with a regularization parameter η. In parametric models, the choice of $\Phi(\boldsymbol{x})$ is defined based on prior knowledge about the signal properties or about the model defining $p(\boldsymbol{x})$. In contrast, this thesis focuses on *non-parametric* strategies, i.e., without strong assumption on parametric form of $p(\boldsymbol{x})$.

1.1.2 Detecting changepoints in time series

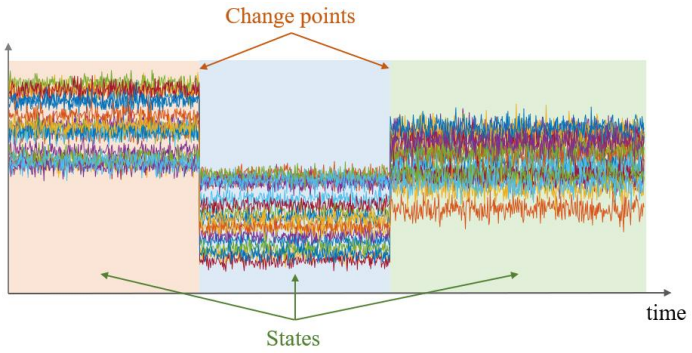

FIGURE 1.2 – Illustration of changepoints between different states of a multivariate time series.

Another common task in signal processing is the identification and analysis of complex systems whose underlying state changes over time. Changepoint detection (CPD) is the problem of finding abrupt variations in the statistical properties of time series data, which may indicate transitions between different states [Aminikhanghahi 2017, Truong 2020]. As a fundamental problem in statistics and signal processing, CPD has seen major interest from the community in the past decades and has been applied to fields as diverse as medical condition monitoring [Gajic 2015], speech recognition [Rybach 2009] and image analysis [Borsoi 2021f]. In addition, this problem also plays a central role in the modeling, analysis, and prediction of time series data, and it has been addressed in many applications ranging from remote sensing [Zeng 2020] and climatology [Reeves 2007] to financial data analysis [Bai 1998].

Let us consider a time series of d-dimensional vector-valued data $\{\boldsymbol{x}_t\}_{t\in\mathbb{N}}$, with $\boldsymbol{x}_t \in \mathbb{R}^d$. We assume that there exists a time index $t_r \in \mathbb{N}$ with an abrupt change in the statistical distribution of $\boldsymbol{x}_t$, that is:

$$t < t_r : \ \boldsymbol{x}_t \sim p(\boldsymbol{x}) \,, \qquad t \geqslant t_r : \ \boldsymbol{x}_t \sim q(\boldsymbol{x}) \,, \tag{1.6}$$

where $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$, which are assumed to be different, denote the probability density functions (PDFs) of the data before and after t_r. The latter is the so-called *changepoint*, as illustrated in FIGURE 1.2. To make the presentation clearer, without loss of generality, the problem in (1.6) presents only a single changepoint for simplicity, but CPD algorithms are typically designed to handle multiple changepoints.

The CPD problem consists of estimating the changepoint $\hat{t}_r$ that is as close as possible to the true changepoint t_r. In this thesis, we consider a more general version

of this problem, in which $\{\boldsymbol{x}_t\}$ might contain multiple changepoints, and $\boldsymbol{x}_t$ is a streaming signal that is observed sequentially over time. We address the requirement that changepoints must be detected *online*, i.e., we need to decide whether each time instant $t \in \mathbb{N}$ is a changepoint based only on past data $\{\boldsymbol{x}_{t'}\}_{t' \leqslant t}$. This leads to two objectives when designing an online CPD algorithm: minimizing the probability of a false alarm (of flagging $t \neq t_r$ as a changepoint), and minimizing the detection delay, i.e., $\hat{t}_r - t_r$ for $\hat{t}_r$ being the first detection after t_r. Moreover, this thesis focuses on non-parametric strategies, in which no parametric form is assumed for the probability measures $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$.

1.2 Motivations and main contributions

The objective of this thesis is to develop new frameworks for integrating physics-based modeling and machine learning methods, with special attention to solving inverse problems in hyperspectral imaging and detecting changepoints in time series. The motivation of this thesis is twofold regarding these two signal processing tasks respectively.

In the hyperspectral imaging task, this thesis formulates and solves the inverse problems to address three challenges, including HI denoising, deconvolution, and fusion. The hyperspectral imaging inverse problems consist of dealing with their ill-posedness and can be formulated as regularized optimization problems. A variety of physics-based modeling methods have been developed with various hand-crafted regularizers to promote the sparsity, spatial continuity, and edge-preservation of images, which are useful prior information. These regularizers play a key role in improving the performance and enhancing the stability of the inversion processing. However, it is a non-trivial task to handcraft a powerful regularizer, and complex regularizers may introduce extra difficulties in solving optimization problems, especially in the case where they are non-differentiable or non-convex. In recent years, inspired by the success of deep learning, convolutional neural networks (CNNs) have been used to restore images end-to-end. These data-driven methods require less handcrafted prior knowledge of images and have been shown to achieve significant performance enhancement compared to physics-based modeling methods. However, they need massive amounts of data for training and may not be consistent with the physical degradation model. To tackle the above issues, this thesis will investigate the integration of the merits of both physics-based modeling and deep learning methods.

For the time series CPD task, the thesis considers sequential inputs with weaker assumptions about the data distribution compared to parametric methods to detect changepoints. The non-parametric and online CPD in time series can be addressed by various physics-based modeling of the changes in data distributions, such as estimating density ratio in sliding windows, or monitoring mean, variance, or general statistics of time series. Based on these physics-based models, various methods have been proposed to detect changepoints, such as kernel-based density ratio estimation, monitoring maximum mean discrepancy, and random features of time series with

exponentially weighted moving averages. However, kernel-based methods or hand-crafted features may not be sufficient to handle complex scenarios, such as data belonging to non-Euclidean spaces. For instance, the performance of kernel-based methods and hand-crafted features heavily depends on the choice of the kernel function and the design of the features. Identifying an appropriate kernel or feature set and fine-tuning its parameters is often a non-trivial task, requiring domain expertise and extensive experimentation. For very complex patterns or highly non-linear data, even sophisticated kernels or hand-crafted features might not capture the underlying structure effectively. Recently, machine learning, including deep neural networks and stochastic optimization, has shown its powerful learning and adaptation ability in many signal processing tasks [Bottou 2018, Wen 2023]. However, directly applying these powerful tools to CPD is not trivial due to the lack of explicit learning objectives, as this problem is different from traditional supervised and unsupervised learning tasks. Therefore, this thesis will investigate appropriate designs of the learning objectives derived from physics-based models and then leverage machine learning to learn information from data to detect changepoints in time series.

Concentrating on the problems and the motivations presented above, the main contributions of this thesis are the following as below:

- Proposition of a joint modeling and learning approach with application to HI deconvolution, i.e., a Plug-and-Play algorithm with deep prior and design of a parameter turning-free mechanism.
- Derivation of a joint modeling and learning algorithm accounting for hyperspectral and multispectral image fusion (HMIF) problem with inter-image variability, including a general imaging model, an iteratively reweighted optimization scheme with deep image-specific prior learning.
- Proposition of a joint modeling and learning approach with a neural online density-ratio estimator for online and non-parametric CPD in Euclidean spaces, which leverages neural networks to learn density ratio between test and reference sliding windows, in the form of a continual learning problem.
- Derivation of a joint modeling and learning framework for non-parametric online CPD on Riemannian manifolds, analysis of the performance guarantees of this algorithm based on a new theoretical finding on the convergence of Riemannian stochastic optimization, and application of this algorithm to two common instances of manifolds.
- Development of distributed CPD for streaming manifold-valued signals over networks with a parallel implementation of a graph filter.

1.3 Thesis organization and contents

The main body of this thesis is divided into two parts. The first part consists of Chapters 2 and 3, and concerns joint modeling and learning approaches for solving inverse problems in hyperspectral imaging. In Chapter 2, we present a tuning-free Plug-and-Play framework for HI deconvolution with deep priors. In Chapter 3, we

address the unsupervised deep HMIF method accounting for inter-image variability. The second part consists of Chapters 4, 5, and 6, and investigates joint modeling and learning approaches for detecting changepoints in time series. In Chapter 4, we introduce a neural online density-ratio estimator for non-parametric online CPD in Euclidean spaces. In Chapter 5, we consider non-parametric online CPD on manifolds with theoretical analyses and its application to two instances of Riemannian manifolds. In Chapter 6, we extend the algorithm in Chapter 5 to process streaming manifold-valued data over networks.

Part I - Joint Modeling and Learning Approaches: Hyperspectral Imaging

Chapter 2: Considering a joint modeling and learning approach, we address HI Deconvolution by solving an ill-posed inverse problem. we introduce a tuning-free Plug-and-Play algorithm for HSI deconvolution. Specifically, we use the alternating direction method of multipliers (ADMM) to decompose the optimization problem into two iterative sub-problems. A flexible blind 3D denoising network (B3DDN) is designed to learn deep priors and to solve the denoising sub-problem with different noise levels. A measure of 3D residual whiteness is then investigated to adjust the penalty parameters when solving the quadratic sub-problems, as well as a stopping criterion. This work is related to the publication [Wang 2023f].

Chapter 3: HMIF is another typical inverse hyperspectral imaging problem that can be addressed by a joint modeling and learning approach. We present a general imaging model that considers inter-image variability of data from heterogeneous sources and flexible image priors. The fusion problem is stated as an optimization problem in the maximum a posteriori framework. We introduce an original image fusion method that, on the one hand, solves the optimization problem accounting for inter-image variability with an iteratively reweighted scheme and, on the other hand, that leverages lightweight CNN-based networks to learn realistic image priors from data. In addition, we propose a zero-shot strategy to directly learn the image-specific prior of the latent images in an unsupervised manner. This work is related to the publications [Wang 2022a, Wang 2023c].

Part II - Joint Modeling and Learning Approaches: changepoint Detection

Chapter 4: Detecting changepoints in streaming time series data is a long-standing problem in signal processing. Nevertheless, leveraging recent advances in deep learning to detect changepoints in time series data is still challenging. We propose a joint modeling and learning method using an online approach based on neural networks to directly estimate the density ratio between current and reference windows of the data stream. A variational continual learning framework is employed to train the neural network in an online manner while retaining information learned from past data. This leads to a statistically-principled fully nonparametric

framework to detect changepoints from streaming data. This work is related to the publication [Wang 2023b].

Chapter 5: Non-parametric detection of changepoints in streaming time series data that belong to Euclidean spaces has been extensively studied in the literature. Nevertheless, when the data belongs to a Riemannian manifold, existing approaches are no longer applicable as they fail to account for the structure and geometry of the manifold. In this chapter, we introduce a joint modeling and learning algorithm for non-parametric online CPD in manifold-valued data streams. This algorithm monitors the generalized Karcher mean of the data, computed using stochastic Riemannian optimization. We provide theoretical bounds on the detection and false alarm rate performances of the algorithm, using a new result on the non-asymptotic convergence of the stochastic Riemannian gradient descent. In addition, we apply our algorithm to two different manifolds. This work is related to the publications [Wang 2023a, Wang 2024a].

Chapter 6: Signal processing methods over networks have recently been proposed to detect changepoints occurring in localized communities of nodes. Nevertheless, all these methods are mostly limited to time series data in Euclidean spaces. In this chapter, we devise a distributed CPD method for streaming manifold-valued signals over graphs. This framework combines a local test statistic at each node to account for the data geometry residing on a Riemannian manifold, with a fully distributed graph filter that incorporates information on network topology. This significantly improves the detection of changepoints in unknown communities of networks. This work is related to the publications [Wang 2023d, Wang 2023e].

At the end of this thesis, Chapter 7 summarizes our contributions and discusses possible extensions and other open problems for future works, including an exploratory study on Riemannian diffusion adaptation over graphs [Wang 2024b].

1.4 List of publications

The publications related to this thesis are listed below, divided into conference papers (starting with C#) and journal papers (J#). They contain the main works [Wang 2023f, Wang 2022a, Wang 2023c, Wang 2023b, Wang 2023a, Wang 2024a, Wang 2023d, Wang 2023e, Wang 2024b], discussed above, as well as preliminary work published in a conference [Wang 2022b] and a review article [Chen 2022].

[C1] **X. Wang**, R. A. Borsoi, C. Richard, "Non-parametric Online Change Point Detection on Riemannian Manifolds", International Conference on Machine Learning (ICML), Vienna, Austria, Jul. 2024.

[C2] **X. Wang**, R. A. Borsoi, C. Richard, "Riemannian Diffusion Adaptation over Graphs with Application to Online Distributed PCA", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Korea, Apr. 2024.

[C3] **X. Wang**, R. A. Borsoi, C. Richard, André Ferrari, "Distributed Change Point Detection in Streaming Manifold-valued Signals over Graphs", Asilomar Conference on Signals, Systems and Computers (ASILOMAR), Pacific Grove (CA), USA, Oct. 2023.

[C4] **X. Wang**, R. A. Borsoi, C. Richard, André Ferrari, "Détection de changements dans des signaux sur graphe à valeur dans des variétés riemanniennes", Colloque Francophone de Traitement du Signal et des Images (GRETSI), Grenoble, France, Aug. 2023.

[C5] **X. Wang**, R. A. Borsoi, C. Richard, "Online change point detection on Riemannian manifolds with Karcher mean estimates", European Signal Processing Conference (EUSIPCO), Helsinki, Finnland, Sep. 2023.

[C6] **X. Wang**, R. A. Borsoi, C. Richard, J. Chen, "Change Point Detection with Neural Online Density-Ratio Estimator", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, June 2023.

[C7] **X. Wang**, R. A. Borsoi, C. Richard, J. Chen, "Deep image fusion accounting for inter-image variability", Asilomar Conference on Signals, Systems and Computers (ASILOMAR), Pacific Grove (CA), USA, Nov. 2022.

[C8] **X. Wang**, J. Chen, C. Richard "Hyperspectral image super-resolution with deep priors and degradation model inversion", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, May 2022.

[J1] **X. Wang**, R. A. Borsoi, J. Chen, C. Richard, "Deep Hyperspectral and Multispectral Image Fusion with Inter-image Variability", IEEE Transactions on Geoscience and Remote Sensing, 2023.

[J2] **X. Wang**, J. Chen, C. Richard, "Tuning-free plug-and-play hyperspectral image deconvolution with deep priors", IEEE Transactions on Geoscience and Remote Sensing, 2023.

[J3] J. Chen, M. Zhao, **X. Wang**, C. Richard, S. Rahardja, "Integration of physics-based and data-driven models for hyperspectral image unmixing", IEEE Signal Processing Magazine, 2023.

Part I

Joint Modeling and Learning Approaches: Hyperspectral Imaging

Context

As mentioned in the introduction, the inverse problems in hyperspectral imaging are typically ill-posed. This makes properly defining priors and designing regularizers very important to improve performance and enhance the stability of the inversion process. Some important types of prior information used in hyperspectral inverse imaging problems are related to the sparsity or smoothness of the estimated values. For instance, $\ell_p (0 \leqslant p \leqslant 1)$ norms are commonly employed to enhance the sparsity of the recovered image when represented in some appropriate basis, while total variation (TV) and Laplacian regularization are often utilized to promote the smoothness of the estimated values [Boyd 2004].

However, it is a non-trivial task to handcraft a powerful regularizer to take all types of prior information into account. Meanwhile, complex regularizers may introduce extra difficulties in solving optimization problems in (1.5), especially in the case of non-differentiable regularizers such as ℓ_1-norm and TV-norm regularization terms. Compared with optimization methods based on predefined priors, deep learning methods require fewer assumptions on the prior knowledge of the latent solution $\boldsymbol{x}$, and can directly learn the relevant information from training data in an end-to-end way. Nevertheless, these learning-based methods ignore the degradation model in (1.1), though this model has a clear physical interpretation that relates the observed data $\boldsymbol{y}$ and latent $\boldsymbol{x}$. Recently, benefiting from the variable splitting principle, various Plug-and-Play methods [Venkatakrishnan 2013, Romano 2017] have been proposed. They consist of plugging image-denoising modules into optimization modules to solve inverse problems.

We shall now outline the main principles of the Plug-and-Play framework [Venkatakrishnan 2013, Romano 2017]. With the alternating direction method of multipliers (ADMM) [Boyd 2011] or the half quadratic splitting (HQS) method [Geman 1995], the optimization problem (1.5) can be solved iteratively consisting of two key operations:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2 + \frac{\rho}{2}\|\boldsymbol{x} - \hat{\boldsymbol{z}}\|_2^2, \tag{1.7}$$

$$\hat{\boldsymbol{z}} = \texttt{Denoiser}(\hat{\boldsymbol{x}}, \sigma), \tag{1.8}$$

where ρ is the penalty parameter, and $\texttt{Denoiser}(\cdot)$ represents a denoising operator with $\sigma = \sqrt{\eta/\rho}$ the denoising strength. Conversely, this formulation can also implicitly define $\boldsymbol{\Phi}(\cdot)$ when plugging an arbitrary denoising operator.

In this part, we explore a novel idea for two distinct inverse problems in hyperspectral imaging by designing Plug-and-Play methods to integrate the merits of physical modeling and deep learning. On the one hand, based on the ADMM algorithm, we propose a completely turning-free Plug-and-Play framework for hyperspectral image deconvolution (HID) with the design of a blind deep denoiser and residual whiteness (Chapter 2). On the other hand, we consider a more challenging problem, i.e., MHIF with the inter-image variability, and design an unsupervised algorithm based on a more efficient variant of the Plug-and-Play framework (Chapter 3) and zero-shot learning strategy. By considering the degradation model, both algorithms

can directly learn prior information from data without any explicit assumption on the properties of the latent HIs. Note that Chapter 3 considers an more efficient variant of the Plug-and-Play framework but involves more hyperparameters, for the ease of tuning-free mechanism design, Chapter 2 considers the original Plug-and-Play framework.

Chapter 2

Tuning-free Plug-and-Play HID with deep priors

Contents

2.1 Introduction

As mentioned in Chapter 1, due to various physical and hardware limitations, one degradation phenomenon is that observed HIs are usually blurred and corrupted by noise during the acquisition process, leading to degraded performance in subsequent analyses. Thus, it is desirable to restore images by deconvolution (inversion of the degradation process) techniques beforehand.

Multichannel images contain abundant spectral information across neighboring wavelengths, which raises the challenge of accounting for spectral correlations while ensuring spatial consistency compared to ordinary 2D images [Bongard 2011, Sarder 2006]. Traditionally, the deconvolution of multichannel (multispectral) images involves, e.g., Wiener filter [Galatsanos 1989, Hunt 1984], Kalman filter [Tekalp 1990], and regularized least-squares [Galatsanos 1991]. For hyperspectral deconvolution, an adaptive 3D Wiener filter [Gaucel 2006] and a filter-based linear method [Bongard 2011] have been used for astronomic HIs. 2D Fast Fourier Transforms (FFTs) and Fourier-wavelet techniques have been considered in [Thiébaut 2005] and [Neelamani 2004] for HID to benefit from computational efficiency in Fourier and wavelet

domains. In [Song 2019], an online deconvolution algorithm was devised to process HIs sequentially collected by a push-broom device.

Considering that deconvolution problems are usually highly ill-posed, it is strongly desirable to incorporate prior information of images to regularize the solutions. To this end, a computationally efficient algorithm in [Henrot 2012] performs HID subject to positivity constraints while accounting for spatial and spectral correlations. The work in [Chang 2020] investigates both the spatial non-local self-similarity and spectral correlations by employing low-rank tensor priors. Defining proper priors and designing regularizers play a key role in these methods. However, it is not a trivial task to handcraft powerful regularizers, keeping in mind that complex regularizers may also introduce extra difficulties in solving optimization problems. Recently, benefiting from the variable splitting principle, various Plug-and-Play methods have been proposed. This framework allows us to benefit from the merits of deep learning and model-based optimization methods [Chen 2022], and to eliminate the need for expensive network retraining whenever the inverse problem (i.e., the operator $\boldsymbol{H}$) changes [Zhang 2017b]. Applications include magnetic resonance imaging (MRI) reconstruction [Venkatakrishnan 2013, Wei 2020a], 2D image restoration [Brifman 2016, Zhang 2017b, Chen 2020, Zhang 2021] and hyperspectral unmixing [Wang 2020b, Zhao 2021]. Despite its effectiveness, this strategy has not yet been employed in HID problems, though similar difficulties of designing regularizers are encountered there.

Regardless of whether the regularizers are manually designed or implicitly learned as in recent Plug-and-Play algorithms, it is desirable to select the regularization parameters properly to balance the contribution of prior information and observations. Classic parameter estimation methods used with handcrafted regularizers include the discrepancy principle (DP) [Thompson 1991], the L-curve [Hansen 1992, Vogel 1996], the generalized cross-validation (GCV) [Golub 1979, Reeves 1994], and Stein's unbiased risk estimate (SURE) [Stein 1981, Van De Ville 2011]. Recently, the authors of [Song 2016] proposed the maximum curvature criterion and the minimum distance criterion (MDC) on the response surface to estimate the regularization parameters in a non-negative HID problem [Henrot 2012]. The MDC has been extended to HI super-resolution by considering a deep prior regularizer in [Wang 2021]. By defining and maximizing some whiteness measures of residual images, the authors of [Almeida 2013] proposed a 2D image deblurring method with objective criteria for adjusting the regularization parameter as well as the stopping criterion. In [Lanza 2020], an exact residual whiteness principle has been proposed for generalized Tikhonov-regularized 2D image restoration. However, a specific-designed criterion for 3D images, such as HIs, is still missing.

Compared to handcrafted regularizers, implicit regularizers in Plug-and-Play algorithms introduce extra challenges that need to be addressed for devising an automatic regularization parameter estimation strategy. In the Plug-and-Play framework (1.7) and (1.8), η is reparameterized by a series of internal parameters, including the penalty parameter ρ, the denoising strength σ, and the number of iterations K (related to stopping criteria). In the plug-and-play approaches with re-

gularizations based on a denoiser [Brifman 2016, Wang 2020b, Zhao 2021], a constant scaling factor is used to increase ρ linearly as iterations proceed. In [Zhang 2017b], σ is exponentially decayed in sequential denoising sub-problems. Nevertheless, the selected parameters in all these handcrafted criteria may lead to sub-optimal performance since the internal parameters may not change monotonically. To address this issue, the methods in [Chen 2020, Zhang 2021] consist of training a blind denoising network to estimate σ automatically. The work in [Chen 2020] considers a fixed ρ while the approach in [Zhang 2021] considers a fixed η. Unlike these semi-automated approaches, deep reinforcement learning is used in [Wei 2020a] to determine all the internal parameters, leading to good convergence behavior and performance.

This chapter introduces a fully automatic Plug-and-Play hyperspectral deconvolution method that uses spectral-spatial priors learned from data by a deep neural network. The HID problem is addressed with an ADMM algorithm. To avoid manually selecting the regularization parameters, we define a non-negative scalar measure of whiteness for 3D residual images, which cooperates with a blind deep denoiser to adaptively adjust all the internal parameters. The contributions of this work are summarized as follows:

— We propose a Plug-and-Play HID framework. Based on the ADMM algorithm, the optimization problem is split into two sub-problems, a simple quadratic sub-problem and a 3D-image denoising sub-problem.
— A blind deep denoiser referred to as B3DDN is designed and plugged into the proposed framework. This denoising operator learns both spatial context and spectral attributes of HIs, bypassing the difficulty in designing regularizers. After training with simulated data, the flexibility of the B3DDN allows it to represent, without any extra training, the priors for real-world images even with a distinct number of spectral channels.
— The proposed Plug-and-Play framework is designed in a completely turning-free manner. Specifically, the penalty parameters are determined automatically by solving a scalar optimization problem while the denoising strengths are implicitly learned by the B3DDN. A stopping criterion for the iterative process is also provided.
— An HI dataset containing six blurring and clear image pairs captured in indoor and outdoor scenes is provided with this work. This dataset allows us to show that our method applies to real-world scenarios. It also provides a benchmark for future research works in hyperspectral deconvolution.

The chapter is organized as follows. In Section 2.2, HID is formulated as a linear inverse problem. Section 2.3 introduces the proposed tuning-free deconvolution method based on the Plug-and-Play framework with learned deep priors. In Section 2.4, experiments with simulated and real-world data are conducted and analyzed. Section 2.5 concludes this chapter.

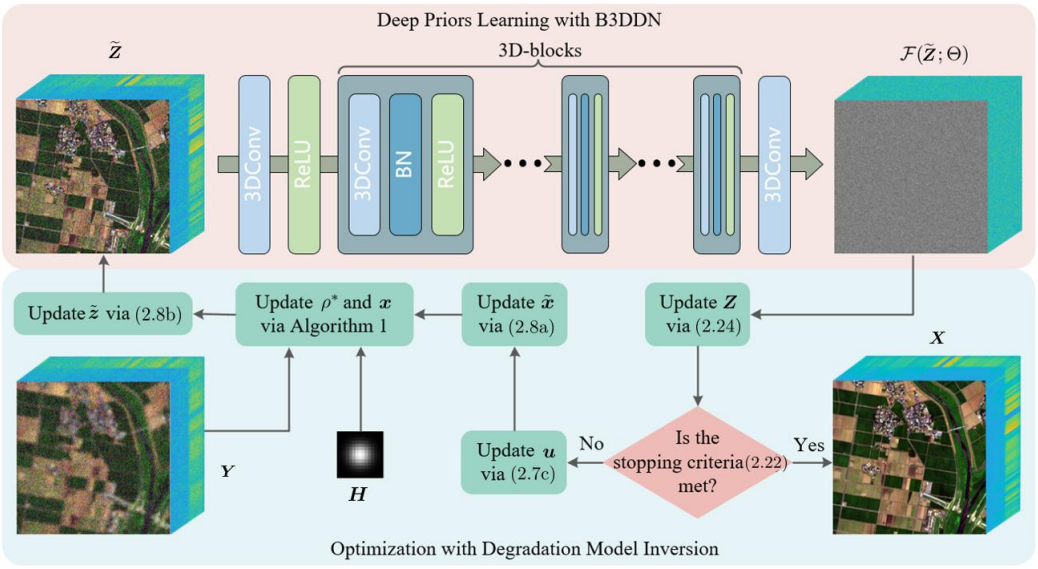

FIGURE 2.1 – Architecture of the proposed tuning-free scheme for HID. **Top panel**: Network structure of the B3DDN. **Bottom panel**: Numerical optimization steps in the ADMM framework.

2.2 Image deconvolution with linear model

We denote a degraded HI and its latent clean counterpart by $\boldsymbol{Y} \in \mathbb{R}^{L\times P\times Q}$ and $\boldsymbol{X} \in \mathbb{R}^{L\times P\times Q}$ respectively, where P, Q, and L are the numbers of rows, columns and spectral bands of the image. Using lexicographical order, $\boldsymbol{Y}$ and $\boldsymbol{X}$ can be reshaped into vectors $\boldsymbol{y} \in \mathbb{R}^{LPQ\times 1}$ and $\boldsymbol{x} \in \mathbb{R}^{LPQ\times 1}$, respectively. The degraded image and the clean image at the i-th spectral band are denoted by $\boldsymbol{Y}_i \in \mathbb{R}^{P\times Q}$ and $\boldsymbol{X}_i \in \mathbb{R}^{P\times Q}$. For ease of mathematical formulation, the columns of $\boldsymbol{Y}_i$ and $\boldsymbol{X}_i$ are stacked to form vectors $\boldsymbol{y}_i \in \mathbb{R}^{N\times 1}$ and $\boldsymbol{x}_i \in \mathbb{R}^{N\times 1}$ with $N \triangleq PQ$ denoting the number of pixels. $\boldsymbol{x}$ and $\boldsymbol{y}$ are vectors obtained by stacking vectors $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ $(1 \leqslant i \leqslant L)$, respectively. This notation system also works for other images.

For the i-th channel, $\boldsymbol{Y}_i$ is generated from $\boldsymbol{X}_i$ according to the following 2D degradation model:

$$\boldsymbol{Y}_i = \mathcal{H}_i * \boldsymbol{X}_i + \boldsymbol{N}_i, \tag{2.1}$$

where $\mathcal{H}_i$ is the convolution kernel, possibly containing null entries, of size $P \times Q$ encoding the Point Spread Function (PSF) of the i-th channel:

$$\mathcal{H}_i = \begin{pmatrix} \mathcal{H}_{1,1} & \cdots & \mathcal{H}_{1,Q} \\ \vdots & \ddots & \vdots \\ \mathcal{H}_{P,1} & \cdots & \mathcal{H}_{P,Q} \end{pmatrix}. \tag{2.2}$$

Operator $*$ denotes the discrete 2D convolution performed in the image domain, and $\boldsymbol{N}_i$ is an additive independent and identically distributed (i.i.d.) Gaussian noise with

standard deviation σ. Following [Henrot 2012], model (2.2) can be written as:

$$\boldsymbol{y}_i = \boldsymbol{H}_i \boldsymbol{x}_i + \boldsymbol{n}_i, \tag{2.3}$$

where $\boldsymbol{H}_i$ is a $N \times N$ block-Toeplitz matrix with $P \times Q$ Toeplitz blocks. Imposing periodic boundary conditions on $\mathcal{H}_i$, $\boldsymbol{H}_i$ can be rewritten as a block circulant matrix with circulant blocks, a structure denoted as circulant-block-circulant (CBC). This property allows us to design a Fourier domain implementation for solving the least square problem in Section 2.3.1.

Assuming that the convolution is separable (i..e, the convolution kernel is invariant across spectral channels) and the noise variance is independent over spectral bands, the hyperspectral degradation model can be written in the form of (1.1) where $\boldsymbol{H}$ is a block-diagonal matrix of size $LN \times LN$:

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{H}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \boldsymbol{H}_N \end{bmatrix}. \tag{2.4}$$

The problem in HID is formulated as an inverse problem, where $\boldsymbol{X}$ is estimated by seeking the minimum of the objective function (1.5).

2.3 Proposed method

Designing an effective regularizer $\Phi(\boldsymbol{x})$ along with an efficient solving method is not trivial. Meanwhile, it is cumbersome to fine-tune the hyperparameter η to balance the contribution of $\Phi(\boldsymbol{x})$ for different images. To tackle these issues, we propose to learn priors from hyperspectral data and incorporate them into the model-based optimization to tackle the regularized inverse problem in (1.5). More specifically, using the variable splitting technique [Boyd 2004], we transform problem (1.5) into two sub-problems, namely, a simple quadratic problem with a penalty parameter and a 3D-image denoising problem with a certain denoising strength. These sub-problems are iteratively solved, using a linear method and a blind deep neural network, respectively, until the convergence criterion is met. In this procedure, the penalty parameter is automatically estimated while the denoising strength is implicitly learned. Finally, the algorithm is automatically terminated by stopping criteria. Our tuning-free HID scheme is illustrated in FIGURE 2.1.

2.3.1 Variable splitting based on the ADMM

The ADMM is adopted to decouple the data fidelity term and the regularization term in (1.5). By introducing an auxiliary variable $\mathbf{z}$, problem (1.5) can be written in the equivalent form:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2 + \eta\Phi(\boldsymbol{z}), \quad \text{s.t.} \quad \boldsymbol{z} = \boldsymbol{x}. \tag{2.5}$$

The associated augmented Lagrangian function [Boyd 2004] is given by

$$\mathcal{L}_\rho(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{v}) = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2 + \eta\Phi(\boldsymbol{z}) + \boldsymbol{v}^T(\boldsymbol{x} - \boldsymbol{z}) + \frac{\rho}{2}\|\boldsymbol{x} - \boldsymbol{z}\|^2, \tag{2.6}$$

with $\boldsymbol{v}$ the dual variable, and $\rho > 0$ the penalty parameter. Scaling $\boldsymbol{v}$ as $\boldsymbol{u} = \rho^{-1}\boldsymbol{v}$, problem (2.6) can be iteratively solved by repeating the following successive steps:

$$\boldsymbol{x}_{k+1} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|^2 + \frac{\rho_k}{2}\|\boldsymbol{x} - \tilde{\boldsymbol{x}}_k\|^2, \tag{2.7a}$$

$$\boldsymbol{z}_{k+1} = \arg\min_{\boldsymbol{z}} \eta\Phi(\boldsymbol{z}) + \frac{\rho_k}{2}\|\tilde{\boldsymbol{z}}_k - \boldsymbol{z}\|^2, \tag{2.7b}$$

$$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \boldsymbol{x}_{k+1} - \boldsymbol{z}_{k+1}, \tag{2.7c}$$

where

$$\tilde{\boldsymbol{x}}_k = \boldsymbol{z}_k - \boldsymbol{u}_k, \tag{2.8a}$$

$$\tilde{\boldsymbol{z}}_k = \boldsymbol{x}_{k+1} + \boldsymbol{u}_k, \tag{2.8b}$$

and ρ_k denotes the penalty parameter at the k-th iteration. In this way, the data fidelity term and the regularization term in (1.5) are decoupled into two sub-problems, (2.7a) and (2.7b). Sub-problem (2.7a) is a least square problem that can be solved analytically as follows:

$$\boldsymbol{x}_{k+1} = (\boldsymbol{H}^T\boldsymbol{H} + \rho_k\mathbf{I})^{-1}(\boldsymbol{H}^T\boldsymbol{y} + \rho_k\tilde{\boldsymbol{x}}_k), \tag{2.9}$$

Subproblem (2.7b) can be reformulated as:

$$\boldsymbol{z}_{k+1} = \arg\min_{\boldsymbol{z}} \frac{1}{2\sigma_k^2}\|\tilde{\boldsymbol{z}}_k - \boldsymbol{z}\|^2 + \Phi(\boldsymbol{z}), \tag{2.10}$$

where $\sigma_k = \sqrt{\eta/\rho_k}$.

From a Bayesian perspective [1], (2.10) can be considered as a denoising problem, removing Gaussian noise with noise-level σ_k from the noisy HI $\tilde{\boldsymbol{z}}_k$ to obtain the clean HI $\boldsymbol{z}_{k+1}$. In other words, a denoising operator can be used for implicitly designing the regularization term $\Phi(\boldsymbol{x})$.

2.3.2 Estimating parameters via 3D residual whiteness

In most real-world applications, no ground-truth information is available for fine-tuning the algorithm parameters or terminating the optimization at a proper

1. Considering a degradation model $\tilde{\boldsymbol{z}}_k = \boldsymbol{z} + \boldsymbol{n}_k$ where $\boldsymbol{n}_k$ is Gaussian noise with standard deviation σ_k. The denoising problem can be formulated as the recovery of the posterior probability density function (PDF) $p(\boldsymbol{z}|\tilde{\boldsymbol{z}}_k)$. Using the Bayes theorem, this PDF can be written as: $p(\boldsymbol{z}|\tilde{\boldsymbol{z}}_k) \propto p(\tilde{\boldsymbol{z}}_k|\boldsymbol{z})p(\boldsymbol{z})$ where $p(\boldsymbol{z})$ is the prior probability distribution of $\boldsymbol{z}$. Finally, the log-posterior distribution can be written as $-\log p(\boldsymbol{z}|\tilde{\boldsymbol{z}}_k) = \frac{1}{2\sigma_k^2}\|\tilde{\boldsymbol{z}}_k - \boldsymbol{z}\|^2 + \log p(\boldsymbol{z}) + C$ where C is a constant. By rewriting $\log p(\boldsymbol{z})$ as $\Phi(\boldsymbol{z})$, estimating $\boldsymbol{z}$ in the sense of the maximum a posterior principle leads to the optimization problem in (2.10).

iteration. To tackle this issue, a measure of residual whiteness of 3D images is defined in this subsection, and the optimal value of ρ_k at each iteration, as well as the number of iterations, can be determined with the help of this measure. To be specific, we propose to evaluate the optimal ρ_k^* in (2.7a) by solving a scalar optimization problem. The stopping criterion then consists of comparing this 3D whiteness measure between two iterations.

2.3.2.1 Measure of 3D residual whiteness

We define the residual image $\boldsymbol{r}_{k+1} \in \mathbb{R}^{LN}$ by:

$$\boldsymbol{r}_{k+1} = \boldsymbol{H}\boldsymbol{x}_{k+1} - \boldsymbol{y}, \tag{2.11}$$

with its equivalent 3D image cube denoted by $\boldsymbol{R}_{k+1} \in \mathbb{R}^{N\times P\times Q}$. The *auto-correlation* of $\boldsymbol{R}_{k+1}$ is defined as:

$$\boldsymbol{A}_{\boldsymbol{R}_{k+1}} = \frac{1}{LN}(\boldsymbol{R}_{k+1} \star \boldsymbol{R}_{k+1}), \tag{2.12}$$

where $\star$ denotes the 3D discrete correlation. The sample auto-correlation at indexes (l, p, q) is given by:

$$\boldsymbol{A}_{\boldsymbol{R}_{k+1}}(l,p,q) = \frac{1}{LN}\sum_{m,i,j} \boldsymbol{R}_{k+1}(l,p,q)\boldsymbol{R}_{k+1}(m+l, i+p, j+q), \tag{2.13}$$

with $1 \leqslant m \leqslant L$, $1 \leqslant i \leqslant P$, $1 \leqslant j \leqslant Q$. When the residual is close to the modeling error $\mathbf{l}$, i.e., a white Gaussian noise, $\boldsymbol{A}_{\boldsymbol{R}_{k+1}}(l,p,q)$ satisfies the following asymptotic property:

$$\lim_{LN\to\infty} \boldsymbol{A}_{\boldsymbol{R}_{k+1}}(l,p,q) \approx \begin{cases} \sigma^2 & \text{if } (l,p,q) = (0,0,0), \\ 0 & \text{if } (l,p,q) \neq (0,0,0). \end{cases} \tag{2.14}$$

The size LN of hyperspectral images is usually large (between 10^6 and 10^8), so that we can assume that the sample auto-correlation at all indexes $(l, p, q) \neq (0, 0, 0)$ is close to zero. This assumption is based on the following result of the Gaussian process $\boldsymbol{n}$ with its equivalent 3D image matrix denoted by $\boldsymbol{N} \in \mathbb{R}^{L\times P\times Q}$ and sample auto-correlation $\boldsymbol{A}_{\boldsymbol{N}_{k+1}}(l,p,q)$ defined by replacing $\boldsymbol{R}$ as $\boldsymbol{N}$ in (2.13).

Theorem 1. *If $\boldsymbol{n}$ has a finite variance σ and LN tends to ∞, any $\boldsymbol{A}_{\boldsymbol{N}_{k+1}}(l,p,q)$ with $(l,p,q) \neq (0,0,0)$ are asymptotically uncorrelated and their limiting distribution is a Gaussian distribution with zero mean and standard deviation $\sigma_a = \sqrt{\frac{\sigma^2}{LN}} \to 0$.*

Proof. The proof follows directly by applying Proposition 1 of [Lanza 2018] to the 3D domain.

The rationale behind imposing residual whiteness is to estimate parameters by constraining the residual auto-correlation at non-zero indexes to be small. To make this measure independent from σ, inspired by [Lanza 2020], we consider the *normalized auto-correlation* defined as follows:

$$\overline{\boldsymbol{A}}_{\boldsymbol{R}_{k+1}} = \frac{\boldsymbol{A}_{\boldsymbol{R}_{k+1}}}{\boldsymbol{A}_{\boldsymbol{R}_{k+1}}(0,0,0)} = \frac{\boldsymbol{R}_{k+1} \star \boldsymbol{R}_{k+1}}{\|\boldsymbol{R}_{k+1}\|_F^2}, \tag{2.15}$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm. All entries $\overline{\boldsymbol{A}}_{\mathbf{rr}}(l,p,q)$ satisfies:

$$\lim_{LN\to\infty} \overline{\boldsymbol{A}}_{\boldsymbol{R}_{k+1}}(l,p,q) \approx \begin{cases} 1 & \text{if } (l,p,q)=(0,0,0), \\ 0 & \text{if } (l,p,q)\neq(0,0,0). \end{cases} \tag{2.16}$$

We can now introduce the σ-independent non-negative scalar measure of 3D residual whiteness defined as:

$$\mathcal{W}(\boldsymbol{R}_{k+1}) = ||\overline{\boldsymbol{A}}_{\boldsymbol{R}_{k+1}}||_F^2 = \frac{||\boldsymbol{R}_{k+1} \star \boldsymbol{R}_{k+1}||_F^2}{||\boldsymbol{R}_{k+1}||_F^4}. \tag{2.17}$$

2.3.2.2 Penalty parameter estimation

Solution $\boldsymbol{x}_{k+1}$ of (2.9) actually depends on the setting of parameter ρ_k. To devise the parameter selection procedure, we make ρ_k explicit by writing $\boldsymbol{x}_{k+1,\rho_k}$. In order to automatically estimate the penalty parameter ρ_k in (2.7a), the term $\|\boldsymbol{x}-\tilde{\boldsymbol{x}}_k\|^2$ can be viewed as a regularizer that enforces the solution $\boldsymbol{x}_{k+1,\rho_k}$ to tend to $\tilde{\boldsymbol{x}}_k$. As the restored image $\boldsymbol{x}_{k+1,\rho_k}$ tends to fit the desired target image, the related residual image $\boldsymbol{r}_{k+1,\rho_k} = \boldsymbol{H}\boldsymbol{x}_{k+1,\rho_k} - \boldsymbol{y}$ tends to be close to the Gaussian noise perturbation $\boldsymbol{n}$ in (1.1). With (2.17), we propose to estimate the optimal penalty parameter by solving the following scalar optimization problem:

$$\rho_k^* = \arg\min_{\rho_k} \mathcal{W}(\boldsymbol{r}_{k+1,\rho_k}). \tag{2.18}$$

The varying range of ρ_k is $(0,\infty)$. In practice, we substitute the ∞ by a sufficiently large value.

A fast golden-section search method is used for determining a local minimum of (2.18). This method operates iteratively over an interval (a,b) and generates two internal points:

$$\begin{aligned} \rho_k^{(1)} &= a + \delta(b-a), \\ \rho_k^{(2)} &= b - \delta(b-a). \end{aligned} \tag{2.19}$$

where $\delta = 0.618$ is the golden ratio. As shown in Algorithm 1, whiteness criterion $\mathcal{W}(\boldsymbol{r}_{k+1,\rho_k})$ is compared at $\rho_k^{(1)}$ and $\rho_k^{(2)}$. If it is smaller at the former point than at the latter point, then b is substituted by $\rho_k^{(2)}$. Otherwise, a is substituted by $\rho_k^{(1)}$. This procedure is repeated with the new smaller interval (a,b) until $b-a<\varepsilon$ with ε a small positive threshold. Finally, the estimated optimal penalty parameter is given by:

$$\rho_k^* = (a+b)/2, \tag{2.20}$$

and the solution of sub-problem (2.7a) is provided by:

$$\boldsymbol{x}_{k+1} = (\boldsymbol{H}^T\boldsymbol{H} + \rho_k^*\mathbf{I})^{-1}(\boldsymbol{H}^T\boldsymbol{y} + \rho_k^*\tilde{\boldsymbol{x}}_k). \tag{2.21}$$

Algorithm 1 Adaptive Penalty Parameter Estimation.

Input : Blurred observation $\mathbf{y}$, internal image $\tilde{\boldsymbol{x}}_k$, blurring matrix $\boldsymbol{H}$.
Output : Optimal adaptive parameter ρ_k^*.
Initialize a, b, ε.
while $b - a > \varepsilon$ **do**
$\rho_k^{(1)} = a + \delta(b-a)$,
$\rho_k^{(2)} = b - \delta(b-a)$,
if $\mathcal{W}(\boldsymbol{r}_{k+1,\rho_k^{(1)}}) < \mathcal{W}(\boldsymbol{r}_{k+1,\rho_k^{(2)}})$
$b = \rho_k^{(2)}$,
else
$a = \rho_k^{(1)}$,
$\rho_k^* = (a+b)/2$.

2.3.2.3 Stopping criterion

To take both HID performance and computational time into account, it is important to properly set the maximum number of iterations. Iterations can be performed until no significant improvement between two consecutive iterations is observed. Considering the whiteness measure in (2.17), we propose to stop the iterative process with the following normalized criterion:

$$\mathcal{W}(\boldsymbol{r}_{k+1}) \geqslant \mathcal{W}(\boldsymbol{r}_k) \quad \text{or} \quad \frac{||\mathcal{W}(\boldsymbol{r}_{k+1}) - \mathcal{W}(\boldsymbol{r}_k)||}{\mathcal{W}(\boldsymbol{r}_{k+1})} < \zeta, \tag{2.22}$$

where ζ is a small positive threshold, $\boldsymbol{r}_k$ and $\boldsymbol{r}_{k+1}$ represent the residual image of the solutions $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k+1}$, respectively.

2.3.3 Learning spectral-spatial priors via B3DDN

Instead of using a handcrafted regularizer $\Phi(\cdot)$ and solving subproblem (2.7b) explicitly, we propose to carry out this task with a deep neural network based denoiser. This denoiser is trained beforehand to extract spectral-spatial prior information from hyperspectral training observations. Then it is plugged into the iterative algorithm to solve subproblem (2.7b). We denote this denoising operator by $\mathcal{D}(\cdot)$. As it is performed in the 3D image domain to jointly capture spatial and spectral information, we write (2.10) as follows:

$$\boldsymbol{Z}_{k+1} = \mathcal{D}(\widetilde{\boldsymbol{Z}}_k, \sigma_k), \tag{2.23}$$

Observe that $\mathcal{D}(\cdot)$ is parameterized by the noise level σ_k. For setting it, most existing methods use empirical strategies that may lead to under-denoising or over-smoothing of $\widetilde{\boldsymbol{Z}}_k$ [Chen 2020]. In addition, since σ_k decreases as iterations progress, some works choose to train a set of specific models that can handle different noise levels [Zhang 2017b]. To avoid these redundant learning tasks, we shall now see how

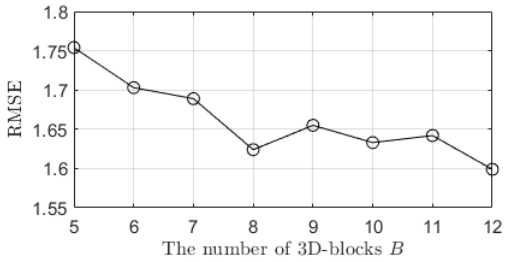

FIGURE 2.2 – The denoising performance of B3DDN with different B.

to design a blind 3D denoising network $\mathcal{F}(\cdot)$ w.r.t. a range of σ_k, denoted as Σ and parameterized by $\Theta(\Sigma)$, by considering residual learning reformulation of (2.23) as:

$$\boldsymbol{Z}_{k+1} = \widetilde{\boldsymbol{Z}}_k - \mathcal{F}(\widetilde{\boldsymbol{Z}}_k; \Theta(\Sigma)). \tag{2.24}$$

2.3.3.1 3D convolution

Unlike 2D convolution resulting in spectral information distortion, 3D convolution extracts spatial features from neighboring pixels and spectral features from adjacent bands, simultaneously, without compromising spectral resolution. 3D convolution also involves fewer parameters, and it is more appropriate for hyperspectral image processing due to the difficulty in capturing a large enough volume of hyperspectral data. In addition, 3D convolution enables the neural network to handle HIs with an arbitrary number of spectral bands without modifying its architecture [Liu 2019b]. In this way, there is no need to retrain a neural network when the number of spectral bands changes. This key property allows our method to be applied to any real-world dataset using a pre-trained neural network.

2.3.3.2 Network architecture

The B3DDN architecture is a non-trivial extension from [Zhang 2017a] to the 3D image domain and is illustrated in FIGURE 2.1 (top). Each 3D-block contains a 3D convolution layer (3DConv), a batch normalization (BN) layer, and a ReLU layer. Batch normalization is used to speed up the training process as well as to boost the denoising performance [Zhang 2017a]. Besides the input layer and the output layer, a 3D convolution layer (3DConv), a ReLU activation function layer, B 3D-blocks and a last 3D convolution layer are sequentially connected to form the proposed network. The last convolutional layer contains one 3D-filter while the others are composed of 32 3D-filters. The kernel size of each 3D-filter is $3\times3\times3$, which means that the depth of the kernel along the spectral dimension and its size over the spatial dimension are 3 and 3×3 respectively. Compared to existing complex network architectures for HI denoising, B3DDN achieves satisfactory performance with fewer parameters. Moreover, it enables us to apply the neural network learned with simulated data, to real data that lacks ground truth. An example is provided in subsection 2.4.4.

Algorithm 2 Tuning-free HID with deep priors learned from B3DDN.

Input : Network parameters $\Theta(\Sigma)$, blurred observation $\boldsymbol{y}$, blurring kernel $\boldsymbol{H}$.
Output : Deblurred HI $\boldsymbol{x}$.
Initialize $\boldsymbol{x} = \boldsymbol{x}_0$, auxiliary variable $\boldsymbol{z}_0 = \boldsymbol{x}_0$,
scaled dual variable $\boldsymbol{u}_0 = 0$, $k = 0$.
while Stopping criteria in (2.22) are not met **do**
$\tilde{\boldsymbol{x}}_k = \boldsymbol{z}_k - \boldsymbol{u}_k$,
Estimate ρ_k^* using Algorithm 1,
$\boldsymbol{x}_{k+1} = (\boldsymbol{H}^T\boldsymbol{H} + \mu\mathbf{I})^{-1}(\boldsymbol{H}^T\boldsymbol{y} + \rho_k^*\tilde{\boldsymbol{x}}_k)$,
$\tilde{\boldsymbol{z}}_k = \boldsymbol{x}_{k+1} + \boldsymbol{u}_k$,
$\boldsymbol{z}_{k+1} = \tilde{\boldsymbol{Z}}_k - \mathcal{F}(\tilde{\boldsymbol{Z}}_k; \Theta(\Sigma))$,
$\boldsymbol{u}_{k+1} = \boldsymbol{u}_k + \boldsymbol{x}_{k+1} - \boldsymbol{z}_{k+1}$,
$k = k + 1$.

The number of 3D-blocks B, and the number and size of 3D filters are fine-tuned according to empirical performance. For example, the influence of the number of 3D-blocks B is examined in FIGURE. 2.2, where the average root-mean-square error (RMSE) value based on the test set in the CAVE dataset is used to evaluate the denoising performance. As shown in FIGURE. 2.2, large values of B generally lead to better results. In our deconvolution experiment, we set $B = 8$ as a larger value by considering the computational cost and memory demand.

2.3.3.3 Learning strategy

The input of the proposed B3DDN is a noisy hyperspectral image $\tilde{\boldsymbol{z}} = \boldsymbol{z} + \boldsymbol{n}$, where $\boldsymbol{n}$ is a Gaussian noise with arbitrary standard deviation. Inspired by 2D image denoising algorithm [Zhang 2017a], we consider the learning residual to predict the residual error $\mathcal{F}(\tilde{\boldsymbol{z}}_k; \Theta(\Sigma)) \approx \boldsymbol{n}$ in our denoising network. Then we can achieve the estimated clean image by $\tilde{\boldsymbol{z}} - \mathcal{F}(\tilde{\boldsymbol{z}}; \Theta(\Sigma))$. To train the blind neural network $\mathcal{F}(\,\cdot\,; \Theta(\Sigma))$, we use the following loss function:

$$\ell(\Theta(\Sigma)) = \|\mathcal{F}(\tilde{\boldsymbol{z}}_m; \Theta(\Sigma)) - (\tilde{\boldsymbol{z}}_m - \boldsymbol{z}_m)\|_1, \tag{2.25}$$

where $\{(\tilde{\boldsymbol{z}}_m, \boldsymbol{z}_m)\}_{m=1}^M$ is a training set of generated noisy-clean HI (patch) pairs with various noise levels. Note that the ℓ_1-norm is used as a loss that is more robust to noise than the ℓ_2-norm, found to provide better performance in image restoration in the literature [Zhao 2016, Wang 2021]. After the B3DDN has been trained, it is incorporated into the ADMM framework as a blind denoiser, yielding Algorithm 2.

2.4 Experiments

In this section, we shall conduct experiments of HID on both simulated and real-world datasets to validate our method. The results provided by the proposed

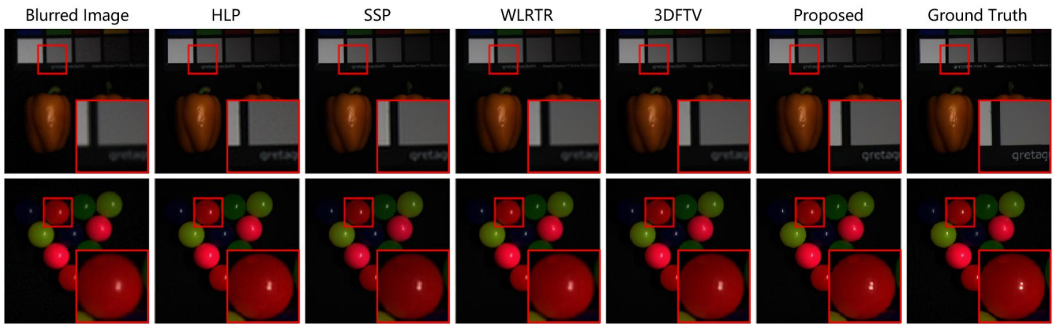

FIGURE 2.3 – Visual results for all methods in the blurring scenario (a) on the CAVE dataset. The first and second rows present the results for two different blurred images. The false color images were generated for clear visualization with the 22nd, 14th and 7th channels used for red, green and blue, respectively.

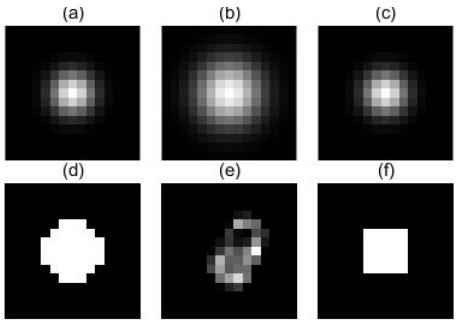

FIGURE 2.4 – Blurring kernels used for the experiments: (a)-(c) are Gaussian kernels, (d)-(f) are circle, motion and square kernels respectively.

method are compared with those of several HID methods from both quantitative and qualitative perspectives. The source code and the proposed real-world data are made available at `https://github.com/xiuheng-wang/Tuning_free_PnP_HSI_deconvolution`.

2.4.1 Simulation datasets and experimental setup

Two simulation datasets, on the one hand the Columbia Multispectral Database (CAVE) [2] [Yasuma 2010], and on the other hand a remotely sensed hyperspectral data over Chikusei [3] [Yokoya 2016], were used to evaluate the performance of our method.

2.4.1.1 CAVE dataset

The CAVE dataset contains 32 HIs recorded under controlled illuminations in a laboratory. Each image has a spatial resolution of 512×512 pixels, over 31 spectral

2. https://www1.cs.columbia.edu/CAVE/databases/multispectral/
3. http://naotoyokoya.com/Download.html

TABLE 2.1 – RMSE, PSNR, SSIM and ERGAS of the different methods applied to the CAVE dataset in the 6 blurring scenarios.

Scenarios	Metrics	HLP	SSP	WLRTR	3DFTV	Proposed
(a)	RMSE	4.420	4.848	4.735	4.332	**3.132**
	PSNR	36.166	35.373	35.872	36.450	**39.252**
	SSIM	0.9167	0.9305	0.9380	0.9401	**0.9493**
	ERGAS	18.15	19.51	18.96	17.34	**13.01**
(b)	RMSE	5.707	5.955	6.439	5.667	**4.581**
	PSNR	34.034	33.541	33.084	34.116	**36.305**
	SSIM	0.8911	0.9031	0.9025	0.9136	**0.9234**
	ERGAS	22.92	23.71	25.46	22.40	**18.54**
(c)	RMSE	7.669	5.270	5.099	5.016	**4.225**
	PSNR	30.599	34.309	34.827	34.741	**36.211**
	SSIM	0.6406	0.8565	**0.8956**	0.8851	0.8708
	ERGAS	33.49	22.28	20.80	20.47	**18.64**
(d)	RMSE	4.189	4.584	4.328	4.167	**2.305**
	PSNR	36.548	35.862	36.686	36.805	**41.653**
	SSIM	0.9165	0.9354	0.9450	0.9403	**0.9542**
	ERGAS	17.36	18.49	17.45	16.69	**9.86**
(e)	RMSE	3.759	3.954	4.335	3.587	**3.041**
	PSNR	37.149	37.160	36.497	37.991	**40.722**
	SSIM	0.9118	**0.9472**	0.9428	0.9510	0.8907
	ERGAS	15.94	16.01	17.46	14.37	**15.56**
(f)	RMSE	3.971	4.356	4.109	3.957	**2.280**
	PSNR	36.910	36.322	37.130	37.225	**41.932**
	SSIM	0.9195	0.9397	0.9480	0.9468	**0.9475**
	ERGAS	16.58	17.60	16.64	15.89	**9.79**

The best results are indicated by boldface numbers.

channels ranging from 400 nm to 700 nm at a wavelength interval of 10 nm.

2.4.1.2 Chikusei dataset

The Chikusei dataset is an airborne hyperspectral scene acquired by a Visible and Near-Infrared imaging sensor over agricultural and urban regions in Chikusei, Ibaraki, Japan. The scene consists of 2517×2335 pixels with a ground sampling distance of 2.5 m, over 128 spectral channels ranging from 363 nm to 1018 nm. The black boundaries in the spatial domain were removed, leading to a scene of size 2048×2048 pixels.

The HIs of the two datasets were scaled to the range $[0, 1]$, and then used as ground truths for $\boldsymbol{x}$. The observations $\boldsymbol{y}$ were generated by using the blurring kernels $\boldsymbol{H}$ and corrupted with a white Gaussian noise $\boldsymbol{n}$ with standard deviation σ, with $\boldsymbol{H}$ and σ defined as follows; see FIGURE 2.4:

(a) 9×9 Gaussian kernel with bandwidth 2, and $\sigma = 0.01$;
(b) 13×13 Gaussian kernel with bandwidth 3, and $\sigma = 0.01$;
(c) 9×9 Gaussian kernel with bandwidth 2, and $\sigma = 0.03$;

(d) Circle kernel with diameter 7, and $\sigma = 0.01$;
(e) Motion kernel from [Levin 2009] of size 13×13, and $\sigma = 0.01$;
(f) Square kernel with side length 5, and $\sigma = 0.01$.

Note these kernels are used in the steps (2.21), but not in the training of B3DDN. Following the previous learning-based methods for hyperspectral imaging [Wang 2021, Xie 2020], the first 20 images were selected from the CAVE dataset for training and the remaining 12 images were used for the test. For the Chikusei dataset, a 1024×2048 sub-image was extracted from the top area of the image for training while the remaining part was cropped into 32 non-overlapping $256 \times 256 \times 128$ sub-images that were used as test data.

2.4.1.3 Implementation details

We implemented the proposed blind denoising network B3DDN with PyTorch framework. The Adam optimizer [Kingma 2014a] with an initial learning rate 0.0002 and batch size 64 was used to minimize the loss function (2.25) with 500 epochs. The weights were initialized by the method in [He 2015]. At every epoch of the training stage, each original HI was randomly cropped into 128 and 512 patches of size 64×64 respectively for the CAVE and the Chikusei datasets, and each patch was then randomly rotated or flipped once for data augmentation purposes. We used empirical methods to determine the learning rate, batch size, and parameters used in data augmentation. To train the B3DDN blindly, we added an i.i.d. Gaussian noise with random standard deviation in the range $\Sigma = [0.2, 10]$ to each patch.

Once the denoiser was trained, assuming that the statistics of the test images differed from the training images, we plugged the B3DDN into the ADMM. Since the computational complexity of 3D discrete correlation in (2.17) can be high ($\mathcal{O}(L^2N^2)$), we used the fast Fourier transform ($\mathcal{O}(LN \log(LN))$) to compute it. Step (2.21) was also efficiently computed in the Fourier domain. For the golden-section search method and the stopping criterion presented in Subsection 2.3.2, we set $a = 0$, $b = 10$, $\varepsilon = 0.001$ and $\zeta = 0.0002$. Note that the performance of our algorithm is not sensitive to the settings of Σ, a, b, ε, and ζ, as these parameters only pertain to the approximate range and accuracy of σ_k, ρ_k, and K. The parameters σ_k, ρ_k, and K are either determined by a measure of 3D residual whiteness or implicitly learned by the B3DDN.

2.4.2 Quantitative metrics and baselines

In order to evaluate the quality of the deconvolution result $\widehat{\boldsymbol{X}}$ by comparing it with the ground truth of $\boldsymbol{X}$, we considered four quantitative metrics. The first one is the Root Mean-Square Error (RMSE), defined as

$$\text{RMSE} = \sqrt{\frac{1}{LPQ}\sum_{i=1}^{L}\|\widehat{\boldsymbol{X}}_i - \boldsymbol{X}_i\|_F^2},$$

which measures the similarities between the deconvolution image and the reference image. A lower RMSE value indicates better quality. The second metric is the Peak-Signal-to-Noise-Ratio (PSNR):

$$\text{PSNR} = \frac{1}{L}\sum_{i=1}^{L} 10\log_{10}\left(\frac{L\,\max(\boldsymbol{X}_i)^2}{\|\widehat{\boldsymbol{X}}_i - \boldsymbol{X}_i\|_F^2}\right),$$

which measures the quality of the deconvolution image compared to the original image. The higher the PSNR, the better the quality. The third metric is the average of Structural SIMilarity (SSIM) [Wang 2004], averaged over all channels of $\widehat{\boldsymbol{X}}$ and $\boldsymbol{X}$, i.e.,

$$\text{SSIM} = \frac{1}{L}\sum_{i=1}^{L} \frac{(2\mu_{\widehat{\boldsymbol{X}}_i}\mu_{\boldsymbol{X}_i} + C_1)(2\sigma_{\widehat{\boldsymbol{X}}_i\boldsymbol{X}_i} + C_2)}{(\mu_{\widehat{\boldsymbol{X}}_i} + \mu_{\boldsymbol{X}_i} + C_1)(\sigma_{\widehat{\boldsymbol{X}}_i} + \sigma_{\boldsymbol{X}_i} + C_2)},$$

where $\mu_{\widehat{\boldsymbol{X}}_i}$ and $\mu_{\boldsymbol{X}_i}$ are the mean values of images $\widehat{\boldsymbol{X}}_i$ and $\boldsymbol{X}_i$, $\sigma_{\widehat{\boldsymbol{X}}_i}$ and $\sigma_{\boldsymbol{X}_i}$ are the standard deviations of $\widehat{\boldsymbol{X}}_i$ and $\boldsymbol{X}_i$, $\sigma_{\widehat{\boldsymbol{X}}_i\boldsymbol{X}_i}$ is the covariance of $\widehat{\boldsymbol{X}}_i$ and $\boldsymbol{X}_i$, and $C_1 > 0$ and $C_2 > 0$ are constants. The SSIM is an indicator of the spatial structure preservation of the deconvolution image. A higher SSIM value indicates better spatial structure preservation. The last metric is the Erreur Relative Globale Adimensionnelle de Synthèse (ERGAS) [Wald 2000] defined as

$$\text{ERGAS} = 100\sqrt{\frac{1}{L}\sum_{i=1}^{L}\frac{\|\widehat{\boldsymbol{X}}_i - \boldsymbol{X}_i\|_F^2}{\text{mean}(\widehat{\boldsymbol{X}}_i)^2}},$$

which characterizes the overall quality of the deconvolution image. A smaller ERGAS means a better result.

We compared our method with three HID methods of reference: hyper-laplacian priors (HLP) [Krishnan 2009a], spatial and spectral priors (SSP) [Henrot 2012], weighted low-rank tensor recovery (WLRTR) [Chang 2020], 3D fractional total variation (3DFTV) [Guo 2021], each with well-designed regularizers. The HLP considers spatial gradient priors, i.e., the hyper-Laplacian priors of images. The SSP exploits both the spatial and spectral smoothness priors of hyperspectral images. The WLRTR simultaneously captures non-local similarity within spectral-spatial cubic and spectral correlation by a low-rank tensor recovery model. The 3DFTV exploits both the local and non-local smoothness of images in all dimensions. We used the codes provided by the authors of these methods and downloaded them, and we tuned their parameters by following the rules as stated in the corresponding papers to achieve the best deconvolution performance.

2.4.3 Performance evaluation on simulated data

We start validating the tuning-free scheme with the CAVE dataset by demonstrating its effectiveness in terms of HID performance over the other methods.

TABLE 2.2 – RMSE, PSNR, SSIM and ERGAS of the different methods applied to the Chikusei dataset in the 6 blurring scenarios.

Scenarios	Metrics	HLP	SSP	WLRTR	3DFTV	Proposed
(a)	RMSE	3.233	3.050	3.138	3.207	**2.560**
	PSNR	38.979	40.182	40.051	39.546	**41.032**
	SSIM	0.9124	0.9334	0.9267	0.9171	**0.9420**
	ERGAS	32.25	28.13	**25.29**	35.37	27.87
(b)	RMSE	3.945	3.819	4.091	4.037	**3.428**
	PSNR	37.604	38.392	37.872	37.708	**38.989**
	SSIM	0.8822	0.9016	0.8871	0.8819	**0.9091**
	ERGAS	35.30	32.40	31.45	39.85	**30.92**
(c)	RMSE	7.094	3.506	3.777	3.662	**3.413**
	PSNR	31.391	**37.942**	37.447	37.756	37.934
	SSIM	0.6268	0.8839	0.8816	**0.8841**	0.8783
	ERGAS	90.14	50.26	**39.95**	48.15	51.38
(d)	RMSE	3.361	2.879	2.890	3.076	**2.335**
	PSNR	39.122	40.625	40.724	39.900	**41.290**
	SSIM	0.9148	0.9399	0.9364	0.9228	**0.9430**
	ERGAS	32.76	27.22	**23.73**	34.59	32.56
(e)	RMSE	2.960	2.436	2.790	2.797	**1.995**
	PSNR	39.127	41.869	41.025	40.574	**42.207**
	SSIM	0.9147	**0.9558**	0.9408	0.9338	0.9507
	ERGAS	35.79	25.42	**23.09**	33.56	36.06
(f)	RMSE	2.990	2.688	2.691	2.913	**2.148**
	PSNR	39.352	41.174	41.313	40.334	**41.971**
	SSIM	0.9188 3	0.9456	0.9438	0.9295	**0.9506**
	ERGAS	32.68	26.19	**22.46**	33.74	30.62

The best performance results are indicated by boldface numbers.

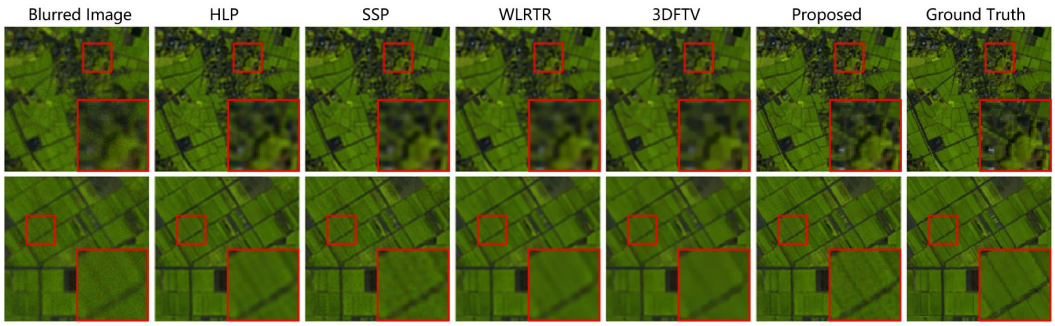

FIGURE 2.5 – Visual results for all methods with the blurring scenario (d) applied to the Chikusei dataset. The first and second rows present the results for two different images. The false color images were generated for clear visualization with the 122nd, 84th and 57th channels used for red, green and blue, respectively.

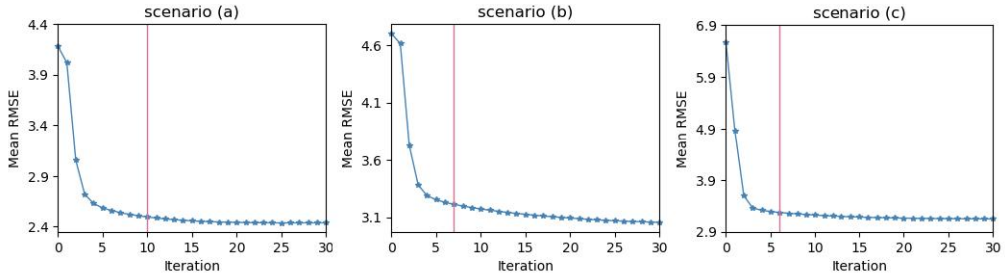

FIGURE 2.6 – RMSE convergence mean curves (blue) of our method with the CAVE dataset and blurring scenarios (a), (b) and (c). Red lines represent the iteration number given by the proposed stopping criterion.

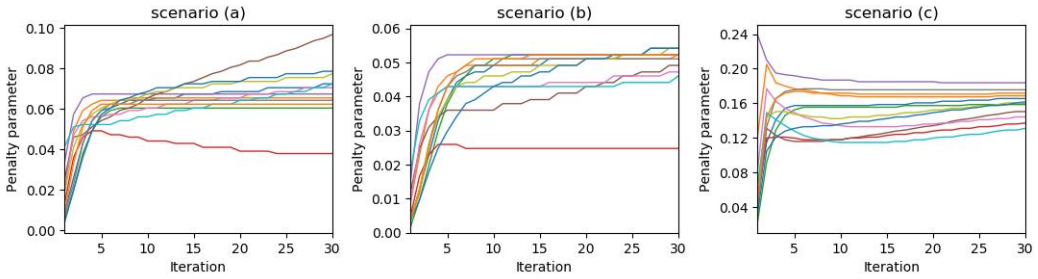

FIGURE 2.7 – Estimated penalty parameters ρ_k as a function of iteration index k, for different images of the CAVE dataset and blurring scenarios (a), (b) and (c). Lines with different colors refer to different test images.

TABLE 2.1 reports the average values and standard deviations of RMSE, PSNR, SSIM and ERGAS. For all blurring scenarios, one can observe that our method outperformed all competing methods in terms of performance and robustness. For quality comparison, consider scenario (a) for example. FIGURE 2.3 provides the blurred image, deblurred images, ground truth of *real and fake peppers* (first row) and *superballs* (second row) from the CAVE dataset. Visually, our method provides more

details, including sharper edges and more vivid gloss. This confirms the effectiveness of the proposed method in recovering the spatial information of the latent clear HIs.

We now evaluate the proposed method on remotely sensed data: the Chikusei dataset. This dataset, with more spectral bands, allows us to analyze how our method exploits spectral information. The mean and variance of the numerical results for all methods in 6 blurring scenarios are provided in TABLE 2.2. It can be observed that the quantitative metrics of our method surpass the other competing methods in most cases. FIGURE 2.5 displays the visual results. As can be seen, the proposed method provides results with clearer and sharper visual effects compared to the other methods. This illustrates the superiority of our method in recovering the latent HIs with more spectral bands.

2.4.3.1 Convergence illustration

In many Plug-and-Play algorithms for inverse imaging problems, the ADMM is widely used as a variable splitting technique. In some works, the convergence of Plug-and-Play schemes based on some linear denoisers, including Non-Local Means (NLM) [Sreehari 2016] and Gaussian Mixture Model (GMM) [Teodoro 2017], has been proved theoretically. It is difficult to prove the convergence of our method as the B3DDN denoiser involves several non-linear operators. In practice, however, as illustrated below, we observed that the proposed deconvolution framework shows good convergence behavior.

FIGURE 2.6 provides the mean RMSE curves of our algorithm obtained for the CAVE dataset in the case of scenarios (a), (b) and (c). It can be observed that the algorithm, even with its nonlinear B3DDN denoiser, exhibits a stable and robust convergence behavior independently of the blurring kernel and noise level. Moreover, a low mean RMSE value was reached after few iterations, which indicates that early stopping can be considered to limit computation time.

2.4.3.2 Behavior with respect to Plug-and-Play internal parameter estimation

Deep priors that capture both the spatial context and spectral correlations of the latent clean HIs mainly contribute to the effectiveness of our method. But the internal parameter setting procedure and the stopping criterion also play a crucial role in achieving satisfactory performance by yielding a good balance with the contribution of deep priors. In contrast, observe that the automatic setting of the regularization parameters is not implemented by the other competing methods during test.

FIGURE 2.7 shows how the penalty parameter varies along with the iterations, for different images of the CAVE dataset, and for scenarios (a), (b), and (c). According to the Plug-and-Play principle, the estimated noise level σ_k is assumed to decrease along with the iterations, as the reconstructed image converges to a desired point. Therefore, the penalty parameter $\rho_k = \eta/\sigma_k^2$ is expected to increase [Zhang 2021]. As can be seen on FIGURE 2.7, parameter ρ changes coincide with this trend for almost

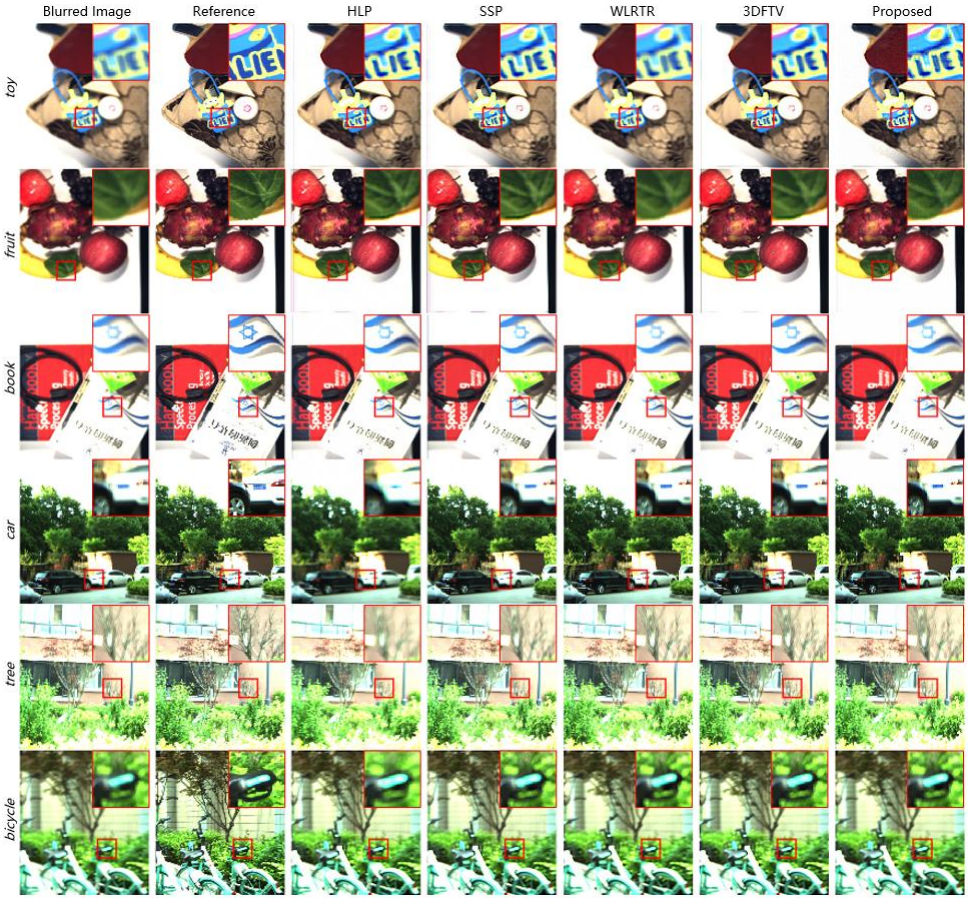

FIGURE 2.8 – Blurred images, reference images and visual results for all methods on the real-world dataset. The false color images were generated for clear visualization with the 38th, 24th and 10th channels used for red, green and blue, respectively.

all test images. FIGURE 2.6 shows the number of iterations K for scenarios (a), (b) and (c). It can be observed that our stopping criterion automatically interrupts the Plug-and-Play algorithm when it has nearly converged, which contributes to save computation time.

2.4.4 Performance evaluation on real-world data

To validate the effectiveness of our method in real-world conditions, we collected six unfocused HIs and the corresponding focused images for different indoor and outdoor scenes. Specifically, as illustrated in FIGURE 2.9, the HIs of the indoor scenes were recorded under controlled illuminations while the outdoor HIs were captured under normal daylight illumination. To fully capture the complex blurs caused by the imaging system, our dataset was elaborated to address hyperspectral deconvolution problem with respect to defocus. In particular, blurred images were obtained by making the camera out of focus while clear references were also captured

FIGURE 2.9 – Indoor (left) and outdoor (right) experimental setups for collecting real data.

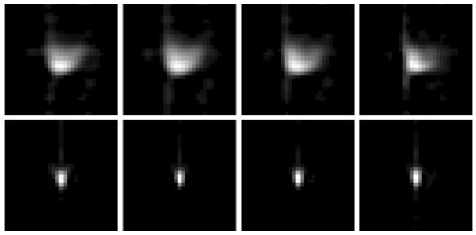

FIGURE 2.10 – Estimated blurring kernels in the 10th, 20th, 30th and 40th channels of the blurred images *fruit* (first row) and *bicycle* (second row) of the real-world dataset.

by focusing the camera. We captured these images with the GaiaField systems (see details in [Zhao 2019]) of our laboratory at Northwestern Polytechnical University. The GaiaField (Jiangsu Dualix Spectral Image Technology Co. Ltd., GaiaField-V10) is a push-broom imaging spectrometer with an HSIA-OL50 lens, covering the visible and NIR wavelengths ranging from 373.70 to 1000.90 nm, with a spectral resolution of 4.6 nm (129 channels in total). The spatial resolution of the images is 780×696 pixels.

For all acquired images, we conducted a pre-processing procedure as described in [Simões 2015]. First, we removed over-noisy and over-exposed bands. We got 45 exploitable bands, which were normalized such that the 0.999 intensity quantile corresponded to the value 1. Then, all HIs were denoised using the approach described in [Roger 1996] to enhance images. Blurred images and their clear counterparts are illustrated in the first and second columns of FIGURE 2.8, respectively. Note that these image pairs are not strictly aligned due to multiple factors affecting the camera mounting. The clear images were used for visual comparisons only. The blurring kernel in each channel was estimated using the method described in [Krishnan 2011]. For illustration purposes, FIGURE 2.10 shows the kernels in the 10th, 20th, 30th and 40th channels of the blurred images *fruit* and *bicycle*. For all experiments, we added an i.i.d. Gaussian noise to the blurred images, with a signal-to-noise ratio (SNR) set to 40 dB.

In real-world HID scenarios, no ground truth is available for training the B3DDN. Benefiting from the flexibility of the B3DDN in denoising HIs of various origins with distinct numbers of spectral bands, in this experiment we used the network

TABLE 2.3 – Time consuming of all compared methods for the blurred image *fruit* of the real-world dataset.

	HLP	SSP	WLRTR	3DFTV	Proposed
Time (sec)	9.7	622.5	10501.2	6044.4	4280.6

parameters $\Theta(\Sigma)$ learned with the CAVE dataset (31 spectral bands). FIGURE 2.8 shows the deblurred images obtained with all the competing algorithms, from columns 3 to 7. It can be seen that our method still performed better, or similarly, in recovering details compared to HLP, SSP, WLRTR, and 3DFTV, though all competing methods only achieved limited performance probably due to deviations in estimating kernels. This demonstrates the applicability of our method in real-world scenarios, as well as the necessity of further investigating blind hyperspectral deconvolution algorithms.

Finally, we conducted the experiment for evaluating the running time using the blurred image *fruit* from our real-world dataset. All the baselines were implemented using MATLAB while our method was carried out using Python. We conducted all the experiments on a server with Intel Xeon Gold 6152 CPU, 512-GB random access memory and NVIDIA Tesla P40 GPU. Time consumption of all the compared methods is shown in TABLE 2.3. It can be observed that our method achieves the most competitive deconvolution results with relatively smaller computation time when compared to WLRTR and 3DFTV.

2.5 Conclusion

In this chapter, we presented a tuning-free HID method based on the Plug-and-Play framework. Instead of using handcrafted priors, we designed a blind B3DDN denoiser based on deep learning to learn the spectral-spatial information of hyperspectral images from data and plugged it into an ADMM optimizer. The internal parameters were automatically estimated by a measure of 3D residual whiteness and learned by the B3DDN during iterations. Experimental results demonstrated that the proposed method cannot only effectively handle various simulated blurring settings but can also be applied to real-world scenarios. In the future, we will address blind HID and computational cost reduction to further enhance the applicability of our method in real-world scenarios.

Chapter 3

Deep HMIF with inter-image variability

Contents

3.1 Introduction

Another degradation phenomenon mentioned in Chapter 1 is that the high spectral resolution of HIs limits their spatial resolution because of hardware limitations [Shaw 2003]. In contrast, multispectral cameras can achieve a much higher spatial resolution but over a small number of spectral bands. Consequently, a strategy to improve the spatial resolution of HIs is to fuse them with MIs of the same scene. This results in the hyperspectral and multispectral image fusion (HMIF) problem.

Several strategies have been proposed to solve the HMIF problem. These strategies can be roughly divided into component substitution or multiresolution analysis methods, matrix or tensor factorization methods, and deep learning approaches. Component substitution or multiresolution analysis methods aim to substitute some patterns of the HI, high-frequency ones in particular, by information extracted from the MI [Yokoya 2012, Liu 2000, Aiazzi 2006]. These techniques employ different representations of the images, e.g., in the wavelet domain, which are also used for pansharpening [Vivone 2018, Loncan 2015].

Subspace-based formulations have become very popular to address HMIF problems since they significantly reduce their dimensionality [Yokoya 2012, Simões 2015]. They also have a close connection with the widely used linear mixing model [Keshava 2002, Dobigeon 2013], which represents each pixel of an HI as a linear combination of a small number of spectral signatures. Several subspace-based formulations have been proposed, often employing prior information about the basis vectors or their contributions in the decomposition, to improve the results. Examples include sparse dictionary learning [Wei 2015a, Akhtar 2015] or matrix factorization [Yokoya 2012] approaches, which can use, e.g., spatial [Simões 2015] and sparse [Kawakami 2011, Lanaras 2015] regularizers or patch-level processing [Veganzones 2016]. Efficient algorithms also convert this problem into solving a Sylvester equation [Wei 2015b]. Some approaches have considered the manifold structure of the image patches [Zhang 2018b]. Other approaches have explored the representation of HIs and MIs as three dimensional tensors [Kanatsoulis 2018, Li 2018, Prévost 2020]. Low-rank tensor models have been used to represent the high-resolution images (HRIs), such as the canonical polyadic decomposition [Kanatsoulis 2018], the Tucker decomposition [Li 2018, Prévost 2020, Borsoi 2021e], and the block term decomposition [Ding 2020].

Deep learning approaches have recently become very popular for HMIF [Li 2022, Yao 2020, Zhang 2020b]. These approaches leverage the capability of neural networks to represent complex signals and images. Early supervised approaches were based upon classical neural network architectures used in image processing such as 3D convolutional neural networks (CNN) [Palsson 2017], while more recent methods explore physical acquisition models to design architectures with improved interpretability [Chen 2022], e.g., incorporating CNN results as priors in model-based frameworks [Dian 2020, Wang 2021] or using architectures inspired by unrolling principle [Xie 2020]. However, the scarcity of training data with ground truth has motivated the development of unsupervised approaches, that depend only on the observed HI and MI. Examples include the use of autoencoders with shared weights [Qu 2018, Liu 2022, Wang 2020c], and approaches based on deep image priors [Ulyanov 2018], which parameterize the HRI as the output of a neural network and train the latter using different options for the network inputs [Zhang 2020a, Wei 2020b].

Although different strategies have been investigated to solve the HMIF problem, these methods assume that the observed HI and MI are acquired at the same time instant and under the same conditions. However, platforms carrying both hyperspectral and multispectral imaging systems are still limited [Borsoi 2020]. On the contrary, due to the wider availability of satellites with multispectral sensors, e.g., the Sentinel, Landsat and Quickbird missions, it has become of great interest to fuse HIs and MIs acquired at different time instants by different instruments [Yokoya 2017b]. When applied in these realistic conditions, most existing methods suffer from severe limitations as they ignore variability between the HI and MI. Inter-image variability includes localized spatial and spectral changes and can occur due to differences in acquisition conditions caused by, e.g., atmospheric, illumination or seasonal variations [Borsoi 2021d], as well as abrupt changes [Liu 2019a].

To tackle this issue, several HMIF frameworks addressing inter-image variability have been recently proposed [Borsoi 2020, Borsoi 2021e, Prévost 2022, Borsoi 2021a, Brezini 2021, Camacho 2022, Fu 2021]. A detailed review of these methods is provided in Section 3.2. These methods formulate the HMIF problem with a key difference when compared to the original approaches: the HI and the MI are assumed to be generated from distinct HRIs, which are allowed to be different because of spatially homogeneous variations [Borsoi 2020, Prévost 2022] or spatially localized ones [Borsoi 2021e]. However, considering inter-image variability renders the HMIF problem significantly more ill-posed, which makes the use of appropriate prior information about the HRIs very important in order to achieve good performance.

Existing HMIF works that consider inter-image variability rely on handcrafted priors, such as low-rank matrix [Borsoi 2020] or tensor [Borsoi 2021e, Prévost 2022] decompositions. However, these priors are not adequate to model complex contents embedded in real HIs. Without considering inter-image variability, this issue has been addressed in the HMIF problem by exploring the powerful representation capability of deep learning methods, as noted by various recent works on this topic. Nevertheless, devising learning-based approaches to address inter-image variability in HMIF incurs additional challenges, first because very little data is available for training. Indeed, since inter-image variability originates from complex physical phenomena, it is difficult to generate realistic synthetic data to be used for training even if HIs of a single scene are available. This makes learning an end-to-end mapping from an HI and an MI to the HRIs unfeasible.

Recently, deep image priors [Ulyanov 2018] and plug-and-play strategies [Venkatakrishnan 2013] have been used to introduce prior information with either pre-trained or unsupervised neural networks. However, adequately addressing inter-image variability requires considering two different HRIs, underlying the HI and the MI, respectively. Thus, directly exploiting such strategies to address inter-image variability in HMIF is not very effective since: 1) existing strategies in this category would fail to account for the joint prior information between the two HRIs, and 2) each of the images can have distinct statistical properties, which makes obtaining adequate priors more difficult. Moreover, although deep image priors are unsupervised [Ulyanov 2018], they require careful setup of the network architecture and the number of stochastic gradient iterations to produce reasonable results. It must be noted that these challenges related to the lack of training data and the corresponding difficulty in learning priors of the scene of interest are also encountered more generally in HMIF, i.e., even when inter-image variability is not present.

In this chapter, we propose a new image fusion method accounting for inter-image variability between HIs and MIs which addresses the aforementioned challenges. First, to adequately represent the image-specific information as well as the joint prior information between the two HRIs, we propose a mixture distribution that accounts for the leptokurtic nature of the inter-image variations while, at the same time, represents complex image content by implicitly exploiting learning-based image priors. An iteratively reweighted optimization strategy is then proposed, and the regularization by denoising (RED) [Romano 2017] framework is employed to implicitly

introduce prior information about the HRIs by means of denoising engines, one for each latent HRI. The denoisers are trained using a zero-shot strategy [Shocher 2018] and adapted during the optimization process, which allows them to account for the content of each individual HRI. The proposed algorithm is called *Deep hyperspectral and multispectral Image Fusion with Inter-image Variability* (DIFIV). Experiments on data with real inter-image variability demonstrate the superiority of DIFIV compared to other state-of-the-art methods. The contributions of the chapter are summarized as follows.

— A general imaging model is formulated, where the inter-image variations of the HRIs are modeled by a hyper-Laplacian distribution to account for the joint image content, while the image content specific to each HRI is learned by two distinct deep CNNs.

— To solve the non-convex, non-smooth HMIF optimization problem, a variable splitting strategy is combined with an iteratively reweighted scheme to tackle the difficulties introduced by both the hyper-Laplacian and deep priors, which are defined implicitly based on CNN denoisers under the RED framework.

— We use a zero-shot strategy inspired by [Shocher 2018] to learn the CNN denoisers based only on the observed HI and MI. Moreover, unlike the original use of zero-shot methods for single image restoration, the denoisers are trained iteratively during the optimization process based on the currently estimated HRIs. This allows the learned priors to represent the individual information in each of the HRIs adaptively while incorporating at the same time information from both low resolution images as the method converges. Furthermore, the architecture of CNNs is made lightweight by considering separable convolutions and a low-rank representation of HIs to yield a small number of network parameters.

The chapter is organized as follows. In Section 3.2, the HI and MI observation processes are presented, as well as a review of recent methods considering inter-image variability. Section 3.3 formulates a new model and introduces the proposed method. Experimental results with data containing real inter-image variability are given in Section 3.4. Finally, Section 3.5 concludes the chapter.

3.2 Image Fusion with Inter-image Variability

Let us denote an HI with L_h bands and N pixels by $\boldsymbol{Y}_h \in \mathbb{R}^{L_h \times N}$, and an MI with L_m bands and M pixels by $\boldsymbol{Y}_m \in \mathbb{R}^{L_m \times M}$, where $L_m < L_h$ and $N < M$. These images are assumed to be degraded versions of a pair of underlying HRIs $\boldsymbol{X}_h \in \mathbb{R}^{L_h \times M}$ and $\boldsymbol{X}_m \in \mathbb{R}^{L_h \times M}$ with high spatial and spectral resolutions, which are related according to the following model:

$$\begin{aligned} \boldsymbol{Y}_h &= \boldsymbol{X}_h \boldsymbol{F} \boldsymbol{D} + \boldsymbol{N}_h \,, \\ \boldsymbol{Y}_m &= \boldsymbol{R} \boldsymbol{X}_m + \boldsymbol{N}_m \,, \end{aligned} \tag{3.1}$$

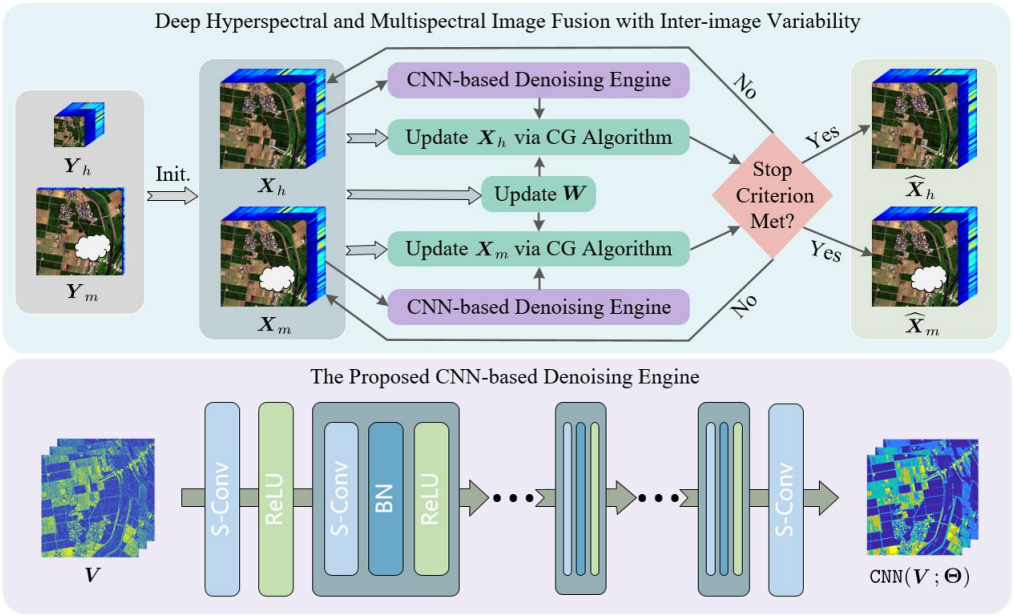

FIGURE 3.1 – **Top panel**: Overall illustration of the proposed Deep hyperspectral and multispectral Image Fusion with Inter-image Variability (DIFIV) method: the HRIs underlying the HI and MI ($\boldsymbol{X}_h$ and $\boldsymbol{X}_m$) are initialized (Init.), with interpolations of observed images ($\boldsymbol{Y}_h$ and $\boldsymbol{Y}_m$), and then used to compute the inter-image variability weighting term $\boldsymbol{W}$ and to update the CNN-based denoisers; afterward, these are used to re-compute the HRIs using a conjugate-gradient based algorithm; this process is repeated iteratively until convergence. **Bottom panel**: The neural network architecture of our CNN-based denoising engine (S-Conv, BN, and ReLU stand for separable convolution, batch normalization and rectified linear unit layers, respectively).

in which matrices $\boldsymbol{F} \in \mathbb{R}^{M\times M}$ and $\boldsymbol{D} \in \mathbb{R}^{M\times N}$ represent optical blurring and spatial downsampling occurring at the hyperspectral sensor, respectively; The matrix $\boldsymbol{R} \in \mathbb{R}^{L_m\times L_h}$ contains the spectral response functions (SRF) of the multispectral instrument, and $\boldsymbol{N}_h \in \mathbb{R}^{L_h\times N}$ and $\boldsymbol{N}_m \in \mathbb{R}^{L_m\times M}$ denote additive noises.

In this setting, the image fusion problem consists of recovering the HRIs $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$ given the observations $\boldsymbol{Y}_h$ and $\boldsymbol{Y}_m$. Most of the previous methods consider that $\boldsymbol{Y}_h$ and $\boldsymbol{Y}_m$ are degraded from the same source, i.e., $\boldsymbol{X}_h = \boldsymbol{X}_m$, which intrinsically assumes that they are acquired under the same conditions, e.g., by sensors on board a single satellite. However, due to the wider availability of satellites equipped with multispectral sensors, it is of great interest to fuse HIs and MIs acquired by different instruments at different time instants [Yokoya 2017b]. In that case, by assuming that $\boldsymbol{X}_h = \boldsymbol{X}_m$, most existing methods ignore variabilities between the HI and MI, which can occur due to differences in acquisition conditions caused by, e.g., atmospheric, illumination or seasonal variations [Borsoi 2021d], or abrupt changes [Liu 2019a].

Recently, image fusion frameworks addressing inter-image variability have been proposed in [Borsoi 2020, Borsoi 2021e, Prévost 2022, Borsoi 2021a, Brezini 2021, Camacho 2022, Fu 2021]. Such methods estimate both HRIs $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$ by using different assumptions to model both the images and the inter-image changes. The first method to address this problem was FuVar [Borsoi 2020]. It considers that the HRIs satisfy the linear mixing model (LMM) [Keshava 2002], but with a distinct set of spectral basis vectors for each image:

$$\boldsymbol{X}_h = \boldsymbol{M}_h\boldsymbol{A}\,, \quad \boldsymbol{X}_m = \boldsymbol{M}_m\boldsymbol{A}\,, \tag{3.2}$$

where $\boldsymbol{M}_h$ and $\boldsymbol{M}_m \in \mathbb{R}^{L_h\times R}$ denote the set of spectral basis vectors related to the HI and MI, respectively, and $\boldsymbol{A} \in \mathbb{R}^{R\times M}$ their corresponding spatial coefficients. Note that $\boldsymbol{M}_h$ and $\boldsymbol{M}_m$ are associated with the spectral signatures of the pure materials (i.e., the endmembers) in the HI and MI, respectively. FuVar considers $\boldsymbol{M}_h$ and $\boldsymbol{M}_m$ to be related to one another through a set of smooth multiplicative scaling factors $\boldsymbol{\Phi} \in \mathbb{R}^{L_h\times R}$ [Imbiriba 2018]:

$$\boldsymbol{M}_m = \boldsymbol{M}_h \odot \boldsymbol{\Phi}\,, \tag{3.3}$$

where $\odot$ denotes the Hadamard product. Thus, this model successfully accounts for changes in the spectral signatures of the endmembers between the HI and the MI, which can occur when the materials are affected by seasonal variations or when the MI is affected by uniform changes caused by, e.g., different illumination conditions. However, the coefficients $\boldsymbol{A}$ shared by both images limit the capability of FuVar to represent inter-image changes in the spatial domain.

This limitation has been addressed by considering spatially and spectrally localized inter-image variations through an additive model in a tensor-based framework [Borsoi 2021e]. This latter work considers a model of the form:

$$\boldsymbol{X}_h\,, \quad \boldsymbol{X}_m = \boldsymbol{X}_h + \boldsymbol{\Psi}\,, \tag{3.4}$$

where $\boldsymbol{\Psi} \in \mathbb{R}^{L_h\times M}$ denotes a set of additive variability factors. Both the HRI $\boldsymbol{X}_h$ and the variability $\boldsymbol{\Psi}$ are assumed to admit a Tucker tensor decomposition

with low multilinear ranks [Sidiropoulos 2017]. This reduces the dimensionality of the problem and allows theoretical identifiability and recovery guarantees to be obtained [Borsoi 2021e].

A related work proposes to jointly address the image fusion and hyperspectral unmixing problems in the presence of inter-image spectral variability [Prévost 2022]. This consists of the recovery of both the HRIs and the spectral signatures of the endmembers and their abundances. An LL1 tensor model is considered, which is closely related to the LMM in (3.2) but involves an additional low-rank assumption on the coefficient maps $\boldsymbol{A}$ that allows theoretical identifiability results to be derived. Other works propose to consider intra-image variability by extending the LMM to consider spatial endmember variability, i.e., variability within a single image [Brezini 2021, Camacho 2022]. Another work considers a robust version of the data fidelity term related to the MI in the cost function to reduce the impact of possible changes or outliers in the image fusion process [Fu 2021]. However, these methods still assume that the HRIs underlying the HI and the MI are equal.

Despite the success of these approaches in addressing the inter-image variability problem, they all rely on handcrafted priors for the HR images $\boldsymbol{X}_h$ and $\boldsymbol{X}_h$, which limits their capability of representing realistic and complex image content. In this work, we propose an image fusion method that leverages the expressive power of CNNs in order to construct accurate image priors for the HRIs while accounting for inter-image variability, as detailed in the following section.

3.3 Proposed method

The proposed image fusion method is based on three important axes/contributions: 1) an imaging model that incorporates inter-image variability with learned image priors, 2) an optimization scheme that can handle these flexible penalties, 3) a lightweight unsupervised (zero-shot) scheme to iteratively learn deep priors of the latent HRIs during the reconstruction process. The proposed image fusion method is presented through four steps. First, we present the imaging model in Subsection 3.3.1 and formulate the optimization problem. In Subsection 3.3.2 we describe an iteratively reweighted scheme to optimize the cost function. The optimization steps, as well as the integration of deep priors, are described in Subsection 3.3.3. We then address the design of CNN architecture and its image-adapted training strategy in Subsection 3.3.4. An overall illustration of the proposed DIFIV method is shown in FIGURE 3.1.

3.3.1 The imaging model

Using a probabilistic framework, the HMIF problem can be formulated as the recovery of the mean or mode of the posterior probability density function (PDF) $p(\boldsymbol{X}_h, \boldsymbol{X}_m | \boldsymbol{Y}_h, \boldsymbol{Y}_m)$ of both HRIs given the LR observations. Using Bayes theorem,

this PDF can be written as:

$$p(\boldsymbol{X}_h, \boldsymbol{X}_m|\boldsymbol{Y}_h, \boldsymbol{Y}_m) \propto p(\boldsymbol{Y}_h|\boldsymbol{X}_h)p(\boldsymbol{Y}_m|\boldsymbol{X}_m)p(\boldsymbol{X}_m, \boldsymbol{X}_h)\,, \tag{3.5}$$

where we assumed the HI and MI to be conditionally independent given their high-resolution counterparts.

The likelihoods of the observed images $\boldsymbol{Y}_h$ and $\boldsymbol{Y}_m$ can be written according to their data generation process in (3.1). More precisely, assuming the elements of $\boldsymbol{N}_h$ and $\boldsymbol{N}_m$ to be i.i.d. Gaussian random variables with variance σ_h^2 and σ_m^2, respectively, the conditional distributions of $\boldsymbol{Y}_m$ and $\boldsymbol{Y}_n$ in (3.5) are given by:

$$p(\boldsymbol{Y}_h|\boldsymbol{X}_h) = \mathcal{MN}(\boldsymbol{X}_h\boldsymbol{F}\boldsymbol{D}, \sigma_h^2\boldsymbol{I}_{L_h}, \boldsymbol{I}_N)\,, \tag{3.6}$$

$$p(\boldsymbol{Y}_m|\boldsymbol{X}_m) = \mathcal{MN}(\boldsymbol{R}\boldsymbol{X}_m, \sigma_m^2\boldsymbol{I}_{L_m}, \boldsymbol{I}_M)\,, \tag{3.7}$$

where $\mathcal{MN}(\boldsymbol{\Upsilon}, \boldsymbol{\Sigma}_r, \boldsymbol{\Sigma}_c)$ denotes the matrix normal distribution with mean matrix $\boldsymbol{\Upsilon}$ and row and column covariance matrices $\boldsymbol{\Sigma}_r$ and $\boldsymbol{\Sigma}_c$, respectively [Wei 2015b].

The challenging question concerns how to meaningfully define the joint prior $p(\boldsymbol{X}_m, \boldsymbol{X}_h)$ for both HRIs. This question is not trivial when the images differ due to acquisition conditions or seasonal variations. A simplistic possibility is to consider the images to be independent and to use priors used for super-resolution without variability, such as low-rank matrix and tensor models [Yokoya 2012, Kanatsoulis 2018, Prévost 2019], piecewise-smoothness [Simões 2015] or learned deep priors [Wang 2021, Dian 2020, Wang 2022b]. However, the images $\boldsymbol{X}_m$ and $\boldsymbol{X}_h$ are observations of the same scene, and thus are strongly dependent. Considering this, we can state the following desirable properties for $p(\boldsymbol{X}_m, \boldsymbol{X}_h)$:

— Apart from possible smooth inter-image variations (such as, e.g., illumination or atmospheric changes, which tend to impact the images uniformly [Borsoi 2021d]), changes between $\boldsymbol{X}_m$ and $\boldsymbol{X}_h$ are generally small and sparse; high magnitude changes are concentrated in a relatively small number of pixels and bands [Liu 2019a].

— The prior should promote images $\boldsymbol{X}_m$ and $\boldsymbol{X}_h$ which are statistically similar to real hyperspectral images (e.g., they can be well represented by learned priors).

To achieve the above desiderata, we consider a mixture distribution, given by:

$$\begin{aligned}\log p(\boldsymbol{X}_m, \boldsymbol{X}_h) \propto\ & -\frac{\eta_p}{2}\sum_{\ell,n}\left|\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}\right|^p \\ & - \eta_m\phi_m(\boldsymbol{X}_m) - \eta_h\phi_h(\boldsymbol{X}_h)\,,\end{aligned} \tag{3.8}$$

for $0 < p \leqslant 1$, where $\delta_h^{(\ell,n)}$ and $\delta_m^{(\ell,n)}$ denote the (ℓ, n)-th locations of a high-pass spatio-spectral filtered version of $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$, which are denoted by $\boldsymbol{\Delta}_h$ and $\boldsymbol{\Delta}_m$, respectively. We assume this filtering to be computed through an operator $\mathcal{G}$ satisfying $\boldsymbol{\Delta}_h = \mathcal{G}(\boldsymbol{X}_h)$, $\boldsymbol{\Delta}_m = \mathcal{G}(\boldsymbol{X}_m)$, and in vector form as $\text{vec}(\boldsymbol{\Delta}_h) = \boldsymbol{G}\,\text{vec}(\boldsymbol{X}_h)$ and $\text{vec}(\boldsymbol{\Delta}_m) = \boldsymbol{G}\,\text{vec}(\boldsymbol{X}_m)$ where $\boldsymbol{G}$ is the matrix form of $\mathcal{G}$. One natural example for $\mathcal{G}$ is the spatio-spectral gradient operator, e.g. Laplacian filter. Parameters η_p, η_m and η_h are regularization parameters.

The first term with $0 < p \leqslant 1$ rather than $p = 2$ in (3.8) corresponds to an i.i.d. hyper-Laplacian distribution for the difference between the filtered HRIs [Krishnan 2009b], which has also been previously used to represent the gradient of the HRI in image fusion [Peng 2021]. This distribution is effective for modeling leptokurtic (i.e., heavy-tailed) distributions such as images [Krishnan 2009b]. This can represent an important characteristic of the inter-image changes since these can be restricted to a comparatively small number of pixels and are concentrated at low-frequency spatial content [Borsoi 2018]. The functions $\phi_h(\cdot)$ and $\phi_m(\cdot)$ encode prior knowledge about each HRI, and will be learned implicitly by using deep CNNs.

Note that the prior in (3.8) also corresponds to a model for the inter-image variability, which can be written as:

$$\boldsymbol{X}_m = \boldsymbol{X}_h + \boldsymbol{\Psi}_\Delta \,. \tag{3.9}$$

What is distinctive in (3.9) when compared to the model in (3.4) is how prior information is chosen. The prior for the inter-image variability term $\boldsymbol{\Psi}_\Delta$ cannot be written in an analytical form; instead, its properties follow from the interactions of the different terms in (3.8). The first term encourages the inter-image variability $\boldsymbol{\Psi}_\Delta$ to have small and sparse gradients. The last two terms employ CNNs that can incorporate realistic prior information about each of the HRIs, and only constrain $\boldsymbol{\Psi}_\Delta$ indirectly through its effect on $\boldsymbol{X}_m$ and $\boldsymbol{X}_h$.

Given this model, the image fusion problem then consists of finding the HRIs $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$ which maximize the logarithm of the posterior distribution $p(\boldsymbol{X}_h, \boldsymbol{X}_m | \boldsymbol{Y}_h, \boldsymbol{Y}_m)$ defined in (3.5). This corresponds to the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{X}_h, \boldsymbol{X}_m} \; & \frac{1}{2}\|\boldsymbol{Y}_h - \boldsymbol{X}_h \boldsymbol{F} \boldsymbol{D}\|_F^2 + \frac{1}{2}\|\boldsymbol{Y}_m - \boldsymbol{R}\boldsymbol{X}_m\|_F^2 \\ & + \eta_h \phi_h(\boldsymbol{X}_h) + \eta_m \phi_m(\boldsymbol{X}_m) \\ & + \frac{\eta_p}{2}\|\mathcal{G}(\boldsymbol{X}_h) - \mathcal{G}(\boldsymbol{X}_m)\|_p^p \,, \end{aligned} \tag{3.10}$$

where $\|\cdot\|_p$ is the entrywise L_p matrix norm, satisfying $\|\mathcal{G}(\boldsymbol{X}_h) - \mathcal{G}(\boldsymbol{X}_m)\|_p^p = \sum_{\ell,n} |\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}|^p$. The spatial and spectral priors of $\boldsymbol{X}_m$ and $\boldsymbol{X}_h$ are encoded in $\phi_h(\boldsymbol{X}_h)$ and $\phi_m(\boldsymbol{X}_m)$, respectively.

3.3.2 An iteratively reweighted update scheme

Optimizing the cost function in (3.10) is challenging. Apart from the image priors $\phi_h(\cdot)$ and $\phi_m(\cdot)$ that will be defined in the sequel, the inter-image prior term (i.e., the last term in (3.10)) is, in general, a non-convex and non-smooth function of both $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$, which is not straightforward to optimize. To address this problem, we consider an iteratively reweighted optimization strategy [Lu 2014, Ammanouil 2014]. First, note that the last term in (3.10) can be written as :

$$\sum_{\ell,n} |\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}|^p = \sum_{\ell,n} w_{\ell,n} |\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}|^2 \,, \tag{3.11}$$

where the weights $w_{\ell,n}$ are given by

$$w_{\ell,n} = |\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}|^{p-2} . \tag{3.12}$$

Since $w_{\ell,n} \geqslant 0$, (3.11) can be expressed as:

$$\sum_{\ell,n} w_{\ell,n} \left|\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}\right|^2 = \|\boldsymbol{W} \odot (\boldsymbol{\Delta}_h - \boldsymbol{\Delta}_m)\|_F^2 , \tag{3.13}$$

where $\boldsymbol{W}$ is a matrix whose (ℓ, n)-th entry is given by $\sqrt{w_{\ell,n}}$, and $\odot$ denotes the Hadamard product.

When matrix $\boldsymbol{W}$ is fixed given $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$, (3.13) becomes a quadratic function of the HRIs, which can be effectively optimized. The nonlinear dependency of $\boldsymbol{W}$ on $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$ will be resolved by using an iterative strategy: first the cost function is optimized considering $\boldsymbol{W}$ fixed to obtain $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$, and afterwards $\boldsymbol{W}$ is updated according to an approximate version of (3.12) by using the values of $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$ computed from previous iteration [Lu 2014]. This leads to the following iterative procedure, which is repeated until convergence:

1) For a fixed $\boldsymbol{W}$, compute $\boldsymbol{X}_h$ and $\boldsymbol{X}_m$ by solving the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{X}_h, \boldsymbol{X}_m, \boldsymbol{\Delta}_h, \boldsymbol{\Delta}_m} \; & \frac{1}{2}\|\boldsymbol{Y}_h - \boldsymbol{X}_h \boldsymbol{F} \boldsymbol{D}\|_F^2 + \frac{1}{2}\|\boldsymbol{Y}_m - \boldsymbol{R}\boldsymbol{X}_m\|_F^2 \\ & + \eta_h \phi_h(\boldsymbol{X}_h) + \eta_m \phi_m(\boldsymbol{X}_m) + \frac{\eta_p}{2}\|\boldsymbol{W} \odot (\boldsymbol{\Delta}_h - \boldsymbol{\Delta}_m)\|_F^2 \\ \text{s.t.} \; & \boldsymbol{\Delta}_h = \mathcal{G}(\boldsymbol{X}_h), \; \boldsymbol{\Delta}_m = \mathcal{G}(\boldsymbol{X}_m) . \end{aligned} \tag{3.14}$$

2) Update the entries of $\boldsymbol{W}$ according to

$$w_{\ell,n} = \left(|\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}| + \varepsilon\right)^{p-2} , \tag{3.15}$$

where $\varepsilon > 0$ is a small constant included in (3.12) to ensure the numerical stability of the algorithm.

3) Return to step 1) and repeat until convergence.

This strategy is efficient in solving sparsity-regularized optimization problems [Daubechies 2010]. Moreover, iteratively reweighted optimization schemes have been shown to converge to a local stationary point under relatively mild conditions [Lu 2014]. The main limitation of this scheme is the assumption that $|\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}|^{2-p}$ (which is equal to ${w_{\ell,n}}^{-1}$) is nonzero, though this limitation is addressed in (3.15) with ε. When this expression approaches zero during optimization, $w_{\ell,n}$ becomes excessively large, causing the term $|\delta_h^{(\ell,n)} - \delta_m^{(\ell,n)}|^2$ in the cost function (3.14) to approach zero. Nevertheless, despite this limitation, it has been employed for sparse unmixing [Zhang 2018c] due to the convenience it offers in optimizing the cost function.

In the following subsection, we shall focus on the minimization problem (3.14).

3.3.3 The optimization problem

Handcrafting powerful regularizers $\phi_h(\boldsymbol{X}_h)$ and $\phi_m(\boldsymbol{X}_m)$ along with solving the associated optimization problems efficiently is not a trivial task. In this subsection, we propose to learn the image prior directly from the observed data and incorporate it into the model-based optimization (3.14) to avoid designing regularizers analytically.

First, by introducing two auxiliary variables, $\boldsymbol{Z}_h = \boldsymbol{X}_h$ and $\boldsymbol{Z}_m = \boldsymbol{X}_m$, problem (3.14) can be rewritten equivalently as:

$$\begin{aligned} \min_{\Omega} \; & \frac{1}{2}\|\boldsymbol{Y}_h - \boldsymbol{X}_h\boldsymbol{F}\boldsymbol{D}\|_F^2 + \frac{1}{2}\|\boldsymbol{Y}_m - \boldsymbol{R}\boldsymbol{X}_m\|_F^2 + \frac{\eta_p}{2}\|\boldsymbol{W} \odot (\boldsymbol{\Delta}_h - \boldsymbol{\Delta}_m)\|_F^2 \\ & + \eta_m \phi_m(\boldsymbol{Z}_m) + \eta_h \phi_h(\boldsymbol{Z}_h) \\ \text{s.t.} \quad & \boldsymbol{Z}_h = \boldsymbol{X}_h \,,\; \boldsymbol{Z}_m = \boldsymbol{X}_m \,,\; \boldsymbol{\Delta}_h = \mathcal{G}(\boldsymbol{X}_h) \,,\; \boldsymbol{\Delta}_m = \mathcal{G}(\boldsymbol{X}_m) \,, \end{aligned} \tag{3.16}$$

where $\Omega = \{\boldsymbol{X}_h, \boldsymbol{X}_m, \boldsymbol{Z}_h, \boldsymbol{Z}_m, \boldsymbol{\Delta}_h, \boldsymbol{\Delta}_m\}$. By using the half-quadratic splitting (HQS) approach [Geman 1995], we can decouple the data fidelity and regularization terms in (3.16) and write this cost function as:

$$\begin{aligned} \mathcal{L}_\rho(\boldsymbol{X}_h, \boldsymbol{X}_m, \boldsymbol{Z}_h, \boldsymbol{Z}_m) = & \frac{1}{2}\|\boldsymbol{Y}_h - \boldsymbol{X}_h\boldsymbol{F}\boldsymbol{D}\|_F^2 + \frac{1}{2}\|\boldsymbol{Y}_m - \boldsymbol{R}\boldsymbol{X}_m\|_F^2 \\ & + \frac{\eta_p}{2}\|\boldsymbol{W} \odot (\mathcal{G}(\boldsymbol{X}_h) - \mathcal{G}(\boldsymbol{X}_m))\|_F^2 \\ & + \frac{\rho}{2}\|\boldsymbol{X}_m - \boldsymbol{Z}_m\|_F^2 + \frac{\rho}{2}\|\boldsymbol{X}_h - \boldsymbol{Z}_h\|_F^2 \\ & + \eta_m \phi_m(\boldsymbol{Z}_m) + \eta_h \phi_h(\boldsymbol{Z}_h) \,, \end{aligned} \tag{3.17}$$

with $\rho \in \mathbb{R}_+$ the penalty parameter. In the following, we consider a block coordinate descent (BCD) strategy and minimize $\mathcal{L}_\rho$ with respect to each variable, one at a time.

Optimization w.r.t. $\boldsymbol{X}_h$: This optimization problem can be written as:

$$\begin{aligned} \min_{\boldsymbol{X}_h} \; & \frac{1}{2}\|\boldsymbol{Y}_h - \boldsymbol{X}_h\boldsymbol{F}\boldsymbol{D}\|_F^2 + \frac{\eta_p}{2}\|\boldsymbol{W} \odot (\mathcal{G}(\boldsymbol{X}_h) - \mathcal{G}(\boldsymbol{X}_m))\|_F^2 \\ & + \frac{\rho}{2}\|\boldsymbol{X}_h - \boldsymbol{Z}_h\|_F^2 \,. \end{aligned} \tag{3.18}$$

By taking the derivative of the cost function in (3.18), setting it equal to zero and using the vectorization property of matrix products, we obtain:

$$\begin{aligned} & -\big[(\boldsymbol{F}\boldsymbol{D})^\top \otimes \boldsymbol{I}\big]^\top \Big(\operatorname{vec}(\boldsymbol{Y}_h) - \big[(\boldsymbol{F}\boldsymbol{D})^\top \otimes \boldsymbol{I}\big] \operatorname{vec}(\boldsymbol{X}_h) \Big) \\ & + \eta_p \boldsymbol{G}^\top \operatorname{Diag}(\operatorname{vec}(\boldsymbol{W}))^2 \boldsymbol{G} \big(\operatorname{vec}(\boldsymbol{X}_h - \boldsymbol{X}_m) \big) + \rho \operatorname{vec}(\boldsymbol{X}_h - \boldsymbol{Z}_h) = \boldsymbol{0} \,. \end{aligned} \tag{3.19}$$

Using the properties of the Kronecker product, this equation can be written as:

$$\begin{aligned} & \Big(\big[(\boldsymbol{F}\boldsymbol{D})(\boldsymbol{F}\boldsymbol{D})^\top \otimes \boldsymbol{I}\big] + \eta_p \boldsymbol{G}^\top \operatorname{Diag}(\operatorname{vec}(\boldsymbol{W}))^2 \boldsymbol{G} + \rho \boldsymbol{I} \Big) \operatorname{vec}(\boldsymbol{X}_h) \\ & = \big[(\boldsymbol{F}\boldsymbol{D})^\top \otimes \boldsymbol{I}\big]^\top \operatorname{vec}(\boldsymbol{Y}_h) + \eta_p \boldsymbol{G}^\top \operatorname{Diag}(\operatorname{vec}(\boldsymbol{W}))^2 \boldsymbol{G} \operatorname{vec}(\boldsymbol{X}_m) \\ & + \rho \operatorname{vec}(\boldsymbol{Z}_h) \,, \end{aligned} \tag{3.20}$$

which is a linear system of equations. However, solving this system directly is prohibitive due to its large dimension. Since the matrix on the left-hand side is symmetric positive-definite, we propose to solve this problem using the conjugate gradient (CG) algorithm, which requires only matrix-vector products that can be implemented implicitly and more efficiently.

Optimization w.r.t. $\boldsymbol{X}_m$: This optimization problem can be written as:

$$\begin{aligned}\min_{\boldsymbol{X}_m} \; & \frac{1}{2}\|\boldsymbol{Y}_m - \boldsymbol{R}\boldsymbol{X}_m\|_F^2 + \frac{\eta_p}{2}\|\boldsymbol{W} \odot (\mathcal{G}(\boldsymbol{X}_h) - \mathcal{G}(\boldsymbol{X}_m))\|_F^2 \\ & + \frac{\rho}{2}\|\boldsymbol{Z}_m - \boldsymbol{X}_m\|_F^2 \,. \end{aligned} \tag{3.21}$$

Following the same steps as for problem (3.18), we obtain:

$$\begin{aligned}&\Big(\big[\boldsymbol{I} \otimes \boldsymbol{R}^\top \boldsymbol{R}\big] + \eta_p \boldsymbol{G}^\top \operatorname{Diag}(\operatorname{vec}(\boldsymbol{W}))^2 \boldsymbol{G} + \rho \boldsymbol{I}\Big) \operatorname{vec}(\boldsymbol{X}_m) \\ &= \big[\boldsymbol{I} \otimes \boldsymbol{R}\big]^\top \operatorname{vec}(\boldsymbol{Y}_m) + \eta_p \boldsymbol{G}^\top \operatorname{Diag}(\operatorname{vec}(\boldsymbol{W}))^2 \boldsymbol{G} \operatorname{vec}(\boldsymbol{X}_h) \\ &+ \rho \operatorname{vec}(\boldsymbol{Z}_m) \,. \end{aligned} \tag{3.22}$$

Considering that the matrix on the left-hand side is symmetric positive-definite, the CG algorithm is used to solve this problem.

Optimization w.r.t. $\boldsymbol{Z}_h$: This optimization problem can be written as:

$$\min_{\boldsymbol{Z}_h} \; \frac{\rho}{2}\|\boldsymbol{Z}_h - \boldsymbol{X}_h\|_F^2 + \eta_h \phi_h(\boldsymbol{Z}_h) \,. \tag{3.23}$$

As discussed above, designing accurate handcrafted regularizers for $\phi_h(\boldsymbol{Z}_h)$ may be complicated. To address this issue efficiently, we propose to use a strategy that leverages a CNN denoiser. Popular strategies are the Plug-and-Play framework [Venkatakrishnan 2013] and the Regularization by Denoising (RED) scheme [Romano 2017]. In this work, we consider the RED strategy since it is associated with an explicit optimization objective and because it was experimentally shown in [Romano 2017] to have more stable convergence and robustness in relation to the selection of hyperparameters when compared to Plug-and-Play methods. Consider denoising an HI $\boldsymbol{Z}$, we define the CNN denoiser as $\mathcal{D}(\cdot)$. RED framework defines $\phi_h(\cdot)$ as the inner product between an image and its denoising residual:

$$\phi_h(\boldsymbol{Z}) = \frac{1}{2}\langle \boldsymbol{Z}, \boldsymbol{Z} - \mathcal{D}(\boldsymbol{Z}) \rangle \,, \tag{3.24}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. This can be interpreted as an image-adaptive Laplacian regularizer. Using (3.24), the optimization problem (3.23) becomes

$$\min_{\boldsymbol{Z}_h} \; \frac{\rho}{2}\|\boldsymbol{Z}_h - \boldsymbol{X}_h\|_F^2 + \frac{\eta_h}{2}\langle \boldsymbol{Z}_h, \boldsymbol{Z}_h - \mathcal{D}(\boldsymbol{Z}_h) \rangle \,. \tag{3.25}$$

Taking the derivative of the cost function and setting it to zero, we obtain:

$$\rho(\boldsymbol{Z}_h - \boldsymbol{X}_h) + \eta_h(\boldsymbol{Z}_h - \mathcal{D}(\boldsymbol{Z}_h)) = \boldsymbol{0} \,. \tag{3.26}$$

To solve this equation, a fixed-point iterative update is used, leading to the following recursive update equation:

$$\boldsymbol{Z}_h^{(i+1)} = \frac{1}{\rho + \eta_h}\left(\rho \boldsymbol{X}_h + \eta_h \mathcal{D}(\boldsymbol{Z}_h^{(i)})\right). \tag{3.27}$$

where $\boldsymbol{Z}_h^{(i)}$ denotes the solution $\boldsymbol{Z}_h$ at the i-th iteration.

Optimization w.r.t. $\boldsymbol{Z}_m$: Following the same strategy as above, we obtain:

$$\boldsymbol{Z}_m^{(i+1)} = \frac{1}{\rho + \eta_m}\left(\rho \boldsymbol{X}_m + \eta_m \mathcal{D}(\boldsymbol{Z}_m^{(i)})\right). \tag{3.28}$$

Note that we only use a single step for the fixed point iteration in (3.27) and (3.28) for computational efficiency.

3.3.4 Learning deep priors via image-specific CNNs

Generally, function $\mathcal{D}(\cdot)$ can be any off-the-shelf denoiser. This offers the opportunity of incorporating a fast CNN denoising engine with powerful prior learning ability into physical model-based iterative optimization procedure [Chen 2022]. However, there are three main challenges in using CNN denoisers to learn priors for hyperspectral images in RED or Plug-and-Play frameworks [Dian 2020, Wang 2020a]: First, there is a limited amount of data available for training; second, there is an even greater scarcity of labeled training data; third, the noise level of the HRI to be denoised in (3.27)-(3.28) changes over the BCD iterations as the method converges. To overcome each of these challenges, we propose a lightweight, unsupervised and image-specific CNN denoiser, which is detailed in the following.

Lightweight network architecture: To overcome the limited number of available data to train efficient CNN denoisers, a lightweight architecture with fewer parameters needs to be considered in the network design. In this work, two strategies have been considered to lighten network architecture, namely: 1) dimensionality reduction of the input image, which reduces the number of CNN filters, and 2) separable convolutions [Howard 2017], which reduces the filter volume (i.e., the number of parameters of each filter).

We considered the DnCNN [Zhang 2017a] as a backbone in network design. For color (i.e., RGB) images, each layer of DnCNN contains 64 filters. Directly using this network architecture to denoise an HI $\boldsymbol{Z}$ with L_h channels would approximately lead to the use of $64 \times L_h/3$ filters in each layer, leading to a very large number of parameters. This increase in the number of network parameters makes it hard to train since the amount of training data is usually very limited. Considering that the spectral channels of $\boldsymbol{Z}$ are highly correlated and contain highly redundant information, we can assume that there exists a subspace of dimension much lower than L_h which captures all the information of $\boldsymbol{Z}$ [Wei 2016]. This allows us to write $\boldsymbol{Z}$ using a low-rank representation as:

$$\boldsymbol{Z} = \boldsymbol{U}\boldsymbol{V}, \tag{3.29}$$

where $\boldsymbol{U} \in \mathbb{R}^{L_h \times l_h}$ ($l_h \ll L_h, \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}$) and $\boldsymbol{V} \in \mathbb{R}^{l_h \times M}$ are the subspace matrix and the representation coefficients, respectively. Small values of l_h correspond to data description in a low-dimensional space. Employing such dimensionality reduction in the CNN denoising engine has a core benefit. It decreases the number of filters by a ratio of l_h/L_h in each layer by removing the burden of learning information that is redundant across spectral channels.

To reduce filter volume, we use separable convolutions to further lighten the backbone architecture as in [Imamura 2019]. In particular, the core idea of separable convolution is decomposing a convolution filter with $3 \times 3 \times \texttt{Depth}$ parameters into a depth-wise filter with $3 \times 3 \times 1$ parameters and a point-wise filter with $1 \times 1 \times \texttt{Depth}$ parameters, where `Depth` is the input depth of this CNN layer. This reduces the number of parameters by a rate of $1/\texttt{Depth} + 1/(3 \times 3)$. Thus, the lightweight DnCNN contains three kinds of operators: 3×3 separable convolution layers (S-Conv), rectified linear units (ReLU) and batch normalization (BN). ReLU is the activation function while BN is used to accelerate the training speed. In the network architecture, the first layer is "S-Conv + ReLU", the hidden layer is "S-Conv + BN + ReLU" and the last layer is "S-Conv". This network architecture is illustrated in the bottom panel of FIGURE 3.1. Furthermore, we adopt the residual learning strategy in [Zhang 2017a] to predict the residual image before achieving the estimated clean image.

With these two strategies, the number of network parameters can be significantly reduced with a ratio of $(l_h/L_h) \times (1/\texttt{Depth} + 1/(3 \times 3))$, which is key to allowing the denoising engine to learn a powerful prior from a small training set.

Zero-shot training strategy: In many real-world scenarios, training data with paired noisy and clean images related to the scene of interest are not available. Moreover, using synthetic training data or images from different sites may lead to the so-called domain shift, where the model does not perform well due to differences between the statistical distribution of training and test data [Dian 2020, Wang 2020a]. Therefore, it is desirable to consider a training strategy that is *zero-shot* [Glasner 2009, Shocher 2018], that is, which is unsupervised and uses only the information of the observed noisy HI and MI pair itself for training.

Thus, we propose to leverage the information inside a single image to train the CNN denoiser. Natural images have significant information redundancy across different spatial positions and scales, which has been successfully exploited in single image restoration algorithms [Glasner 2009]. Consider the CNN-based denoiser $\texttt{CNN}(\cdot\,;\Theta)$ with network parameters Θ, and an observed noisy image $\boldsymbol{V}$ generated following the degradation model $\boldsymbol{V} = \boldsymbol{V}^{\#} + \boldsymbol{N}$, where $\boldsymbol{N}$ is i.i.d. Gaussian noise with a standard deviation σ. $\texttt{CNN}(\cdot\,;\Theta)$. To learn the CNN denoiser $\texttt{CNN}(\cdot\,;\Theta)$, we make the important assumption that the set of parameters Θ which allow $\boldsymbol{V}^{\#}$ to be recovered from $\boldsymbol{V}$, are the same as those which allow $\texttt{CNN}(\cdot\,;\Theta)$ to recover $\boldsymbol{V}$ from $\boldsymbol{V} + \boldsymbol{N}$. This assumption has been used to learn image-adapted CNNs for super-resolution in [Shocher 2018]. It allow us to train the denoising engine $\texttt{CNN}(\cdot\,;\Theta)$ using the image pair $(\boldsymbol{V} + \boldsymbol{N}, \boldsymbol{V})$ by minimizing the following ℓ_1-norm loss function:

$$\ell(\Theta) = \|\texttt{CNN}(\boldsymbol{V} + \boldsymbol{N}; \Theta) - \boldsymbol{V}\|_1 \,. \tag{3.30}$$

Algorithm 3 The Proposed CNN-based denoising engine.

Input: Noisy image $\boldsymbol{Z}$ and subspace dimension l_h.

Output: Denoised image $\mathcal{D}(\boldsymbol{Z})$.

Find $\boldsymbol{U}$ and $\boldsymbol{V}$ in (3.29) using the (truncated) SVD of $\boldsymbol{Z}$.

Optimize Θ by minimizing (3.30) with back-propagation.

Denoise $\boldsymbol{V}$ with Θ as $\texttt{CNN}(\boldsymbol{V};\Theta)$.

Transform $\texttt{CNN}(\boldsymbol{V};\Theta)$ to $\mathcal{D}(\boldsymbol{Z}) = \boldsymbol{U}\,\texttt{CNN}(\boldsymbol{V};\Theta)$.

Note that the noisy-clean image pair $(\boldsymbol{V}+\boldsymbol{N},\boldsymbol{V})$ is generated by adding Gaussian noise with standard deviation σ to the observation $\boldsymbol{V}$. We adopted the method described in [Donoho 1994] to estimate σ in each channel of $\boldsymbol{V}$. In (3.30), the ℓ_1-norm is used as a loss that is more robust to noise than the ℓ_2-norm, found to provide better performance in image restoration in the literature [Zhao 2016, Wang 2021].

The procedure for learning the proposed CNN-based denoising engine is summarized in Algorithm 3. Note that the training procedure considers the entire image, $\boldsymbol{V}$. However, for large images, other learning objectives that decompose the image into different patches or across multiple scales can provide ways to parallelize the training procedure, which might reduce the execution times.

Image-specific prior learning: Since there exist some inter-image variations between $\boldsymbol{V}_h$ and $\boldsymbol{V}_m$, we considered to train two independent denoising engines $\texttt{CNN}(\,\cdot\,;\Theta_h)$ and $\texttt{CNN}(\,\cdot\,;\Theta_m)$ to denoise $\boldsymbol{Z}_h$ and $\boldsymbol{Z}_m$, respectively. This leads to different denoising engines, which can be expressed by substituting $\mathcal{D}$ by $\mathcal{D}_h$ in (3.27), and by $\mathcal{D}_m$ in (3.28).

In general, the equivalent noise levels of $\boldsymbol{Z}_h$ and $\boldsymbol{Z}_m$ decrease over the block coordinate descent (BCD) iterations since the reconstructed images get closer to the ground truth. Thus, $\texttt{CNN}(\,\cdot\,;\Theta_h)$ and $\texttt{CNN}(\,\cdot\,;\Theta_m)$ should have the ability to tackle multiple noise levels. To address this issue, we propose a strategy that adaptively updates network parameters Θ_h and Θ_m to learn an image-specific prior at each BCD iteration. This is performed by re-training $\texttt{CNN}(\,\cdot\,;\Theta_h)$ and $\texttt{CNN}(\,\cdot\,;\Theta_m)$ to denoise the estimates of the HRIs at the current BCD iteration. To make the algorithm faster, we consider training $\texttt{CNN}(\,\cdot\,;\Theta_h)$ and $\texttt{CNN}(\,\cdot\,;\Theta_m)$ in the first BCD iteration and then fine-tune them in all the remaining iterations.

Overall, after overcoming the discussed challenges with the above strategies, the denoising engine in Algorithm 3 is incorporated into the model-based optimization procedure described in Subsection 3.3.3. The overall DIFIV strategy is described in Algorithm 4.

3.4 Experiments

In this section, the effectiveness of the proposed DIFIV method is illustrated through numerical experiments considering two categories of real data, i.e., observed images with moderate and significant inter-image variability. The results provided by

Algorithm 4 Deep Hyperspectral and Multispectral Image Fusion with Inter-image Variability (DIFIV).

Input: $\boldsymbol{Y}_h, \boldsymbol{Y}_m, \boldsymbol{F}, \boldsymbol{D}, \boldsymbol{R}$, paramters $p, \eta_p, \eta_h, \eta_m, \rho$.
Output: The estimated high-resolution images $\widehat{\boldsymbol{X}}_h, \widehat{\boldsymbol{X}}_m$.
Interpolate $\boldsymbol{Y}_h$ and $\boldsymbol{Y}_m$ as $\widetilde{\boldsymbol{Y}}_h$ and $\widetilde{\boldsymbol{Y}}_m$, respectively.
Initialize $\boldsymbol{X}_h = \boldsymbol{Z}_h = \widetilde{\boldsymbol{Y}}_h$ and $\boldsymbol{X}_m = \boldsymbol{Z}_m = \widetilde{\boldsymbol{Y}}_m$.
Initialize $\boldsymbol{W}$ using (3.12).
while stopping criteria are not met **do**
  Calculate $\boldsymbol{X}_h$ by solving (3.20) via CG algorithm.
  Calculate $\boldsymbol{X}_m$ by solving (3.22) via CG algorithm.
  Update $\boldsymbol{W}$ using (3.12).
  Learn deep priors via denoising $\boldsymbol{Z}_h$ with Algorithm 3.
  Update $\boldsymbol{Z}_h$ via (3.27).
  Learn deep priors via denoising $\boldsymbol{Z}_m$ with Algorithm 3.
  Update $\boldsymbol{Z}_m$ via (3.28).

the DIFIV method are compared with other state-of-the-art hyperspectral and multispectral image fusion methods from both quantitative and qualitative perspectives. The code is made available at `https://github.com/xiuheng-wang/DIFIV_release`.

3.4.1 Baselines and experimental setup

We compared our method to nine other techniques, namely: the matrix factorization-based methods HySure [Simões 2015] and CNMF [Yokoya 2012], tensor-based image fusion methods STEREO [Kanatsoulis 2018] and SCOTT [Prévost 2020], the multiresolution analysis-based GLPHS algorithm [Aiazzi 2006], and the unsupervised deep learning based algorithm PAR [Wei 2020b]. We also considered approaches accounting for inter-image variability, including FuVar [Borsoi 2020], GSFus [Fu 2021] and CB-STAR [Borsoi 2021e]. In this study, three real data sets with moderate variability, namely, the Ivanpah Playa, the Lake Isabella and the Lookwood, and three real data sets with significant variability, namely, the Lake Tahoe A and B, and the Kern River, were used to evaluate the performance of each method. These data sets contained one reference HRI $\boldsymbol{X}_h$ and an MI $\boldsymbol{Y}_m$ acquired by the AVIRIS and the Sentinel-2A instruments, respectively, with a pixel of 20m resolution [Borsoi 2020]. The HI and MI contain $L_h = 173$ and $L_m = 10$ bands, respectively.

For all acquired HRIs $\boldsymbol{X}_h$, which have the same spatial resolution as the MIs $\boldsymbol{Y}_m$, a pre-processing procedure as described in [Simões 2015] was performed. Specifically, spectral bands that were overly noisy or corresponded to water absorption spectral regions were removed manually, and then all bands of HRIs $\boldsymbol{X}_h$ and MIs $\boldsymbol{Y}_m$ were normalized such that the 0.999 intensity quantile corresponded to the value of 1. Moreover, all HRIs $\boldsymbol{X}_h$ were denoised using the approach described in [Roger 1996]. To illustrate the existence of the inter-image variability in the considered datasets, we computed the average absolute difference images $\frac{1}{L_m}\sum_{\ell=1}^{L_m} |\boldsymbol{Y}_m(\ell,:) - \boldsymbol{R}(\ell,:)\boldsymbol{X}_h|$

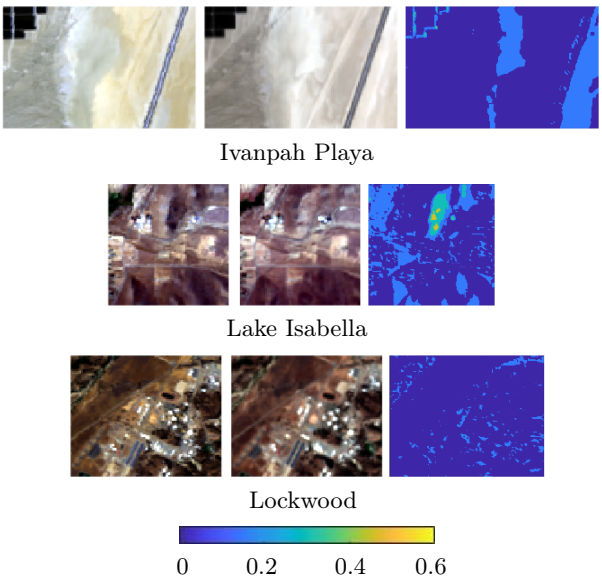

FIGURE 3.2 – Visible representation of the hyperspectral (left panels) and multispectral images (middle panels) with moderate variability used in the experiments and their inter-image change maps (right panels).

(where the modulus operation $|\cdot|$ is applied elementwise), and displayed them in Figures 3.2 and 3.3. The observed HIs $\boldsymbol{Y}_h$ were generated according to (3.1), where $\boldsymbol{F}$ was an 8×8 Gaussian blurring operator with standard deviation 4 and $\boldsymbol{D}$ a downsampling operator with the scaling factor 4. The SRF $\boldsymbol{R}$ was acquired from calibration measurements of the Sentinel-2A instrument and known a priori [1]. For all experiments, Gaussian noise was added to both HIs and MIs to obtain a signal-to-noise ratio (SNR) of 35 dB. To set up all baselines, we used the code provided by the authors and tuned all parameters to achieve the best fusion performance.

We implemented the proposed DIFIV method with the CNN-based denoising engine using the PyTorch framework. The dimension of subspace l_h was set to 5 and the number of network layers was set to 8, the first and hidden layers contained $l_h \times 4$ S-Conv operators while the last layer was composed by l_h S-Conv operators. The Adam optimizer [Kingma 2014a] with an initial learning rate 0.0002 was used to minimize the loss function in (2.25). The numbers of iterations of our DIFIV method (Algorithm 4) and fix-point updates in (3.27) and (3.28) were set to 20, 1 and 1, which were sufficient to ensure convergence. The weights were initialized with the method in [He 2015], trained for 10000 epochs in the first iteration, and fine-tuned for 2000 epochs in the remaining iterations. We set $p = 1.5, \eta_p = 0.01$ and $\eta_m = \eta_n = 0.1$ for the data with moderate variability. For the data with significant variability, we set $p = 1.8, \eta_p = 0.002$ and $\eta_m = \eta_n = 0.01$. For the other parameters, we set $\rho = 0.1$ and $\varepsilon = 10^{-6}$. We used empirical methods to determine the above parameters. Note the data does not need to be divided for cross-validation since our

1. The SRF can be downloaded online at https://github.com/xiuheng-wang/DIFIV_release.

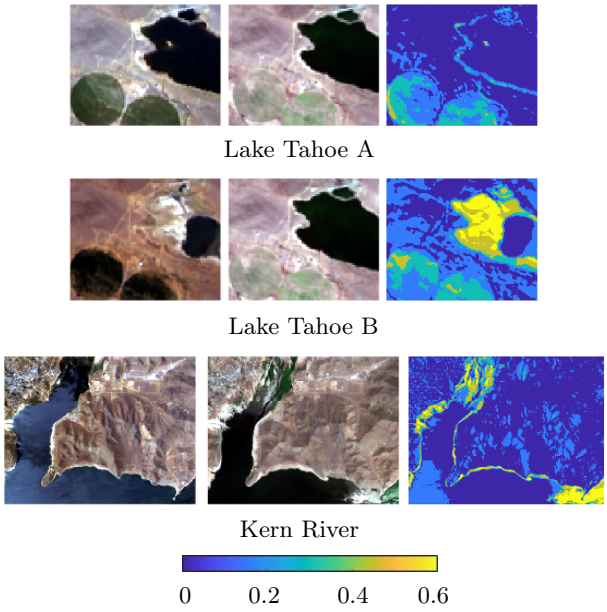

FIGURE 3.3 – Visible representation of the hyperspectral (left panels) and multispectral images (middle panels) with significant variability used in the experiments and their inter-image changes maps (right panels).

CNN-based denoising engine is unsupervised. In the following, the performance of the methods is compared via $\boldsymbol{X}_h$ only since the HRI corresponding to $\boldsymbol{Y}_m$ was not available in the experiments.

3.4.2 Quality measure and visual assessment

Four quality metrics were considered to evaluate the quality of the fusion result $\widehat{\boldsymbol{X}}_h$ compared to the ground truth $\boldsymbol{X}_h$. The first one is the peak signal to noise ratio (PSNR):

$$\text{PSNR} = \frac{1}{L_h}\sum_{\ell=1}^{L_h} 10\log_{10}\left(\frac{M\ \max(\boldsymbol{X}_h(\ell,:))^2}{\left\|\widehat{\boldsymbol{X}}_h(\ell,:)-\boldsymbol{X}_h(\ell,:)\right\|^2}\right),$$

where $\boldsymbol{X}_h(\ell,:)$ and $\widehat{\boldsymbol{X}}_h(\ell,:)$ represent the ℓ-th channel of $\boldsymbol{X}_h$ and $\widehat{\boldsymbol{X}}_h$, respectively.

The second metric is the Spectral Angle Mapper (SAM):

$$\text{SAM} = \frac{1}{M}\sum_{m=1}^{M} \arccos\left(\frac{\widehat{\boldsymbol{X}}_h^\top(:,m)\boldsymbol{X}_h(:,m)}{\left\|\widehat{\boldsymbol{X}}_h(:,m)\right\|\left\|\boldsymbol{X}_h(:,m)\right\|}\right),$$

where $\boldsymbol{X}_h(:,m)$ and $\widehat{\boldsymbol{X}}_h(:,m)$ denote the m-th pixel of $\boldsymbol{X}_h$ and $\widehat{\boldsymbol{X}}_h$, respectively.

The third metric is the ERGAS [Wald 2000], which provides a global statistical

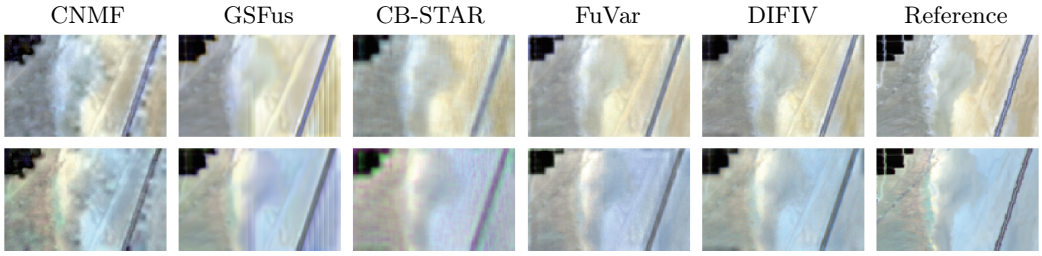

FIGURE 3.4 – Visible (top) and infrared (bottom) representation for the estimated and true versions of the Ivanpah Playa HI.

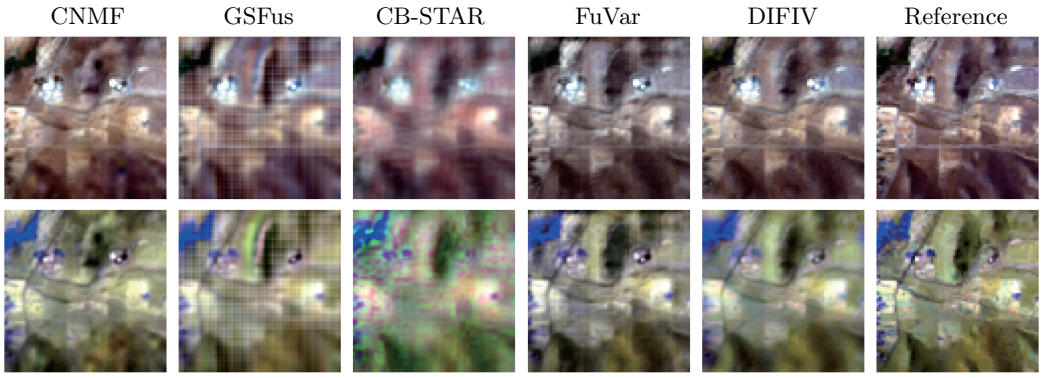

FIGURE 3.5 – Visible (top) and infrared (bottom) representation for the estimated and true versions of the Lake Isabella HI.

measure of the fused image quality, defined as:

$$\text{ERGAS} = \frac{M}{N}\sqrt{\frac{10^4}{L_h}\sum_{\ell=1}^{L_h}\frac{\left\|\widehat{\boldsymbol{X}}_h(\ell,:)-\boldsymbol{X}_h(\ell,:)\right\|^2}{\text{mean}(\widehat{\boldsymbol{X}}_h(\ell,:))^2}}\,.$$

This metric is the average of the UIQI [Wang 2002] across bands. It evaluates image distortions including correlation loss and luminance and contrast distortions, and tends to 1 as $\widehat{\boldsymbol{X}}_h$ tends to $\boldsymbol{X}_h$.

For the visual assessment of the reconstructed images, we displayed color images at the visual spectrum (with band image at the wavelength 0.66, 0.56 and 0.45 μm as red, green and blue channels) and false color images at the infrared spectrum (with band image at the wavelength 2.20, 1.50 and 0.80 μm as red, green and blue channels). Due to space limitations, in the following, we only display the results of the five methods with the best quantitative performances, namely, CNMF, FuVar, GSFus, CB-STAR and DIFIV. Note that the last four algorithms account for inter-image variability.

3.4.3 Category 1: Moderate variability

In this category, we evaluated the methods using HI and MI pairs with moderate variability, including Ivanpah Playa, Lake Isabella and Lockwood.

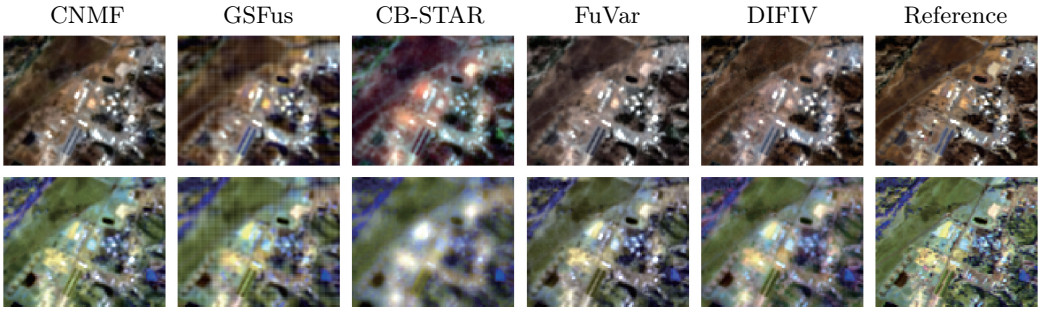

FIGURE 3.6 – Visible (top) and infrared (bottom) representation for the estimated and true versions of the Lockwood HI.

The first image pair considered in this category was acquired over the area surrounding Ivanpah Playa with a resolution of 80×128 pixels. The second pair of images, with 80×80 pixels, was captured over the Lake Isabella region, while the third pair of images containing 80×100 pixels was acquired near Lockwood. The visualizations of these three image pairs and their inter-image variability are shown in FIGURE 3.2. In this category, the HI and MI look visually similar, which is typical when small differences between acquisition dates are considered (which is the case for the Lake Isabella and Lockwood images). Nevertheless, slight variations still exist, as can be seen in the overall color hue of the Ivanpah Playa and Lockwood images, and in the up part of the Lake Isabella image.

TABLE 3.1 – Quantitative Results on the Ivanpah Playa HI

Algorithm	SAM	ERGAS	PSNR	UIQI
HySure	2.262	2.639	21.923	0.511
CNMF	1.532	2.258	23.729	0.73
GLPHS	2.924	3.139	20.949	0.508
STEREO	29.173	1,643.756	17.744	0.49
SCOTT	41.025	618.314	9.388	0.307
PAR	3.506	2.26	24.011	0.752
FuVar	1.469	1.804	25.622	0.868
GSFus	1.72	1.497	27.264	0.874
CB-STAR	1.91	1.517	27.506	0.875
DIFIV	**1.358**	**1.335**	**28.283**	**0.903**

SAM, PSNR, ERGAS and UIQI metrics for all methods are reported in tables 3.1 to 3.3. As shown in tables 3.1 and 3.2, DIFIV outperforms all competing methods in all metrics for the Ivanpah Playa and Lake Isabella images. Moreover, it can be seen in TABLE 3.3 that DIFIV achieves overall the best results for the Lookwood data, surpassing the other methods in all metrics except for SAM, where CNMF yields the best results for this metric. Figures 3.4 to 3.6 illustrate the color and

Table 3.2 – Quantitative Results on the Lake Isabella HI

Algorithm	SAM	ERGAS	PSNR	UIQI
HySure	3.021	5.363	19.905	0.637
CNMF	2.206	3.414	25.611	0.792
GLPHS	2.755	3.572	25.207	0.793
STEREO	27.859	2,145.707	19.221	0.573
SCOTT	26.281	282.097	8.453	0.076
PAR	7.482	4.044	25.454	0.805
FuVar	2.487	3.234	27.213	0.899
GSFus	2.759	3.787	26.448	0.864
CB-STAR	3.263	3.406	26.556	0.864
DIFIV	**2.114**	**2.323**	**29.186**	**0.923**

Table 3.3 – Quantitative Results on the Lockwood HI

Algorithm	SAM	ERGAS	PSNR	UIQI
HySure	3.384	4.384	22.678	0.881
CNMF	**3.243**	3.349	26.469	0.857
GLPHS	3.706	3.971	24.704	0.781
STEREO	28.185	883.508	21.079	0.639
SCOTT	20.109	204.538	9.273	0.094
PAR	6.61	4.433	23.634	0.754
FuVar	3.518	3.345	26.509	0.874
GSFus	3.331	3.332	26.329	0.87
CB-STAR	4.137	3.867	25.535	0.805
DIFIV	3.394	**2.934**	**27.307**	**0.885**

false color visualization of the fusion results of several algorithms. Visually, DIFIV provides the best results in recovering details and spatial reconstructions closest to the ground truth at both the visual and infrared spectra. Specifically, CNMF and GSFus introduce artifacts and fail to recover many details while CB-STAR produces blurry effects and color aberrations. FuVar and DIFIV give similar visual effects, this demonstrates the efficiency of DIFIV in recovering the spatial information of the latent HRIs is comparable to FuVar in this category.

3.4.4 Category 2: Significant variability

This category evaluates the performance of the different methods when there is significant inter-image variability. We consider two image pairs acquired over the Lake Tahoe area at different time instant, namely, Lake Tahoe A and B. Besides, an image pair captured over the Kern River scene, which comprises a larger spatial

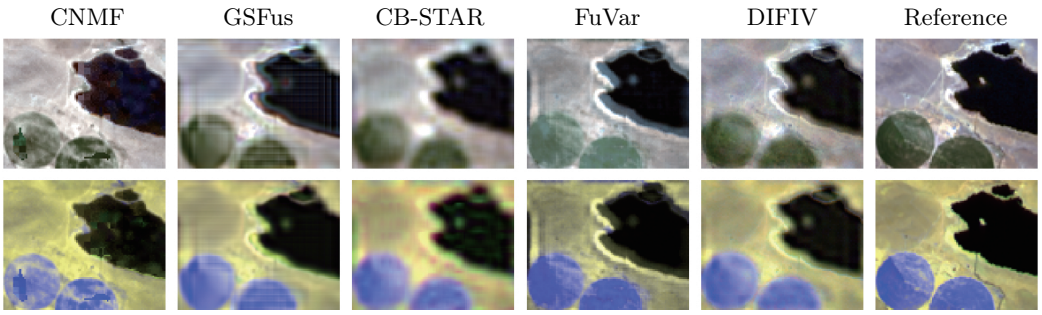

FIGURE 3.7 – Visible (top) and infrared (bottom) representation for the estimated and true versions of the Lake Tahoe A HI.

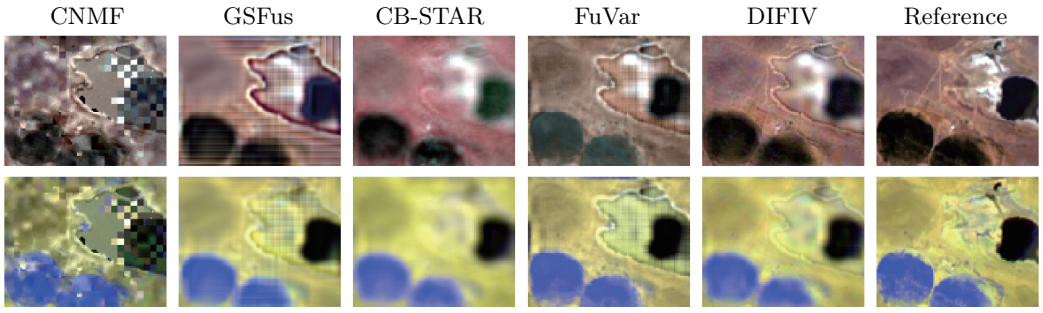

FIGURE 3.8 – Visible (top) and infrared (bottom) representation for the estimated and true versions of the Lake Tahoe B HI.

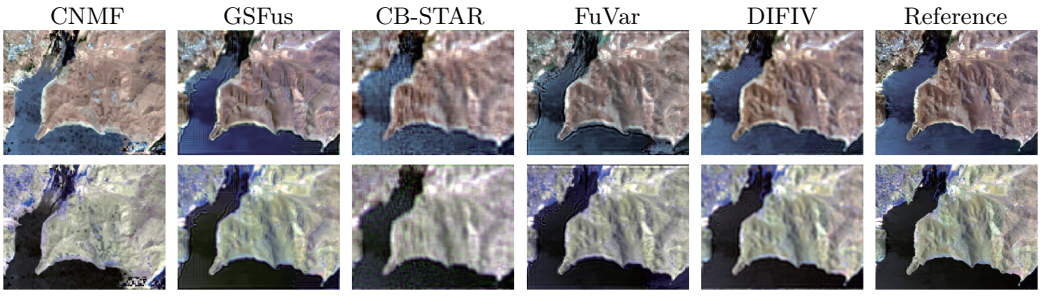

FIGURE 3.9 – Visible (top) and infrared (bottom) representation for the estimated and true versions of the Kern River HI.

area, was also considered.

The two Lake Tahoe image pairs contain 100×80 pixels, while the Kern River image pair contains 260×340 pixels. The visualization of these HIs and MIs and their corresponding inter-image changes maps can be seen in FIGURE 3.3. Significant variability between the HI and MI can be easily verified in these cases. For the two Lake Tahoe image pairs in this category, the color hue of the ground and the crop circles is quite different. Moreover, an island on the lake is not visible in the MI of Lake Tahoe A. For Lake Tahoe B, the lake in the MI is much larger than that in the HI. For the Kern River image pair, the river in the MI is narrower, has an upstream deposit, and shows a darker color in the water area.

The quantitative metrics are reported in tables 3.4, 3.5 and 3.6. As shown in TABLE 3.4, DIFIV obtains the best results for most metrics for Lake Tahoe A and only performs slightly worse in terms of SAM compared to GSFus. It can be observed in TABLE 3.5 and 3.6 that the performance of DIFIV for Lake Tahoe B and Kern River exceeds those of the competing methods for all metrics. A visual illustration of the fusion results for Lake Tahoe A and B in color and false color is displayed in FIGURE 3.7 and FIGURE 3.8. FIGURE 3.9 shows the visualization of the fusion results for the Kern River dataset. It can be seen that DIFIV reconstructs more details and produces a color hue closer to the reference images at both visual and infrared spectral ranges. In particular, CNMF produced many artifacts and lost some details. GSFus and FuVar generate results with blockiness and ghosting effects while the results of CB-STAR are blurry and have some color distortions. This demonstrates the superiority of DIFIV in recovering the latent HRIs when significant variability exists.

TABLE 3.4 – Quantitative results on Lake Tahoe A HI

Algorithm	SAM	ERGAS	PSNR	UIQI
HySure	10.643	7.775	16.531	0.655
CNMF	12.371	7.514	18.102	0.676
GLPHS	10.803	7.206	18.303	0.701
STEREO	30.605	2,541.149	15.991	0.575
SCOTT	42.839	457.101	9.243	0.215
PAR	15.886	6.065	20.579	0.811
FuVar	8.373	6.545	19.258	0.78
GSFus	**6.628**	4.376	22.537	0.883
CB-STAR	7.548	3.769	24.165	0.917
DIFIV	6.737	**3.706**	**24.174**	**0.922**

TABLE 3.5 – Quantitative results on Lake Tahoe B HI

Algorithm	SAM	ERGAS	PSNR	UIQI
HySure	13.458	12.042	11.913	0.235
CNMF	7.954	7.289	16.387	0.428
GLPHS	6.662	4.786	19.824	0.665
STEREO	29.877	7,936.808	15.208	0.463
SCOTT	42.427	491.817	7.504	0.136
PAR	11.787	6.21	21.405	0.728
FuVar	4.688	3.729	21.86	0.79
GSFus	4.182	3.16	23.425	0.826
CB-STAR	3.95	2.597	25.221	0.881
DIFIV	**3.265**	**2.396**	**25.834**	**0.899**

TABLE 3.6 – Quantitative results on Kern River HI

Algorithm	SAM	ERGAS	PSNR	UIQI
HySure	9.094	8.933	21.717	0.442
CNMF	5.851	8.471	22.853	0.356
GLPHS	8.231	7.279	24.19	0.492
STEREO	30.337	636.136	22.568	0.45
SCOTT	27.652	220.14	13.239	0.045
PAR	11.695	6.742	28	0.739
FuVar	4.654	5.144	28.335	0.797
GSFus	5.037	4.243	29.404	0.785
CB-STAR	5.298	5.004	28.884	0.729
DIFIV	**3.412**	**3.734**	**31.506**	**0.852**

3.4.5 Parameter sensitivity and computational cost

In this subsection, we study the sensitivity of DIFIV to the choice of values for regularization parameters η_p, η_h, η_m. Considering the Ivanpah Playa scene as an example, we varied each parameter individually while keeping the remaining ones fixed at the values described in Subsection 3.4.1. The PSNR values of the fusion results as a function of the ratio $\log_{10}(\eta/\eta_{opt})$ are shown in FIGURE 3.10, where η_{opt} is the empirically selected value of the corresponding parameters. The PSNR values of two selected competing methods (CB-STAR and GSFus) are also shown for reference. It can be observed that varying parameters of DIFIV even by various orders of magnitude only leads to moderate variations of PSNR values, which are consistently higher than that of the competing methods. Moreover, the parameters of GSFus and CB-STAR were adjusted to provide the best performance in each example, and their performance would likewise degrade if their parameters move away

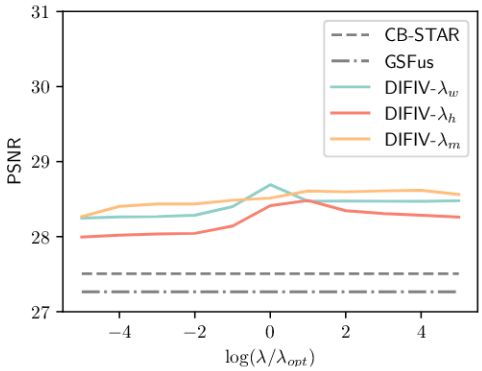

FIGURE 3.10 – Sensitivity of the proposed DIFIV method with respect to regularization parameters η_p, η_h, η_m.

TABLE 3.7 – Execution times of the algorithms that consider inter-image variability (in seconds)

	FuVar	GSFus	CB-STAR	DIFIV
Ivanpah Playa	354.6	31.4	11.1	2963.4
Lake Isabella	199.3	17.7	8.8	2928.7
Lockwood	228.8	23.2	30.1	2954.0
Lake Tahoe A	679.5	23.1	7.8	2178.4
Lake Tahoe B	718.9	22.2	7.6	2143.5
Kern River	1762.0	307.3	96.0	5908.4

from their optimal values, as discussed in the original works [Fu 2021, Borsoi 2021e]. This indicates the performance of DIFIV is not overly sensitive to the choice of regularization parameters.

This experiment aims to compare the computational cost of the algorithms accounting for inter-image variability. DIFIV was implemented using Python while the remaining methods were implemented using MATLAB. We conducted all the experiments on a computer with an Intel Core i7-10700 CPU, 32-GB random access memory and an NVIDIA Quadro P2200 GPU. The execution times of the algorithms for all the tested image pairs are shown in TABLE 3.7. It can be seen that the computation times of DIFIV are substantially higher than those of the competing methods, which comes as a compromise for its superior image fusion quality results. Nevertheless, the computation times of DIFIV scale reasonably with the image sizes; for instance, comparing the results for the Lake Isabella and Kern River images, we see that an increase of about ten times in the number of pixels in the image leads to an increase of about two times in the computation times. The development of computationally efficient extensions to the DIFIV method will be investigated in future work.

3.5 Conclusion

This chapter presented an unsupervised deep learning-based HMIF method accounting for inter-image variability. We first formulated a new imaging model considering both the joint as well as the image-specific priors related to the two latent HRIs. The inter-image variations were modeled using a hyper-Laplacian distribution, while the image-specific priors of the latent HRIs were defined implicitly by deep denoising engines. An iteratively reweighted scheme was then investigated to solve the non-convex cost function and tackle the joint image prior term. The optimization problem was solved using a variable splitting strategy, and the deep image priors were implemented by means of CNN-based denoising operations. A lightweight, image-specific CNN-based denoiser with a zero-shot training strategy was designed. The network parameters were iteratively updated during the optimization procedure in order to adapt to variations in the statistical properties of the estimated HRIs as the method converged. The proposed method achieved superior experimental performance in the presence of both moderate and significant inter-image variability when compared to state-of-the-art approaches.

Part II

Joint Modeling and Learning Approaches: Changepoint Detection

Context

The challenges associated with the appropriate design of methods incorporating machine learning and physical modeling are also present in other applications, particularly in detecting changepoints in time series. Moreover, the CPD task also involves additional challenges, including the need to tackle streaming data, real-time processing, and low memory usage. A typical physics-based modeling of the changes in data distributions involves estimating changes in the parameters of a family of probability density functions (PDFs) representing the data.

In Part II, we investigate the integration of advanced machine learning techniques, including neural networks and Riemannian stochastic optimization, with the learning objectives derived by taking inspiration from physics-based models. Without relying on any prior knowledge regarding the underlying data distribution, we apply joint modeling and learning approaches to address different challenges of the non-parametric CPD problem in streaming Euclidean data, data lying on Riemannian manifolds, and manifold-valued data over graphs.

We begin by leveraging deep learning to detect changepoints in streaming Euclidean data in Chapter 4. We present a novel approach for joint modeling and learning, employing an online neural network-based method to estimate the density ratio between current and reference windows within a data stream. Our method utilizes a variational continual learning framework to facilitate online training of the neural network while preserving information gleaned from previous data instances. As a result, we establish a statistically-grounded, fully non-parametric framework for detecting changepoints within streaming data.

Chapter 5 moves on to introduce the CPD on Riemannian manifolds, which is a more challenging setting compared to the Euclidean case due to the nonlinear geometry involved and the lack of a vector space structure. We present an algorithm that integrates modeling and learning for non-parametric online CPD in data streams with manifold-valued measurements. This algorithm tracks the generalized Karcher mean of the data, calculated through stochastic Riemannian optimization. Theoretical bounds on the performance metrics of detection and false alarm rates are derived, leveraging a novel result concerning the non-asymptotic convergence of stochastic Riemannian gradient descent. Furthermore, we demonstrate the efficacy of our algorithm on two distinct manifolds.

Finally, in Chapter 6 we extend the CPD algorithm presented in Chapter 5 to process the streaming manifold-valued data over graphs. Our approach integrates a local test statistic at each node to handle the inherent geometry of data lying on a Riemannian manifold, along with a fully distributed graph filter that incorporates network topology information. This integration leads to notable improvements in the detection of changes occurring within unknown network communities.

Chapter 4

CPD with neural online density-ratio estimation

Contents

4.1 Introduction

Numerous approaches have been proposed to perform CPD. Depending on whether prior information on data distributions is available, recent CPD approaches can be roughly divided into parametric and non-parametric strategies. Parametric ones rely on model assumptions describing the probability density function (PDF) of the data before and after an abrupt change. Examples of parametric CPD strategies include the cumulative sum (CUSUM) [Inclan 1994], the generalized likelihood ratio test (GLRT) [Gustafsson 1996], and subspace identification (SI) [Kawahara 2007]. The CUSUM algorithm [Inclan 1994] assumes that the parameters changing are known and typically assumes that the underlying data follows a specific statistical distribution. The GLRT method [Gustafsson 1996] assumes that observations are driven by a linear state-space model. By explicitly considering a noise factor in a linear state-space model, the SI approach [Kawahara 2007] detects changes using distances between the subspaces spanned by two sequence windows.

Parametric CPD methods operate well when all the assumptions on the problem at hand are met. Nevertheless, deriving a model that accurately describes the data is usually intractable and makes parametric approaches sensitive to modeling errors [Truong 2020]. Non-parametric CPD methods have been introduced to address this issue. These approaches make weaker assumptions about the data and include,

for instance, the use of empirical estimation of the cumulative data distribution, or the deviation of a kernel embedding of the data from its mean [Truong 2020]. A non-parametric strategy of particular interest is the use of density-ratio estimation. Although the distribution of the pre- and post-change data can be hard to estimate, only their ratio – which can be easier to estimate – is necessary to perform CPD [Sugiyama 2012]. Several CPD methods based on density-ratio estimation have been proposed in the literature. Examples include the Kullback-Leibler (KL) divergence based importance estimation procedure (KLIEP) [Sugiyama 2007], the unconstrained least squares importance fitting (uLSIF) and the relative uLSIF [Liu 2013].

Unlike these offline methods that detect changes in a dataset collected a priori, online CPD algorithms process streaming data iteratively in an adaptive fashion. The method in [Chen 2019] considers the k-nearest neighbors (kNN) algorithm to tackle online CPD by extending SI techniques in non-linear subspaces. In [Keriven 2020], the moving average-based algorithm NEWMA is introduced to monitor the mean of the process in a feature space. An online version of the relative uLSIF-based method NOUGAT is designed in [Ferrari 2022] to detect changepoints by learning density-ratios with the kernel trick. Another important branch of non-parametric online CPD is based on virtual classifiers (VC) [Desobry 2005, Yamada 2013]. These methods train a binary classifier with pseudo labels to learn density-ratio over past and future data, and consider the separability of data to detect changepoints. An online Bayesian approach using a latent class model for the data whose number of classes can increase over time was proposed in [Moreno-Muñoz 2020]. However, Bayesian methods can have high complexity when compared to approaches such as [Keriven 2020, Ferrari 2022].

Recently, deep learning has become a popular framework for addressing a variety of signal processing tasks. Several works considered deep learning for CPD. In [De Ryck 2021], an autoencoder is used to learn a time-invariant representation of the data which is more amenable for CPD. Neural networks are used for density-ratio estimation in [Khan 2019]. However, both approaches do not operate online. Another method based on the reconstruction error of an autoencoder is proposed in [Gupta 2022] with real-time preprocessing. However, it relies on strong assumptions about the nature of the changes. A related approach using an autoencoder based on recurrent neural networks was proposed in [Atashgahi 2022]. Current deep learning and density-ratio learning CPD algorithms are still limited in combining flexibility with the ability to retain knowledge from past data while maintaining a low complexity. Continual learning [De Lange 2021, Nguyen 2018] has the ability to adapt to recent data while at the same time retaining past knowledge. This made it successful in various online learning tasks.

In this chapter, a new online CPD strategy based on neural density-ratio estimation and continual learning is proposed. First, density-ratio estimation is represented as a binary classification problem over two sliding (reference and test) data windows. This allows us to leverage state-of-the-art probabilistic classification neural networks to perform CPD in a non-parametric manner. Moreover, to obtain an adaptive detection strategy that leverages past information while operating online,

a variational continual learning objective is devised to train the neural network classifier in a Bayesian framework. Specifically, the statistical distribution of the network parameters at each time step is used as a prior for the next classification objective in a regularization-based framework. This allows the trade-off between temporal smoothness and fast adaptation to be controlled using a single regularization parameter. Experimental results with both synthetic and real data show the effectiveness of the proposed strategy.

4.2 Proposed method

The basic idea of the proposed CPD strategy consists of estimating changepoints by evaluating the density-ratio between the PDFs of the data over a reference and a test window, given by:

$$r(\boldsymbol{x}_t) = \frac{p_{\text{test}}(\boldsymbol{x}_t)}{p_{\text{ref}}(\boldsymbol{x}_t)}\,, \tag{4.1}$$

with $p_{\text{test}}(\boldsymbol{x})$ the data PDF over the test window with N samples:

$$\mathcal{X}_t = \left\{\boldsymbol{x}_{t-N+1}, \ldots, \boldsymbol{x}_{t-1}, \boldsymbol{x}_t\right\}, \tag{4.2}$$

and $p_{\text{ref}}(\boldsymbol{x})$ the data PDF over the reference window with N' samples:

$$\mathcal{X}'_t = \left\{\boldsymbol{x}_{t-N-N'+1}, \ldots, \boldsymbol{x}_{t-N-1}, \boldsymbol{x}_{t-N}\right\}. \tag{4.3}$$

Our objective is to estimate the density-ratio $r(\boldsymbol{x}_t)$, at each time $t \in \mathbb{N}$, given only the data $\boldsymbol{x}_t$ observed sequentially over windows $\mathcal{X}_t$ and $\mathcal{X}'_t$. To this end, we will consider two steps: first, a probabilistic classification-based approach is introduced to estimate the density-ratio; afterwards, we propose to use a Bayesian continual learning strategy in order to learn the classifier online.

4.2.1 Neural Online Density-ratio Estimator

Without additional knowledge about $p_{\text{test}}(\boldsymbol{x}_t)$ and $p_{\text{ref}}(\boldsymbol{x}_t)$, computing these PDFs can be intractable, and a non-parametric estimation of $r(\boldsymbol{x}_t)$ becomes more desirable. Within this context, online kernel [Ferrari 2022] or offline deep learning [Khan 2019] strategies have been proposed to estimate density-ratio with the design of specific learning objectives, such as LSIF and RuLSIF [Liu 2013]. An important property of the density-ratio is that it can be related to probabilistic binary classification, allowing us to leverage state-of-the-art classification methods to address this problem [Cranmer 2015, Menon 2016, Durkan 2020]. First, let us annotate the samples in datasets $\mathcal{X}'_t$ and $\mathcal{X}_t$ with pseudo labels 0 and 1, respectively. This way, considering the labels to be a random variable $y_t \in \{0, 1\}$, we can express the distributions $p_{\text{test}}(\boldsymbol{x}_t)$ and $p_{\text{ref}}(\boldsymbol{x}_t)$ in the form of a single conditional PDF:

$$p_{\text{test}}(\boldsymbol{x}_t) = p(\boldsymbol{x}_t|y_t = 1)\,, \tag{4.4}$$

$$p_{\text{ref}}(\boldsymbol{x}_t) = p(\boldsymbol{x}_t|y_t = 0)\,. \tag{4.5}$$

Using Bayes' rule and the above definition and assuming the two classes with equal a priori marginal class probabilities, according to Theorem 1 in [Cranmer 2015], equation (4.1) can be written as:

$$r(\boldsymbol{x}_t) = \frac{p(\boldsymbol{x}_t|y_t=1)}{p(\boldsymbol{x}_t|y_t=0)} = \frac{p(y_t=1|\boldsymbol{x}_t)}{p(y_t=0|\boldsymbol{x}_t)} = \frac{p(y_t=1|\boldsymbol{x}_t)}{1-p(y_t=1|\boldsymbol{x}_t)}\,. \tag{4.6}$$

In this way, the density-ratio between $p_{\text{test}}(\boldsymbol{x}_t)$ and $p_{\text{ref}}(\boldsymbol{x}_t)$ can be recovered by the optimal binary classifier $p(y_t|\boldsymbol{x}_t)$ that distinguishes between samples from these two distributions.

By concatenating the two datasets corresponding to the reference and test windows as:

$$\mathcal{D}_t = \left\{(\boldsymbol{x}_t, y_t=0) : \boldsymbol{x}_t \in \mathcal{X}'_t\right\} \bigcup \left\{(\boldsymbol{x}_t, y_t=1) : \boldsymbol{x}_t \in \mathcal{X}_t\right\},$$

which leads the CPD problem to be formulated as learning a binary classifier at each time $t \in \mathbb{N}$ based on the training data $\mathcal{D}_t$, also called as virtual classifier [Desobry 2005, Yamada 2013]. We denote this learnable classifier by $p(y_t|\boldsymbol{x}_t, \boldsymbol{\phi}_t)$, where $\boldsymbol{\phi}_t$ denotes a vector containing its parameters at each time instant. It has been shown in [Menon 2016] that a wide range of losses used in binary classification are suitable to perform density-ratio estimation.

It is popular to parameterize this classifier as $p(y_t=1|\boldsymbol{x}_t) = \sigma(f_{\boldsymbol{\phi}_t}(\boldsymbol{x}_t))$, where $\sigma(\cdot)$ is the logistic sigmoid function given by $\sigma(x) = e^x/(e^x+1)$ and $f_{\boldsymbol{\phi}_t} : \mathbb{R}^d \to \mathbb{R}$ is a neural network parameterized by $\boldsymbol{\phi}_t$. When the classifier is trained using maximum likelihood estimation, the optimal value for $f_{\boldsymbol{\phi}_t}$ is $\log r(\boldsymbol{x}_t)$ [Durkan 2020]. This yields the proposed neural online density-ratio estimator (NODE):

$$r_{\boldsymbol{\phi}_t}(\boldsymbol{x}_t) = e^{f_{\boldsymbol{\phi}_t}(\boldsymbol{x}_t)}\,, \tag{4.7}$$

where the subscript $\boldsymbol{\phi}_t$ emphasizes that $r_{\boldsymbol{\phi}_t}(\boldsymbol{x}_t)$ depends on the learned classifier. CPD is then performed by comparing the test statistic $|\frac{1}{N}\sum_{\boldsymbol{x}\in\mathcal{X}_t}(r_{\boldsymbol{\phi}_t}(\boldsymbol{x})-1)|$ to a given threshold $\xi \in \mathbb{R}_+$ since $r_{\boldsymbol{\phi}_t}(\boldsymbol{x}_t)$ (the ratio of the two posteriors) is biased from 1 when there is a changepoint. For a time series of d-dimensional vector-valued data $\{\boldsymbol{x}_t\}_{t\in\mathbb{N}}$, the the input and output dimensions of NODE are d and 1. The structure of NODE is detailed in Section 4.3. Here we provide more details of the training and test phases of our NODE:

Training phase: At each time instant t, all the samples of the reference and test windows with labels from $\mathcal{D}_t$ are used to estimate $\boldsymbol{\phi}_t$;

Test phase: Only the samples of the test window $\mathcal{X}'_t$ are used to estimate the density ratio $r_{\boldsymbol{\phi}_t}(\boldsymbol{x}_t)$ and the average over $\mathcal{X}_t$ is used to obtain more stable detection.

A crucial consideration for the proper estimation of $p(y_t=1|\boldsymbol{x}_t, \boldsymbol{\phi}_t)$ is that this classifier should avoid overfitting given the limited dataset $\mathcal{D}_t$ with $N+N'$ samples. This is particularly important since the window lengths directly impact the performance of the algorithm: they need to be small to limit the detection delay, but large in order to supply enough training data. This issue will be alleviated in the following by considering an online continual learning strategy for NODE, in which information from previous windows is leveraged when learning the current classifier.

4.2.2 Continual learning strategy

As discussed above, we train a neural classifier and estimate its parameters $\boldsymbol{\phi}_t$ using samples in $\mathcal{D}_t$ at each time instant t with label $y_t = 0$ in the reference window and label $y_t = 1$ in the test window. Since the overlapping part in training datasets at neighboring time instants, e.g., $\mathcal{D}_{t-1}$ and $\mathcal{D}_t$, is relatively large, it is beneficial to retain the knowledge acquired from $\mathcal{D}_{1:t-1}$ when training the classifier on $\mathcal{D}_t$. This is particularly important to benefit from past information and avoid overfitting when the window length is small. To iteratively learn the classifier while retaining the knowledge acquired from past iterations, we investigate a variational continual learning (VCL) strategy [Nguyen 2018] in our CPD algorithm.

Given an independent input $\boldsymbol{x}$, let us consider that the classifier returns a probability distribution $p(y|\boldsymbol{x}, \boldsymbol{\phi}_t)$ of its label y, given its parameters $\boldsymbol{\phi}_t$. Note that the classifier parameters are assumed to be random variables as this allows one to account for their uncertainty, which can be important when training with small amounts of data. In the continual learning setting, we aim to compute the distribution of the parameters $\boldsymbol{\phi}_t$, given the dataset $\mathcal{D}_t$. This is computed using Bayes' rule:

$$p(\boldsymbol{\phi}_t|\mathcal{D}_t) \propto p(\mathcal{D}_t|\boldsymbol{\phi}_t)p(\boldsymbol{\phi}_t)\,, \tag{4.8}$$

where $p(\boldsymbol{\phi}_t)$ is a properly selected prior for the parameters which captures the information from the past data. To compute $p(\boldsymbol{\phi}_t|\mathcal{D}_t)$ recursively, as in a Bayesian filtering framework, the prior $p(\boldsymbol{\phi}_t)$ is selected as the posterior distribution of the parameters computed at the previous iteration, $p(\boldsymbol{\phi}_{t-1}|\mathcal{D}_{t-1})$.

However, the posterior distribution is intractable in general and needs to be approximated. VCL [Nguyen 2018] approximates the posterior distribution by another distribution q belonging to a tractable family $\mathcal{Q}$. This is performed by finding the distribution $q \in \mathcal{Q}$ which minimizes the KL divergence to the true posterior:

$$q_t(\boldsymbol{\phi}_t) = \arg\min_{q\in\mathcal{Q}} \mathrm{KL}\Big(q(\boldsymbol{\phi})\Big\|\frac{1}{Z_t}p(\mathcal{D}_t|\boldsymbol{\phi})q_{t-1}(\boldsymbol{\phi})\Big)\,, \tag{4.9}$$

where $q_{t-1}(\boldsymbol{\phi})$ is the approximate posterior that was computed at time $t-1$, and Z_t is a normalizing constant (which will not be required in the optimization process). The zeroth approximated posterior $q_0(\boldsymbol{\phi})$ is defined as the prior distribution of the parameters $p(\boldsymbol{\phi})$, which is chosen to be a multivariate Gaussian distribution [Nguyen 2018]. Training the classifier using the variational inference in (4.9) is equivalent to maximizing the evidence lower bound to the data log-likelihood $\log p(\mathcal{D}_t)$, which leads to the following cost function:

$$\begin{aligned}\mathcal{L}_t(q_t(\boldsymbol{\phi})) = &\sum_{n=0}^{N+N'-1} \mathbb{E}_{\boldsymbol{\phi}\sim q_t(\boldsymbol{\phi})}\big\{\log p(y_{t-n}|\boldsymbol{\phi}, \boldsymbol{x}_{t-n})\big\} \\ &- \eta\,\mathrm{KL}\big(q_t(\boldsymbol{\phi})||q_{t-1}(\boldsymbol{\phi})\big)\,.\end{aligned} \tag{4.10}$$

Here we introduce a hyperparameter η to trade-off between stability of the continual learning strategy, and its ability to adapt in the presence of a changepoint.

Algorithm 5 CPD with NODE

Input : $\{\boldsymbol{x}_t\}$, parameter η, number of epochs M, threshold ξ.
Initialization : optimize (4.10) with $\eta = 0$ for $t = 1$.
for $t = 2, 3, \cdots$ **do**
  Update data windows to build the dataset $\mathcal{D}_t$.
  Optimize cost function (4.10) for M epochs to estimate parameter $\boldsymbol{\phi}_t$ at time t.
  Compute the density-ratio using (4.7).
  if $|\frac{1}{N}\sum_{\boldsymbol{x}\in\mathcal{X}_t}(r_{\boldsymbol{\phi}_t}(\boldsymbol{x}) - 1)| > \xi$ **then**
    Flag t as a changepoint.

We consider the variational family $\mathcal{Q}$ as the set of Gaussian distributions with diagonal covariance matrices (i.e., a mean field assumption), which makes the learning process more efficient since the KL divergence in (4.10) can be computed in closed form. In this work, $p(y|\boldsymbol{\phi}_t, \boldsymbol{x})$ is modeled as a Bernoulli distribution. The optimization of $\mathcal{L}_t(q_t(\boldsymbol{\phi}_t))$ is performed by using the Adam [Kingma 2014a] gradient-based optimizer, where the *reparametrization trick* [Kingma 2014b] was used to tackle the expectation with respect to $q_t(\boldsymbol{\phi}_t)$. At each time instant, (4.10) is maximized for M epochs, with the parameters of the distribution $q_t(\boldsymbol{\phi}_t)$ initialized with those of $q_{t-1}(\boldsymbol{\phi}_{t-1})$, obtained as the solution at $t-1$ (i.e., *warm start*). The proposed CPD procedure is summarized in Algorithm 5.

4.3 Experiments

In this section, we validate the proposed online CPD method with NODE and compare it with three baselines, namely, the kNN [Chen 2019], MA [Keriven 2020, Ferrari 2022] and NOUGAT [Ferrari 2022]. For all experiments, $f_{\boldsymbol{\phi}_t}$ was a fully connected network with three hidden layers, where each layer contained 16 units (32 units for real data) with Tanh activations. The reference and test window lengths were both set to $N = N' = 64$ for all algorithms. The network was trained for 20 epochs during initialization, then for $M = 1$ epoch for $t > 1$. We set $\eta = 20$ for simulated data and $\eta = 5$ for real data. We applied empirical methods to determine the above parameters. The codes are made available at `www.github.com/xiuheng-wang/NODE_release`.

4.3.1 Monte Carlo validation

The simulated signals $\boldsymbol{x}_t$ were sampled from mixtures of k d-dimensional Gaussian distributions $\mathcal{N}_d(\boldsymbol{m}_q, q^{-1}\boldsymbol{C}_q)$ with $q = 1, \ldots, d$. The weights α_q of the mixture model were generated from a flat Dirichlet distribution with concentration coefficient β. The means $\boldsymbol{m}_q$ and the covariance matrix $\boldsymbol{C}_q$ were sampled from $\mathcal{N}_d(\boldsymbol{0}, \boldsymbol{I})$ and a Wishart distribution with the scaling matrix $\boldsymbol{I}$ and $d+2$ degrees of freedom. We generated 700 samples and introduced a changepoint at $t_r = 400$. We set $d = 6, k = 3, \beta = 5$, and all parameters $\{\boldsymbol{m}_q, \alpha_q, \boldsymbol{C}_q\}$ were resampled at time $t = t_r$.

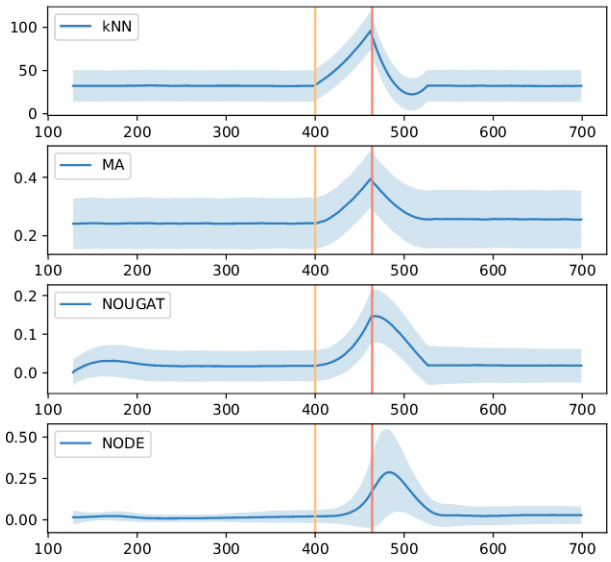

FIGURE 4.1 – Mean of the test statistic ($\pm$ standard deviation) for all compared algorithms. The changepoint t_r is located at the yellow line and $t_r + N$ at the red line.

FIGURE 4.1 shows the mean $\pm$ standard deviation of the test statistic of all compared algorithms for 10^4 Monte Carlo runs. Comparing the ratio between the test statistic at the peak at $t_r + N$ and before t_r, NODE achieved the best performance compared to the other methods. This can be seen more clearly in the Receiver Operating Characteristic (ROC) curves computed using the multiple Monte Carlo runs and shown in FIGURE 4.2, where NODE achieves an improvement of detection rate for false alarm rates from 0 to 0.4. Note the false alarm rate was computed as the probability to detect a changepoint at a time instant t with $t < t_r$ while the detection rate was estimated as the probability to detect *at least* a change at a time instant t with $t \geqslant t_r$, i.e. the probability that the test statistics is larger than the threshold at least once [Ferrari 2022]. The detection delay can be reflected as the peak of test statistics in FIGURE 4.1, one can observe NODE has a longer delay when achieves the best performance.

4.3.2 Credit card fraud detection

The real data in the credit card fraud detection dataset is composed of transactions made in September 2013 by European cardholders [1]. The raw data were preprocessed by applying PCA, and the first five components ($d = 5$) were considered to obtain streaming signals $\boldsymbol{x}_t$. This dataset contains 492 frauds out of 284,807 transactions. We inserted the 492 frauds after the first 1000 genuine transactions to create two changepoints at $t_r = 1000$ and $t_r = 1492$.

FIGURE 4.3 illustrates the detection statistics of kNN, MA, NOUGAT and

1. www.kaggle.com/datasets/mlg-ulb/creditcardfraud

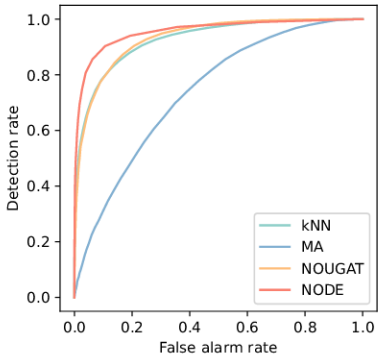

FIGURE 4.2 – ROC curves for all compared algorithms. The closer a ROC curve is to the upper left corner, the better the algorithm performs.

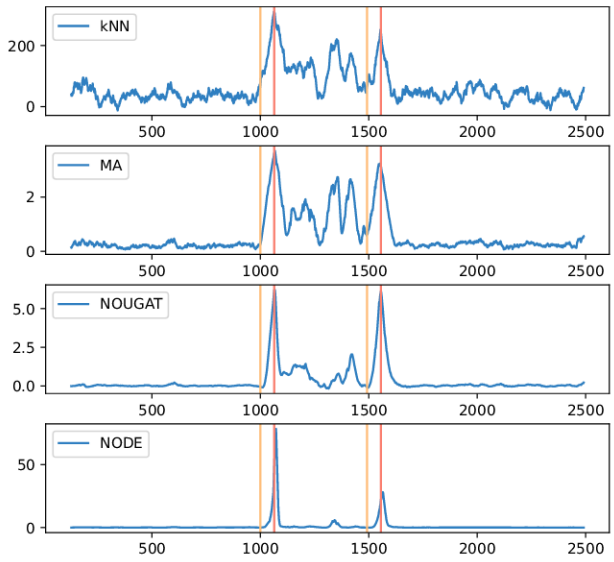

FIGURE 4.3 – Credit card fraud detection. The changepoint t_r is located at the yellow line and $t_r + N$ at the red line.

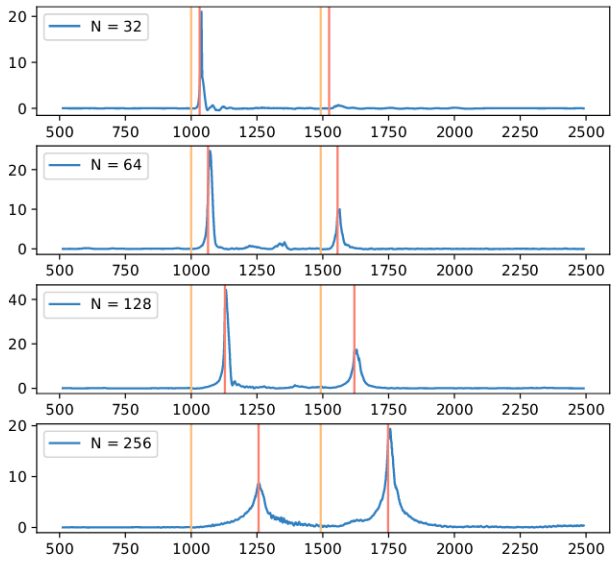

FIGURE 4.4 – The proposed algorithm for credit card fraud detection with a varying N. changepoints t_r are located at the yellow line. The red lines show $t_r + N$.

NODE. The test statistic values for all algorithms were significantly larger in the vicinity of $t_r + N$ compared to intervals not in the vicinity of the changepoints. However, the test statistics of kNN and MA showed large values between the two changepoints. NOUGAT performed better than MA and kNN, however, NODE obtained the best performance, with test statistic values that were only non-negligible after the changepoints, which translates into a very low false alarm rate.

To illustrate how sensitive the algorithm is w.r.t. the window length N, we conducted extra experiments with the credit card fraud detection setup where N varies. The results are provided in FIGURE 4.4. We observe that there is a trade-off between getting pronounced peaks in the test statistic, detecting consecutive changes, and achieving small detection delays.

4.3.3 Text language detection

The real dataset for text language detection was created from a dataset containing text from 17 different languages [2]. Raw texts were first cleaned by removing symbols and numbers and then represented via a linear embedding of dimensionality $d = 20$ using `word2vec`. Time series $\boldsymbol{x}_t$ was formed by concatenating the representations of 1014 French, 594 Malayalam, and 526 Arabic texts.

The results are provided in FIGURE 4.5. The test statistic of kNN produced very large values in the absence of changepoints when compared to the other algorithms, which leads to a large false alarm rate. MA, NOUGAT and NODE provided comparable results. However, NODE's test statistic was more stable outside of the vicinity of changepoints.

2. www.kaggle.com/datasets/basilb2s/language-detection

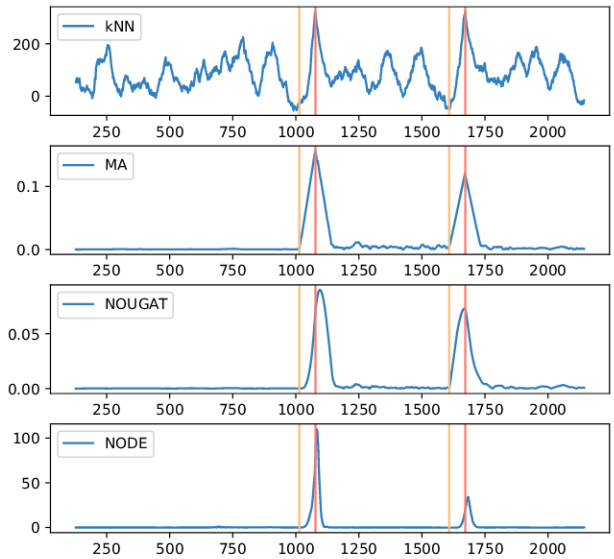

FIGURE 4.5 – Text language detection. The changepoint t_r is located at the yellow line and $t_r + N$ at the red line.

The execution times of NODE were about an order of magnitude larger than the other methods in all experiments. However, NODE was implemented in a different computation platform (Python) than the baselines (Julia), which reduces the appropriateness of comparing their execution times. A more in-depth study of the complexity of the proposed method and the development of more efficient solutions will be the subject of future work.

4.4 Conclusion

In this chapter, we introduced a novel strategy for online CPD that leverages the powerful learning ability of neural networks to estimate density-ratio in a non-parametric manner. A continual learning framework was exploited to devise an adaptive detection algorithm that retains past information. Experiments illustrated the superiority of the proposed strategy compared to state-of-the-art methods.

Chapter 5

Non-parametric online CPD on Riemannian manifolds

Contents

5.1 Introduction

Research on CPD can be categorized into two primary branches: offline and online. Offline CPD necessitates access to all received samples, as extensively covered in the literature [Truong 2020]. In contrast, online CPD methods process data in real-time and aim to detect changepoints with minimal delay after their occurrence. In numerous real-world scenarios, the pursuit of non-parametric CPD is also highly relevant since it can be challenging to possess prior knowledge of the data distribution. However, even with the longstanding history and continued interest in CPD techniques, it is noteworthy that the overwhelming majority of existing algorithms assume that the data resides in Euclidean spaces.

Recent developments in statistical learning and signal processing have increasingly confronted the analysis of data residing in non-Euclidean spaces. Among these spaces, Riemannian manifolds have garnered attention due to their wide-ranging applications, such as in diffusion tensor imaging [Pennec 2006b], pedestrian detection [Tuzel 2008], and human behavior understanding [Kacem 2018]. Consequently, Riemannian optimization [Absil 2009, Boumal 2023a] has emerged as an area of significant interest, offering essential and potent tools for handling data on manifolds, especially with the recent advancements in Riemannian stochastic gradient descent (R-SGD) algorithms [Bonnabel 2013, Zhang 2016b]. While the detection of change-points in Euclidean spaces has been notably successful, it is noteworthy that only a limited number of CPD methods have been specifically crafted for Riemannian manifolds [Bouchard 2020, Dubey 2020, Wang 2023a], and these still lack theoretical analyses or online operation. The main hurdles stem from the need to account for the intrinsic non-linear geometry of these spaces and the absence of a vector space structure in the data, making the adaptation of algorithms originally conceived for Euclidean spaces a complex undertaking.

In response to the aforementioned challenges, the objective of this chapter is to introduce a unified framework for non-parametric and online CPD on Riemannian manifolds. Our contributions are as follows:

1. **General non-parametric framework**: We propose a comprehensive non-parametric framework for CPD by monitoring central values within Riemannian manifolds. Our framework places particular emphasis on the generalized Karcher mean. We update two estimates of the generalized Karcher mean using the R-SGD algorithm, each with distinct constant stepsizes. These two estimates, one with longer memory and the other more adaptive, are compared to construct an online CPD statistic.
2. **Theoretical analyses**: We provide theoretical analyses related to the proposed CPD statistic. We establish non-asymptotic convergence results for R-SGD with a curvature-dependent linear rate under the condition of constant stepsize (Theorem 5.2.1). Additionally, in the absence of any change, we derive an upper bound for the false alarm rate (Theorem 5.2.2). Furthermore, in the presence of a change, we establish a lower bound for the detection rate (Theorem 5.2.3).
3. **Application to specific manifolds**: We tailor our algorithm to suit two common instances of Riemannian manifolds, specifically, the manifold of symmetric positive definite (SPD) matrices and the Grassmann manifold. We then provide empirical illustrations of the performance of our CPD algorithm on these manifolds through numerical experiments on synthetic and real-world datasets.

By introducing this framework and offering theoretical insights into its performance, we aim to contribute to the advancement of non-parametric and online CPD methods for data residing on Riemannian manifolds, which can impact a range of

applications such as, e.g., voice activity detection, pedestrian detection and subspace change detection.

5.1.1 Related work

In this section, we review related works on online changepoint detection and Riemannian optimization which are connected to the proposed approach.

Online CPD: Online CPD methods can be broadly categorized into two main groups: parametric and non-parametric, depending on whether prior knowledge about the data distribution is available. Parametric CPD techniques, illustrated by methods such as the cumulative sum (CUSUM) [Page 1954, Tartakovsky 2014] and the generalized likelihood ratio test (GLRT) [Gustafsson 1996], assume that the data distribution conforms to a known parametric family.

In many applications, prior knowledge of the data distribution cannot be guaranteed, leading to the development of non-parametric methods. These approaches encompass various techniques, including monitoring changes in the mean or variance of a data stream, as seen in methods like the Exponentially Weighted Moving Average (EWMA) [Costa 2006], and the use of kernel maximum mean discrepancy (MMD) derived from the data stream [Gretton 2006]. Recent advancements in this field have introduced innovative non-parametric methods. For instance, the NEWMA algorithm [Keriven 2020] was introduced to detect changepoints without the necessity of retaining historical data samples. This is achieved by comparing two EWMAs of data stream statistics, each computed with distinct forgetting factors. The non-parametric kernel MMD statistic initially introduced for hypothesis testing in [Gretton 2006] has recently been widely employed in the context of kernel CPD with both offline [Harchaoui 2008, Sinn 2012] as well as online algorithms [Gong 2012, Li 2019]. Kernel extensions of the CUSUM statistic have also been considered in [Madrid Padilla 2023, Arlot 2019, Wei 2022]. A computationally efficient approximation of the kernel MMD based on the neural tangent kernel has also been proposed [Cheng 2021]. Another non-parametric online algorithm was developed in [Ferrari 2022], making use of adaptive kernel density ratio estimation. The capabilities of neural networks were explored in [Wang 2023b] to enhance the effectiveness of non-parametric online CPD.

These algorithms, however, assume that the data belongs to an Euclidean space. While some non-parametric online CPD algorithms have been extended to specific non-Euclidean domains, such as graphs [Ferrari 2020a] and categorical data [Ienco 2014], very few works have investigated scenarios where the data belongs to a Riemannian manifold. In [Bouchard 2020], an online CPD algorithm was specifically designed for the compound Gaussian distribution, which, however, is parametric and not broadly applicable. For data lying on manifolds, a non-parametric offline algorithm [Duan 2019] was developed to detect changepoints of rigid body motions in the special Euclidean group. Another non-parametric technique, monitoring changes in the Fréchet means and variances, was proposed in [Dubey 2020]. However, it can

only detect a single changepoint and operates offline. A work extending NEWMA to manifolds was introduced in [Wang 2023a], but the algorithm is not general and does not have any theoretical analyses.

This chapter presents a general formulation for CPD on manifolds based on the generalized Karcher mean, a discussion of its related existence and uniqueness questions, theoretical results related to the convergence, false alarm, and detection performance of the algorithm, and exemplifies its application to different manifolds with challenging examples.

Riemannian optimization: Riemannian optimization has recently garnered significant interest as it takes into account the geometry of data manifolds, which is prevalent in many practical applications. Both the books [Absil 2009] and [Boumal 2023a] provide detailed presentations on Riemannian optimization. Substantial work has also been undertaken in order to extend optimization algorithms that were originally developed in Euclidean spaces, such as steepest descent [Smith 1994] and quasi-Newton [Huang 2015] algorithms, to Riemannian manifolds, as well as to study their convergence behavior.

The R-SGD algorithm, as presented in [Bonnabel 2013], has gained significant attention for its capability to handle noisy gradient estimates. Sophisticated variance reduction techniques have been recently introduced to provide algorithms with accelerated convergence rate [Zhang 2016a, Zhang 2018a, Zhou 2019]. While the asymptotic convergence of the R-SGD was studied in [Bonnabel 2013] for diminishing stepsizes, explicit convergence rates were not provided. Results on the sublinear convergence rates of first-order Riemannian optimization on geodesically convex problems were recently obtained in [Zhang 2016b]. However, these rates were derived under the assumption of diminishing stepsizes or deterministic gradients.

5.1.2 Background

This section introduces some basic concepts of Riemannian geometry, focusing on the essential tools for optimization on manifolds. Detailed presentations can be found in [Absil 2009] and [Boumal 2023a].

A *Riemannian manifold* $(\mathcal{M}, g)$ is a constrained set $\mathcal{M}$ endowed with a *Riemannian metric* $g_{\boldsymbol{x}}(\cdot,\cdot) : T_{\boldsymbol{x}}\mathcal{M} \times T_{\boldsymbol{x}}\mathcal{M} \to \mathbb{R}$, defined for every point $\boldsymbol{x} \in \mathcal{M}$, with $T_{\boldsymbol{x}}\mathcal{M}$ the so-called *tangent space* of $\mathcal{M}$ at $\boldsymbol{x}$. A *geodesic* $\gamma : [0,1] \to \mathcal{M}$ is the curve of minimal length linking two points $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{M}$ such that $\boldsymbol{x} = \gamma(0)$ and $\boldsymbol{y} = \gamma(1)$, with $\boldsymbol{v} \in T_{\boldsymbol{x}}\mathcal{M}$ the velocity of γ at 0 denoted by $\dot{\gamma}(0)$. The *geodesic distance* $d_{\mathcal{M}}(\cdot\,,\cdot) : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is defined as the length of the geodesic linking two points $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{M}$. It satisfies all the conditions to be a metric.

The *exponential map* $\boldsymbol{w} = \exp_{\boldsymbol{x}}(\boldsymbol{v})$ is defined as the point $\boldsymbol{w} \in \mathcal{M}$ located on the unique geodesic $\gamma(t)$ with endpoints $\boldsymbol{x} = \gamma(0)$, $\boldsymbol{w} = \gamma(1)$ and velocity $\boldsymbol{v} = \dot{\gamma}(0)$. Since calculating the exponential map can be computationally demanding, in practice it is common to employ a *retraction* $R_{\boldsymbol{x}} : T_{\boldsymbol{x}}\mathcal{M} \to \mathcal{M}$ instead, defined at every $\boldsymbol{x} \in \mathcal{M}$, which consists of a second-order approximation to the exponential map,

satisfying $d_{\mathcal{M}}(R_{\boldsymbol{x}}(t\boldsymbol{v}), \exp_{\boldsymbol{x}}(t\boldsymbol{v})) = \mathcal{O}(t^3)$. Consider a smooth function $f : \mathcal{M} \to \mathbb{R}$. The *Riemannian gradient* of f at $\boldsymbol{x} \in \mathcal{M}$ is defined as the unique tangent vector $\nabla f(\boldsymbol{x}) \in T_{\boldsymbol{x}}\mathcal{M}$ satisfying $\frac{d}{dt}\big|_{t=0} f(\exp_{\boldsymbol{x}}(t\boldsymbol{v})) = \langle \nabla f(\boldsymbol{x}), \boldsymbol{v} \rangle_{\boldsymbol{x}}$, for all $\boldsymbol{v} \in T_{\boldsymbol{x}}\mathcal{M}$.

5.2 Proposed method

5.2.1 Problem Background

Consider a sequence of statistically independent samples $\boldsymbol{x}_t$ belonging to a Riemannian manifold $\mathcal{M}$. The Riemannian CPD problem consists of estimating the time index $t_r \in \mathbb{N}$, referred to as the *changepoint*, at which the probability measure of $\boldsymbol{x}_t$ undergoes a change [Pennec 2004]:

$$\begin{aligned} t < t_r &: \ \boldsymbol{x}_t \sim P_1(\boldsymbol{x}) \,, \\ t \geqslant t_r &: \ \boldsymbol{x}_t \sim P_2(\boldsymbol{x}) \,. \end{aligned} \tag{5.1}$$

Here, $P_1(\boldsymbol{x})$ and $P_2(\boldsymbol{x})$ are probability measures on $\mathcal{M}$, such that $P_1(\boldsymbol{x}) \neq P_2(\boldsymbol{x})$, representing how $\boldsymbol{x}_t$ is distributed before and after the changepoint, respectively. Throughout this chapter, it is assumed that the difference between the generalized Karcher means of $P_1(\boldsymbol{x})$ and $P_2(\boldsymbol{x})$ (see Section 5.2.2 for a definition) is sufficiently large, to make this problem tractable.

While various CPD algorithms have been proposed for Euclidean spaces, the constraint that the data $\boldsymbol{x}_t$ belongs to a Riemannian manifold $\mathcal{M}$, which typically lacks a vector space structure, presents challenges for algorithm design.

5.2.2 The algorithm

In this study, we introduce a non-parametric CPD strategy designed for situations where there is no prior knowledge about the probability measures of the data. In Euclidean spaces, this has been accomplished in particular by monitoring changes in either the mean or the variance [Costa 2006], or in a generalized statistics [Gretton 2006] of the data stream. We propose to extend such strategies to Riemannian manifolds by monitoring changes in a generalized moment of $\boldsymbol{x}_t \in \mathcal{M}$. This generalized moment can include the Fréchet mean [Fréchet 1948], which extends the concept of the Euclidean mean to metric spaces. In a broader sense, we consider a *generalized Fréchet mean* of $\mathcal{M}$, as defined in [Schötz 2019]:

$$\boldsymbol{m}^* \in \arg\min_{\boldsymbol{m} \in \mathcal{M}} f(\boldsymbol{m}) \,, \tag{5.2}$$

where $f(\boldsymbol{m})$ is given by:

$$f(\boldsymbol{m}) = \mathbb{E}_{\boldsymbol{x} \sim P(\boldsymbol{x})}\big\{c(\boldsymbol{x}, \boldsymbol{m})\big\} = \int c(\boldsymbol{x}, \boldsymbol{m}) dP(\boldsymbol{x}) \,,$$

with $c : \mathcal{M} \times \mathcal{M} \to [0, +\infty)$ an appropriate cost function. This framework generalizes several important central values on Riemannian manifolds, including the Fréchet

mean by considering $c(\boldsymbol{x}, \boldsymbol{m}) = d^2_{\mathcal{M}}(\boldsymbol{x}, \boldsymbol{m})$ where $d_{\mathcal{M}}(\boldsymbol{x}, \boldsymbol{m})$ denotes the geodesic distance between $\boldsymbol{x}$ and $\boldsymbol{m}$, and the median by setting $c(\boldsymbol{x}, \boldsymbol{m}) = d_{\mathcal{M}}(\boldsymbol{x}, \boldsymbol{m})$.

The existence and uniqueness of minimizers for (5.2) is not guaranteed in general, even in the case of the Fréchet mean. When $c = d^2_{\mathcal{M}}$, the *Karcher mean* relaxes this definition by considering the local optima of $f(\boldsymbol{m})$ rather than only the global one. This allows us to establish existence and uniqueness conditions [Kendall 1990] and compute $\boldsymbol{m}$ by solving (5.2) locally using Riemannian optimization methods [Pennec 2004]. In particular, if the support of $P(\boldsymbol{x})$ is included in a regular geodesic ball (see definition 5 in [Pennec 2006a]), then the Karcher mean exists and is unique [Kendall 1990]. This condition is satisfied for connected manifolds with non-positive curvature [Afsari 2011], referred to as *Hadamard manifolds* [Shiga 1984]. In this work, we extend this concept by defining the *generalized Karcher mean* as the set of local minimizers of (5.2) with various central values. Although our framework is considered in a broader sense, we will focus on the case of Karcher mean in Section 5.2.3, as discussed in [Wang 2023a], for the sake of convenience and to facilitate the theoretical analysis.

The proposed CPD strategy on manifolds will be designed to monitor abrupt changes in a generalized Karcher mean. An important requirement is that change-points must be detected in an online manner, meaning that they are based only on past data. Consequently, we will adopt stochastic Riemannian optimization to estimate the generalized Karcher mean of the streaming data $\boldsymbol{x}_t$ in an online manner. This will constitute a fundamental component of our approach.

5.2.2.1 Online estimation

In a non-parametric setting, it is not possible to compute the solution to the optimization problem in (5.2) explicitly because $P(\boldsymbol{x})$ is unknown. Instead, we assume that we have access to observations $\boldsymbol{x}_t$ and can evaluate both the cost function $c(\boldsymbol{m}, \boldsymbol{x}_t)$ and its Riemannian gradient for any parameter $\boldsymbol{m}$ and sample $\boldsymbol{x}_t$. This enables us to construct a stochastic approximation of the gradient $\nabla f(\boldsymbol{m})$ using the input $\boldsymbol{x}_t$. Consequently, we can utilize the R-SGD algorithm [Bonnabel 2013] to compute an online solution to (5.2). An update of $\boldsymbol{m}$ can be computed on $\mathcal{M}$ as:

$$\boldsymbol{m}_{t+1} = \exp_{\boldsymbol{m}_t}\big(-\alpha H(\boldsymbol{m}_t, \boldsymbol{x}_t)\big)\,, \tag{5.3}$$

with $\alpha > 0$ a constant stepsize. In this expression, $\exp_{\boldsymbol{m}}$ denotes the exponential map at $\boldsymbol{m}$, and $H(\boldsymbol{m}, \boldsymbol{x})$ is the stochastic Riemannian gradient, assumed to be an unbiased estimate of the full gradient $\nabla f(\boldsymbol{m})$,

$$\mathbb{E}_{\boldsymbol{x}\sim P(\boldsymbol{x})}\big\{H(\boldsymbol{m}, \boldsymbol{x})\big\} = \int H(\boldsymbol{m}, \boldsymbol{x}) dP(\boldsymbol{x}) = \nabla f(\boldsymbol{m})\,.$$

The mathematical definitions of $H(\boldsymbol{m}, \boldsymbol{x})$ for two specific manifolds are given in (5.53) and (5.57) for examples. The exponential map in (5.3) can also be replaced by a computationally simpler retraction $R_{\boldsymbol{m}_t}$. It is important to note that we are considering R-SGD in a non-standard setting. The estimates provided by this algorithm

should be able to adapt to changes in the data distribution and, consequently, to the underlying cost function $f(\boldsymbol{m})$. This necessitates the use of a constant (instead of diminishing) stepsize α, which will impact the theoretical analysis in Section 5.2.3 since non-asymptotic convergence results will be required.

5.2.2.2 An adaptive CPD

Our goal is to detect changepoints by monitoring abrupt changes in the value of $\boldsymbol{m}$ over time. In simpler terms, we label a time index t' as a changepoint if there is a sudden shift in the value of $\boldsymbol{m}$ at that time. This requires knowledge of two quantities of interest, $\boldsymbol{m}_{\text{bef}}$ and $\boldsymbol{m}_{\text{aft}}$, which respectively represent the generalized Karcher mean before and after a candidate changepoint t'. First, we propose an approach to compute estimates of these quantities, denoted as $\widehat{\boldsymbol{m}}_{\text{bef}}$ and $\widehat{\boldsymbol{m}}_{\text{aft}}$. Then, a test statistic is designed to compare these two quantities using the Riemannian distance, specifically $d_{\mathcal{M}}(\widehat{\boldsymbol{m}}_{\text{bef}}, \widehat{\boldsymbol{m}}_{\text{aft}})$. The larger the Riemannian distance between the generalized Karcher mean estimates before and after time instant t', the higher the likelihood of identifying t' as a changepoint.

The challenge is to find a computationally efficient online method for calculating $\widehat{\boldsymbol{m}}_{\text{bef}}$ and $\widehat{\boldsymbol{m}}_{\text{aft}}$. Previous work [Dubey 2020] proposed dividing a data stream $\{\boldsymbol{x}_t\}_{t=1}^{N}$ with N samples into two segments, $\{1, \ldots, t'-1\}$ and $\{t', \ldots, N\}$ for every t', and testing for differences between their Karcher mean and variance. However, this approach of comparing the values of the Karcher means estimated using samples before and after the time instant t is not suitable for processing data streams on the fly or detecting multiple changepoints. In [Keriven 2020], estimates of $\widehat{\boldsymbol{m}}_{\text{bef}}$ and $\widehat{\boldsymbol{m}}_{\text{aft}}$ were computed considering the data $\boldsymbol{x}_t$ to belong to a Euclidean space. This was achieved using two Exponentially weighted Moving Averages (EWMAs) with different forgetting factors: one adapting quickly to track $\widehat{\boldsymbol{m}}_{\text{aft}}$ after a changepoint, and another adapting slowly to keep track of $\widehat{\boldsymbol{m}}_{\text{bef}}$. However, this approach cannot be directly applied to Riemannian manifolds due to its lack of accounting for manifold geometry. Instead, we propose using two iterative estimates computed using R-SGD algorithms, described in Section 5.2.2.1, with two different fixed stepsizes $\lambda < \Lambda$. The generalized Karcher mean estimates are updated according to (5.3) as:

$$\boldsymbol{m}_{\lambda,t+1} = \exp_{\boldsymbol{m}_{\lambda,t}}\big(-\lambda H(\boldsymbol{m}_{\lambda,t}, \boldsymbol{x}_t)\big)\,, \tag{5.4}$$

$$\boldsymbol{m}_{\Lambda,t+1} = \exp_{\boldsymbol{m}_{\Lambda,t}}\big(-\Lambda H(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{x}_t)\big)\,, \tag{5.5}$$

with initialization $\boldsymbol{m}_{\lambda,0} = \boldsymbol{m}_{\Lambda,0} = \boldsymbol{x}_0$. The convergence of the updates (5.4) and (5.5) is directly influenced by λ and Λ, with a larger stepsize typically resulting in faster convergence, as we will demonstrate in Theorem 5.2.1 in the next section. Therefore, having $0 < \lambda < \Lambda$ means that $\boldsymbol{m}_{\Lambda,t}$ is more likely to adapt to new data and quickly approximate $\widehat{\boldsymbol{m}}_{\text{aft}}$, while $\boldsymbol{m}_{\lambda,t}$ has a longer memory and is better suited for estimating $\widehat{\boldsymbol{m}}_{\text{bef}}$. The motivation is similar to NEWMA [Keriven 2020] because $\boldsymbol{m}_{\Lambda,t}$ with a larger stepsize gives more importance to upcoming $\boldsymbol{x}_t$ than $\boldsymbol{m}_{\lambda,t}$ with a smaller stepsize. Using constant stepsizes is crucial to allow the algorithm to adapt to changes in the data distribution and detect multiple changepoints. In the limit,

Algorithm 6 Online CPD on Riemannian manifolds

Input: $\{\boldsymbol{x}_t\}$, stepsizes λ, Λ, threshold ξ.
Initialization: $\boldsymbol{m}_{\lambda,0} = \boldsymbol{m}_{\Lambda,0} = \boldsymbol{x}_0$.
for $t = 1, \ldots$ **do**
  Update the generalized Karcher mean estimates $\boldsymbol{m}_{\lambda,t}$ and $\boldsymbol{m}_{\Lambda,t}$ using (5.4) and (5.5);
  Compute the test statistic $g_t = d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{m}_{\Lambda,t})$;
  if $g_t > \xi$ **then**
    Flag t as a changepoint;

the estimates provided in (5.4) and (5.5) will converge to one Karcher mean under the null hypothesis and converge to another Karcher mean after a changepoint.

Based on the estimates provided in (5.4) and (5.5), we can formulate an adaptive CPD statistic by calculating the difference between $\boldsymbol{m}_{\lambda,t}$ and $\boldsymbol{m}_{\Lambda,t}$ using the Riemannian distance on $\mathcal{M}$ as follows:

$$g_t = d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{m}_{\Lambda,t})\,. \tag{5.6}$$

The CPD procedure involves comparing the statistic g_t to a given threshold ξ. The complete CPD procedure is outlined in Algorithm 6. It is important to note that the selection of ξ directly impacts its average run length and detection delay, as will be shown in Theorems 5.2.2 and 5.2.3, which give bounds on the probability of a false alarm and of detecting a true changepoint. Moreover, as in (N)EWMA methods, the time interval between changepoints must be sufficiently large so that the algorithms converge to obtain adequate detection and false alarm performance.

5.2.3 Theoretical analysis

In this section, we will assess the performance of the proposed CPD algorithm in two main aspects: i) the probability of a false alarm, which refers to the probability of incorrectly identifying a time step as a changepoint, and ii) the probability of correctly identifying a changepoint when there is a shift in the generalized Karcher mean of the data stream. To achieve this, we will also need a supplementary outcome, iii) the non-asymptotic convergence analysis of the R-SGD algorithm with a constant stepsize.

For the sake of feasibility in our theoretical analysis, we will concentrate on the Karcher mean with $c = d^2_{\mathcal{M}}$, and the R-SGD cost function $f(\boldsymbol{m}) = \mathbb{E}\{d^2_{\mathcal{M}}(\boldsymbol{m}, \boldsymbol{x})\}$, which corresponds to the Karcher variance, and is minimized using the iterative updates in the from of (5.3). However, it is important to note that our convergence analysis of R-SGD will not be limited to this particular cost function. We will also focus on Hadamard manifolds as in [Zhang 2016b]. Before presenting the theoretical results, let us introduce some definitions related to the cost function f and its properties as follows.

Definition 1 (Geodesically strong convexity) A function $f : \mathcal{M} \to \mathbb{R}$ is geodesically μ-strongly convex if for any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{M}$, we have:

$$f(\boldsymbol{y}) \geqslant f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \exp_{\boldsymbol{x}}^{-1}(\boldsymbol{y}) \rangle + \frac{\mu}{2} \| \exp_{\boldsymbol{x}}^{-1}(\boldsymbol{y}) \|^2 . \tag{5.7}$$

Definition 2 (Lipschitz gradients) The gradient of a function $f : \mathcal{M} \to \mathbb{R}$ is said to be L-Lipschitz if, for any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{M}$ in the domain of f, it satisfies:

$$\left\| \nabla f(\boldsymbol{x}) - \Gamma_{\boldsymbol{y}}^{\boldsymbol{x}} \nabla f(\boldsymbol{y}) \right\| \leqslant L \, d_{\mathcal{M}}(\boldsymbol{x}, \boldsymbol{y}) , \tag{5.8}$$

where $\Gamma_{\boldsymbol{y}}^{\boldsymbol{x}}$ denotes the parallel transport from $\boldsymbol{y}$ to $\boldsymbol{x}$.

Definition 3 (Smoothness) Any differentiable function $f : \mathcal{M} \to \mathbb{R}$ is geodesically L-smooth if its gradient is L-Lipschitz, i.e., for any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{M}$, it satisfyies:

$$f(\boldsymbol{y}) \leqslant f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \exp_{\boldsymbol{x}}^{-1}(\boldsymbol{y}) \rangle + \frac{L}{2} \| \exp_{\boldsymbol{x}}^{-1}(\boldsymbol{y}) \|^2 . \tag{5.9}$$

5.2.3.1 Non-asymptotic convergence of R-SGD

The following theorem shows that the R-SGD algorithm (5.3) with a fixed stepsize $\alpha > 0$ has a curvature-dependent linear rate of convergence for geodesically strongly convex and smooth functions on Riemannian manifolds.

Theorem 5.2.1. *Assuming that $f : \mathcal{M} \to \mathbb{R}$ is geodesically μ-strongly convex with geodesically L-Lipschitz gradient, the diameter of the domain is bounded by D, the sectional curvature of the manifold is bounded below by κ, and the stochastic gradient is an unbiased estimator of the gradient, namely, $\mathbb{E}_{\boldsymbol{x}_t}\big\{H(\boldsymbol{m}_t, \boldsymbol{x}_t)\big\} = \nabla f(\boldsymbol{m}_t)$ with variance $\mathbb{E}_{\boldsymbol{x}_t}\{\|\nabla f(\boldsymbol{m}_t) - H(\boldsymbol{m}_t, \boldsymbol{x}_t)\|^2\} \leqslant \sigma^2$ and magnitude bounded by $\|H(\boldsymbol{m}_t, \boldsymbol{x}_t)\| < \rho$. We assume that the stepsize satisfies $0 < \alpha \leqslant \min\{\frac{1}{2L}, \frac{I}{\rho}\}$, where I is the injectivity radius of $\mathcal{M}$. Then, for any $s \in \mathbb{N}_*$, the stochastic Riemannian gradient descent algorithm satisfies*:

$$\mathbb{E}\{f(\boldsymbol{m}_s) - f(\boldsymbol{m}^*)\} \leqslant \frac{(1-\varepsilon)^{(s-1)} D^2}{2\alpha} + \frac{\alpha \sigma^2}{2\varepsilon} , \tag{5.10}$$

with $\boldsymbol{m}^*$ the optimum, $\varepsilon = \min\{\frac{1}{\zeta(\kappa, D)}, \alpha\mu\}$ and $\zeta(\kappa, D) = \frac{\sqrt{|\kappa|} D}{\tanh(\sqrt{|\kappa|} D)}$.

Proof. Assume f is a geodesically L-smooth function, that is, its gradient is geodesically L-Lipschitz. As this property is related to deterministic gradient $\nabla f(x)$, we shall first reformulate it with respect to the stochastic gradient. Replacing $y = \boldsymbol{m}_{t+1}$, $x = \boldsymbol{m}_t$ in (5.9), denote $\Delta_t = f(\boldsymbol{m}_t) - f(\boldsymbol{m}^*)$, and considering the fact

$\langle a, b\rangle \leqslant \frac{\alpha}{2}a^2 + \frac{1}{2\alpha}b^2$, we have:

$$\begin{aligned}
\Delta_{t+1} - \Delta_t &= f(\boldsymbol{m}_{t+1}) - f(\boldsymbol{m}_t) \\
&\leqslant \langle \nabla f(\boldsymbol{m}_t), \exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}_{t+1})\rangle + \frac{L}{2}\|\exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}_{t+1})\|^2 \\
&= \langle H(\boldsymbol{m}_t, \boldsymbol{x}_t), \exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}_{t+1})\rangle + \langle \nabla f(\boldsymbol{m}_t) - H(\boldsymbol{m}_t, \boldsymbol{x}_t), \exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}_{t+1})\rangle \\
&\quad + \frac{L}{2}\|\exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}_{t+1})\|^2 \\
&\leqslant \langle H(\boldsymbol{m}_t, \boldsymbol{x}_t), \exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}_{t+1})\rangle + \frac{\alpha}{2}\|\nabla f(\boldsymbol{m}_t) - H(\boldsymbol{m}_t, \boldsymbol{x}_t)\|^2 \\
&\quad + \left(\frac{L}{2} + \frac{1}{2\alpha}\right)\|\exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}_{t+1})\|^2. \qquad (5.11)
\end{aligned}$$

Assuming $\|H(\boldsymbol{m}_t, \boldsymbol{x}_t)\| < \rho$ and $0 < \alpha \leqslant \frac{I}{\rho}$ where I is the injectivity radius of $\mathcal{M}$, we have $\|\alpha H(\boldsymbol{m}_t, \boldsymbol{x}_t)\| < I$. By Proposition 10.22 of [Boumal 2023a], $\exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}_{t+1}) = \exp_{\boldsymbol{m}_t}^{-1}(\exp_{\boldsymbol{m}_t}(-\alpha H(\boldsymbol{m}_t, \boldsymbol{x}_t)) = -\alpha H(\boldsymbol{m}_t, \boldsymbol{x}_t)$, taking the expectation w.r.t. $\{\boldsymbol{x}_s\}_{s=0}^t$, one obtains:

$$\begin{aligned}
\mathbb{E}\Delta_{t+1} - \mathbb{E}\Delta_t &\leqslant -\alpha\mathbb{E}\|H(\boldsymbol{m}_t, \boldsymbol{x}_t)\|^2 + \frac{\alpha\sigma^2}{2} + \left(\frac{L}{2} + \frac{1}{2\alpha}\right)\alpha^2\mathbb{E}\|H(\boldsymbol{m}_t, \boldsymbol{x}_t)\|^2 \\
&= \frac{\alpha\sigma^2}{2} + \left(\frac{\alpha L + 1}{2} - 1\right)\alpha\mathbb{E}\|H(\boldsymbol{m}_t, \boldsymbol{x}_t)\|^2. \qquad (5.12)
\end{aligned}$$

Assuming $0 \leqslant \alpha \leqslant \frac{1}{2L}$, we have:

$$\mathbb{E}\Delta_{t+1} - \mathbb{E}\Delta_t \leqslant \frac{\alpha\sigma^2}{2} - \frac{\alpha}{4}\mathbb{E}\|H(\boldsymbol{m}_t, \boldsymbol{x}_t)\|^2. \qquad (5.13)$$

Assume f is a geodesically μ-strongly convex function, replacing $y = \boldsymbol{m}^*$, $x = \boldsymbol{m}_t$ in (5.7), we have:

$$\begin{aligned}
f(\boldsymbol{m}_t) - f(\boldsymbol{m}^*) &\leqslant \langle -\nabla f(\boldsymbol{m}_t), \exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}^*)\rangle - \frac{\mu}{2}\|\exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}^*)\|^2 \\
&= \langle -\nabla f(\boldsymbol{m}_t), \exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}^*)\rangle - \frac{\mu}{2}d_{\mathcal{M}}^2(\boldsymbol{m}_t, \boldsymbol{m}^*)\,. \qquad (5.14)
\end{aligned}$$

Assume the diameter of the domain is bounded above by D, and the sectional curvature lower-bounded by $\kappa < 0$, use the trigonometric distance bound, i.e., Corollary 8 in [Zhang 2016b], we have:

$$\begin{aligned}
\langle -H(\boldsymbol{m}_t, \boldsymbol{x}_t), \exp_{\boldsymbol{m}_t}^{-1}(\boldsymbol{m}^*)\rangle &\leqslant \frac{1}{2\alpha}\big(d_{\mathcal{M}}^2(\boldsymbol{m}_t, \boldsymbol{m}^*) - d_{\mathcal{M}}^2(\boldsymbol{m}_{t+1}, \boldsymbol{m}^*)\big) \\
&\quad + \frac{\zeta(\kappa, D)\alpha}{2}\|H(\boldsymbol{m}_t, \boldsymbol{x}_t)\|^2. \qquad (5.15)
\end{aligned}$$

By taking the expectation of (5.15) w.r.t. $\boldsymbol{x}_t$ and combining it with (5.14) by using the fact $\mathbb{E}_{\boldsymbol{x}_t}\big\{H(\boldsymbol{m}_t, \boldsymbol{x}_t)\big\} = \nabla f(\boldsymbol{m}_t)$, and then taking expectation of the combined result w.r.t. $\{\boldsymbol{x}_s\}_{s=0}^t$, we obtain:

$$\begin{aligned}
\mathbb{E}\Delta_t = \mathbb{E}\{f(\boldsymbol{m}_t) - f(\boldsymbol{m}^*)\} &\leqslant \left(\frac{1 - \alpha\mu}{2\alpha}\right)\mathbb{E}d_{\mathcal{M}}^2(\boldsymbol{m}_t, \boldsymbol{m}^*) - \frac{1}{2\alpha}\mathbb{E}d_{\mathcal{M}}^2(\boldsymbol{m}_{t+1}, \boldsymbol{m}^*) \\
&\quad + \frac{\zeta(\kappa, D)\alpha}{2}\mathbb{E}\|H(\boldsymbol{m}_t, \boldsymbol{x}_t)\|^2\,.
\end{aligned}$$

Multiplying (5.13) by $2\zeta(\kappa, D)$ and adding to the previous inequality, we have:

$$2\zeta(\kappa, D)\mathbb{E}\Delta_{t+1} - (2\zeta(\kappa, D) - 1)\mathbb{E}\Delta_t \leqslant \Big(\frac{1-\alpha\mu}{2\alpha}\Big)\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_t, \boldsymbol{m}^*) - \frac{1}{2\alpha}\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_{t+1}, \boldsymbol{m}^*) + \alpha\sigma^2\zeta(\kappa, D)\,. \tag{5.16}$$

Multiplying (5.16) by $(1-\varepsilon)^{-t}$, we have:

$$\begin{aligned} &2(1-\varepsilon)^{-t}\zeta(\kappa, D)\mathbb{E}\Delta_{t+1} - 2(1-\varepsilon)^{-t}\Big(1 - \frac{1}{2\zeta(\kappa, D)}\Big)\zeta(\kappa, D)\mathbb{E}\Delta_t \\ &\leqslant (1-\varepsilon)^{-t}\big(1-\alpha\mu\big)\frac{1}{2\alpha}\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_t, \boldsymbol{m}^*) - (1-\varepsilon)^{-t}\frac{1}{2\alpha}\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_{t+1}, \boldsymbol{m}^*) \\ &\quad + (1-\varepsilon)^{-t}\alpha\sigma^2\zeta(\kappa, D)\,. \end{aligned} \tag{5.17}$$

We want to sum (5.17) from $t = 0$ to $t = s-1$. However, to simplify the summation, we consider the case $t = 0$ and $t \geqslant 1$ separately, because in the latter case, we can get a simpler upper bound. First, let us consider the case $t \geqslant 1$. Let $\varepsilon = \min\{\frac{1}{2\zeta(\kappa,D)}, \alpha\mu\}$ [Zhang 2016b], this implies $\varepsilon \leqslant \frac{1}{2\zeta(\kappa,D)}$ and $\varepsilon \leqslant \alpha\mu$. For $t \geqslant 1$, from (5.17) we have:

$$\begin{aligned} 2(1-\varepsilon)^{-t}\zeta(\kappa, D)\mathbb{E}\Delta_{t+1} - 2(1-\varepsilon)^{-(t-1)}\zeta(\kappa, D)\mathbb{E}\Delta_t &\leqslant (1-\varepsilon)^{-(t-1)}\frac{1}{2\alpha}\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_t, \boldsymbol{m}^*) \\ &\quad - (1-\varepsilon)^{-t}\frac{1}{2\alpha}\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_{t+1}, \boldsymbol{m}^*) \\ &\quad + (1-\varepsilon)^{-t}\alpha\sigma^2\zeta(\kappa, D)\,. \end{aligned} \tag{5.18}$$

Now, let us consider the case $t = 0$. This case is simple, directly from (5.17) we have:

$$2\zeta(\kappa, D)\mathbb{E}\Delta_1 - (2\zeta(\kappa, D) - 1)\mathbb{E}\Delta_0 \leqslant \Big(\frac{1-\alpha\mu}{2\alpha}\Big)\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_0, \boldsymbol{m}^*) - \frac{1}{2\alpha}\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_1, \boldsymbol{m}^*) + \alpha\sigma^2\zeta(\kappa, D)\,. \tag{5.19}$$

Finally, summing (5.17) over t from $t = 0$ to $t = s-1$, and using the previous results, we have:

$$\begin{aligned} 2(1-\varepsilon)^{-(s-1)}\zeta(\kappa, D)\mathbb{E}\Delta_s - (2\zeta(\kappa, D) - 1)\mathbb{E}\Delta_0 &\leqslant \Big(\frac{1-\alpha\mu}{2\alpha}\Big)\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_0, \boldsymbol{m}^*) \\ &\quad - \frac{(1-\varepsilon)^{-(s-1)}}{2\alpha}\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_s, \boldsymbol{m}^*) \\ &\quad + \sum_{t=0}^{s-1}(1-\varepsilon)^{-t}\alpha\sigma^2\zeta(\kappa, D) \\ &\leqslant \Big(\frac{1-\alpha\mu}{2\alpha}\Big)\mathbb{E}d^2_{\mathcal{M}}(\boldsymbol{m}_0, \boldsymbol{m}^*) \\ &\quad + \sum_{t=0}^{s-1}(1-\varepsilon)^{-t}\alpha\sigma^2\zeta(\kappa, D)\,, \end{aligned} \tag{5.20}$$

and plugging in $d_{\mathcal{M}}(\boldsymbol{m}_0, \boldsymbol{m}^*) \leqslant D$ (the diameter of the domain is bounded above by D), we have:

$$\begin{aligned} 2(1-\varepsilon)^{-(s-1)}\zeta(\kappa, D)\mathbb{E}\Delta_s - (2\zeta(\kappa, D) - 1)\mathbb{E}\Delta_0 &\leqslant \Big(\frac{1}{2\alpha} - \frac{\mu}{2}\Big)D^2 \\ &\quad + \sum_{t=0}^{s-1}(1-\varepsilon)^{-t}\alpha\sigma^2\zeta(\kappa, D) \\ &\leqslant \frac{D^2}{2\alpha} + \sum_{t=0}^{s-1}(1-\varepsilon)^{-t}\alpha\sigma^2\zeta(\kappa, D)\,. \end{aligned} \tag{5.21}$$

Replacing $y = \boldsymbol{m}_0$, $x = \boldsymbol{m}^*$ in (5.9), considering an alternative definition of geodesic L-smoothness (Proposition 4.5 and 4.6. of [Boumal 2023a]) and plugging in $d_{\mathcal{M}}(\boldsymbol{m}_0, \boldsymbol{m}^*) \leqslant D$ and $\nabla f(\boldsymbol{m}^*) = 0$, we have:

$$\begin{aligned} \Delta_0 = f(\boldsymbol{m}_0) - f(\boldsymbol{m}^*) &\leqslant \langle \nabla f(\boldsymbol{m}^*), \exp^{-1}_{\boldsymbol{m}^*}(\boldsymbol{m}_0)\rangle + \frac{L}{2}\|\exp^{-1}_{\boldsymbol{m}^*}(\boldsymbol{m}_0)\|^2 \\ &= \langle \nabla f(\boldsymbol{m}^*), \exp^{-1}_{\boldsymbol{m}^*}(\boldsymbol{m}_0)\rangle + \frac{L}{2}d^2_{\mathcal{M}}(\boldsymbol{m}_0, \boldsymbol{m}^*) \leqslant \frac{LD^2}{2}\,. \end{aligned} \tag{5.22}$$

This ensures $\mathbb{E}\Delta_0 \leqslant \frac{LD^2}{2} \leqslant LD^2$ so that we have $\mathbb{E}\Delta_0 \leqslant \frac{D^2}{2\alpha}$ since $0 \leqslant \alpha \leqslant \frac{1}{2L}$, one can obtain from (5.21) that

$$\begin{aligned} \mathbb{E}\Delta_s = \mathbb{E}\{f(\boldsymbol{m}_s) - f(\boldsymbol{m}^*)\} &\leqslant \frac{(1-\varepsilon)^{(s-1)}D^2}{2\alpha} + \sum_{t=0}^{s-1}(1-\varepsilon)^t\frac{\sigma^2}{2} \\ &\leqslant \frac{(1-\varepsilon)^{(s-1)}D^2}{2\alpha} + \sum_{t=0}^{\infty}(1-\varepsilon)^t\frac{\sigma^2}{2} \\ &\leqslant \frac{(1-\varepsilon)^{(s-1)}D^2}{2\alpha} + \frac{\alpha\sigma^2}{2\varepsilon}\,, \end{aligned} \tag{5.23}$$

as desired. □

This proof is based on certain results in [Boumal 2023a] and the trigonometric distance bound, specifically, Corollary 8 in [Zhang 2016b]. However, it is important to note that Theorem 5.2.1 differs from Theorems 14 (diminishing stepsizes) and 15 (deterministic optimization) in [Zhang 2016b]. In our case, we consider a stochastic optimization method with a constant stepsize to compute the CPD statistics g_t. If f is geodesically strongly convex and smooth and the manifold satisfies the conditions in Theorem 5.2.1, convergence can be guaranteed for sufficiently small stepsizes α.

5.2.3.2 Performance guarantee

We now provide two performance guarantees of our CPD statistics g_t as defined in (5.6). These guarantees consist of an upper bound on the false alarm rate under the null hypothesis (i.e., when no changepoint has occurred) and a lower bound on the detection rate under the alternative hypothesis.

Theorem 5.2.2. *We assume that, under the null hypothesis* H_0*,* $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1}$ *are drawn i.i.d. from* $P(\boldsymbol{x})$ *with the Karcher mean* $\boldsymbol{m}^*$*. We also assume that the conditions in Theorem 5.2.1 on* $f(\boldsymbol{m})$*,* $H(\boldsymbol{m}, \boldsymbol{x}_t)$*,* $\mathcal{M}$ *and the stepsizes* λ *and* Λ *hold. At a steady state (i.e., when* $t \to \infty$*), the false alarm rate can be upper bounded by:*

$$\mathbb{P}(g_\infty \geqslant \xi | \mathsf{H}_0) \leqslant \frac{2}{\xi} \left(f(\boldsymbol{m}^*) + \frac{(\lambda + \Lambda)\sigma^2}{4\varepsilon} \right)^{\frac{1}{2}} , \tag{5.24}$$

with $\varepsilon = \min\left\{ \frac{1}{\zeta(\kappa, D)}, \lambda\mu \right\}$ *and* ξ *the detection threshold.*

Proof. Using Markov's inequality with $\xi > 0$,

$$\mathbb{P}(g_t \geqslant \xi | H_0) \leqslant \frac{1}{\xi} \mathbb{E}\{g_t | \mathsf{H}_0\} . \tag{5.25}$$

Now, it remains to find an upper bound to $\mathbb{E}\{g_t|\mathsf{H}_0\}$. Let us ignore the conditioning of the expectation on H_0 to simplify the notation. The rest of the analysis is built upon the triangle inequality and the definition of g_t, which is,

$$g_t = d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{m}_{\Lambda,t}) \leqslant d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{x}) + d_{\mathcal{M}}(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{x}) \tag{5.26}$$

for any $\boldsymbol{x} \in \mathcal{M}$. Take the expectation w.r.t. $\{\boldsymbol{x}_s\}_{s=0}^{t-1}$, with Theorem 5.2.1, Jensen's inequality and the fact $\left(\frac{\sqrt{a}+\sqrt{b}}{2}\right)^2 \leqslant \frac{a+b}{2}$ for nonnegative a and b, we can upper bound $\mathbb{E}\{g_t\}$ as

$$\begin{aligned}
\mathbb{E}\{g_t\} &\leqslant \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{x})\} + \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{x})\} && (5.27)\\
&\leqslant \mathbb{E}\{d^2_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{x})\}^{\frac{1}{2}} + \mathbb{E}\{d^2_{\mathcal{M}}(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{x})\}^{\frac{1}{2}} && (5.28)\\
&= (\mathbb{E}\{f(\boldsymbol{m}_{\lambda,t})\})^{\frac{1}{2}} + (\mathbb{E}\{f(\boldsymbol{m}_{\Lambda,t})\})^{\frac{1}{2}} && (5.29)\\
&\leqslant \left(f(\boldsymbol{m}^*) + \frac{(1-\varepsilon)^{t-1} D^2}{2\lambda} + \frac{\lambda\sigma^2}{2\varepsilon} \right)^{\frac{1}{2}} + \left(f(\boldsymbol{m}^*) + \frac{(1-\varepsilon')^{t-1} D^2}{2\Lambda} + \frac{\Lambda\sigma^2}{2\varepsilon'} \right)^{\frac{1}{2}} && (5.30)\\
&\leqslant 2 \left(f(\boldsymbol{m}^*) + \frac{(1-\varepsilon)^{t-1}(\lambda + \Lambda) D^2}{4\lambda\Lambda} + \frac{(\lambda + \Lambda)\sigma^2}{4\varepsilon} \right)^{\frac{1}{2}} , && (5.31)
\end{aligned}$$

with $\varepsilon' = \min\{\frac{1}{\zeta(\kappa, D)}, \Lambda\mu\}$ satisfying $\varepsilon' \geqslant \varepsilon$ due to the stepsize condition $\lambda < \Lambda$. Taking the limit as $t \to \infty$, we get the following bound for $\mathbb{E}\{g_t\}$ at steady state:

$$\lim_{t\to\infty} \mathbb{E}\{g_t\} \leqslant 2 \left(f(\boldsymbol{m}^*) + \frac{(\lambda + \Lambda)\sigma^2}{4\varepsilon} \right)^{\frac{1}{2}} . \tag{5.32}$$

Combining this bound with (5.25) we obtain the desired result. □

Theorem 5.2.2 shows that when no change occurs, a higher detection threshold ξ leads to a lower upper bound on the false alarm rate. It is worth noting that the bound on the false alarm rate is influenced by the Karcher variance term, which implies that

the bound will be tighter when the data distribution has lower dispersion. Smaller values of λ and Λ are also recommended for a tighter bound because they reduce the impact of gradient noise captured by σ^2. However, choosing larger detection thresholds and smaller stepsizes also reduces the probability of detecting an actual changepoint, as indicated by the following theorem.

Theorem 5.2.3. *We assume that, under the alternative hypothesis* H_1, $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-B-1}$ *are drawn i.i.d. from* $P_1(\boldsymbol{x})$ *with Karcher mean* $\boldsymbol{m}_1^*$, *and* $\boldsymbol{x}_{t-B}, \boldsymbol{x}_{t-B+1}, \ldots, \boldsymbol{x}_{t-1}$ *are drawn i.i.d. from* $P_2(\boldsymbol{x})$ *with Karcher mean* $\boldsymbol{m}_2^*$ *(*$\boldsymbol{m}_1^* \neq \boldsymbol{m}_2^*$*). We also assume that the conditions in Theorem 5.2.1 on* $f(\boldsymbol{m})$, $H(\boldsymbol{m}, \boldsymbol{x}_t)$, *the manifold* $\mathcal{M}$ *and the stepsizes* λ *and* Λ *hold, and that* t *is sufficiently large such that the algorithms converged before the changepoint. Then, the detection rate can be lower bounded as*:

$$\mathbb{P}(g_t > \xi | \mathsf{H}_1) \geqslant \frac{d_{\mathcal{M}}(\boldsymbol{m}_1^*, \boldsymbol{m}_2^*) - \psi(\lambda) - \phi(\Lambda) - \xi}{D - \xi}, \tag{5.33}$$

$$\textit{where} \quad \psi(\lambda) = \left(2f_{\text{bef}}(\boldsymbol{m}_1^*) + \frac{\lambda\sigma^2}{\varepsilon}\right)^{\frac{1}{2}} + \lambda\rho B,$$

$$\phi(\Lambda) = \left(2f_{\text{aft}}(\boldsymbol{m}_2^*) + \frac{(1-\varepsilon)^B D^2}{\Lambda} + \frac{\Lambda\sigma^2}{\varepsilon}\right)^{\frac{1}{2}},$$

with $\varepsilon = \min\left\{\frac{1}{\zeta(\kappa, D)}, \lambda\mu\right\}$, $f_{\text{bef}}(\boldsymbol{m}_1^*) = \min_{\boldsymbol{m}\in\mathcal{M}} \mathbb{E}_{\boldsymbol{x}\sim P_1(\boldsymbol{x})}\{d_{\mathcal{M}}^2(\boldsymbol{m}, \boldsymbol{x})\}$ *and* $f_{\text{aft}}(\boldsymbol{m}_2^*) = \min_{\boldsymbol{m}\in\mathcal{M}} \mathbb{E}_{\boldsymbol{x}\sim P_2(\boldsymbol{x})}\{d_{\mathcal{M}}^2(\boldsymbol{m}, \boldsymbol{x})\}$ *the Karcher variances of the data before and after the changepoint.*

Proof. Let us ignore the conditioning of the expectation on H_1 to simplify the notation. Since the diameter of the domain is bounded above by D, $g_t \leqslant D$, thus, we can apply Markov's inequality to the nonnegative random variable $D - g_t$ to obtain

$$\mathbb{P}(D - g_t \geqslant D - \xi) \leqslant \frac{D - \mathbb{E}\{g_t\}}{D - \xi}, \tag{5.34}$$

which leads to

$$\mathbb{P}(g_t > \xi) \geqslant \frac{\mathbb{E}\{g_t\} - \xi}{D - \xi}. \tag{5.35}$$

We now have to lower bound $\mathbb{E}\{g_t\}$. Using the reverse triangle inequality:

$$\mathbb{E}\{g_t\} = \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{m}_{\Lambda,t})\} \geqslant \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{m}_2^*)\} - \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{m}_2^*)\}, \tag{5.36}$$

with $\boldsymbol{m}_2^*$ being the Karcher mean after the changepoint.

Notice the procedure of optimizing the Karcher mean loss function $f_{\text{aft}}(\boldsymbol{m}) = \mathbb{E}_{\boldsymbol{x}\sim P_2(\boldsymbol{x})}\{d_{\mathcal{M}}^2(\boldsymbol{m}, \boldsymbol{x})\}$ after the changepoint (i.e., where the expectation is defined w.r.t. $P_2(\boldsymbol{x})$), with solution $\boldsymbol{m}_2^*$, by the SGD algorithms (5.4) and (5.5) can be recognized as started from $\boldsymbol{x}_{t-B}$. Let us take the expectation w.r.t. $\{\boldsymbol{x}_s\}_{s=t-B}^{t-1}$ in the following steps.

Now we can upper bound $\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{m}_2^*)\}$ with Jensen's inequality, Theorem 5.2.1, and the fact $\left(\frac{\sqrt{a}+\sqrt{b}}{2}\right)^2 \leqslant \frac{a+b}{2}$ for nonnegative a and b, leading to

$$\begin{aligned}
\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{m}_2^*)\} &\leqslant \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{x})\} + \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_2^*, \boldsymbol{x})\} && (5.37)\\
&\leqslant \mathbb{E}\{d^2_{\mathcal{M}}(\boldsymbol{m}_{\Lambda,t}, \boldsymbol{x})\}^{\frac{1}{2}} + \mathbb{E}\{d^2_{\mathcal{M}}(\boldsymbol{m}_2^*, \boldsymbol{x})\}^{\frac{1}{2}} && (5.38)\\
&\leqslant \left(f_{\text{aft}}(\boldsymbol{m}_2^*) + \frac{(1-\varepsilon')^B D^2}{2\Lambda} + \frac{\Lambda\sigma^2}{2\varepsilon'}\right)^{\frac{1}{2}} + (f_{\text{aft}}(\boldsymbol{m}_2^*))^{\frac{1}{2}} && (5.39)\\
&\leqslant \left(f_{\text{aft}}(\boldsymbol{m}_2^*) + \frac{(1-\varepsilon)^B D^2}{2\Lambda} + \frac{\Lambda\sigma^2}{2\varepsilon}\right)^{\frac{1}{2}} + (f_{\text{aft}}(\boldsymbol{m}_2^*))^{\frac{1}{2}} && (5.40)\\
&\leqslant \left(2f_{\text{aft}}(\boldsymbol{m}_2^*) + \frac{(1-\varepsilon)^B D^2}{\Lambda} + \frac{\Lambda\sigma^2}{\varepsilon}\right)^{\frac{1}{2}}, && (5.41)
\end{aligned}$$

where $\boldsymbol{x} \sim P_2(\boldsymbol{x})$, and $\varepsilon' = \min\{\frac{1}{\zeta(\kappa,D)}, \Lambda\mu\}$ satisfying $\varepsilon' \geqslant \varepsilon$ due to the stepsize condition $\lambda < \Lambda$.

To lower bound $\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{m}_2^*)\}$, we can use the reverse triangle inequality, which gives us

$$\begin{aligned}
\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{m}_2^*)\} &\geqslant \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-1}, \boldsymbol{m}_2^*)\} - \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-1}, \boldsymbol{m}_{\lambda,t})\}\\
&\geqslant \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-B-1}, \boldsymbol{m}_2^*)\} - \sum_{u=t-B}^{t} \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,u-1}, \boldsymbol{m}_{\lambda,u})\}. \qquad (5.42)
\end{aligned}$$

Using the stochastic gradient update equation, $\boldsymbol{m}_{\lambda,t} = \exp_{\boldsymbol{m}_{\lambda,t-1}}(-\lambda H(\boldsymbol{m}_{\lambda,t-1}, \boldsymbol{x}_{t-1}))$, we can express $d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,u-1}, \boldsymbol{m}_{\lambda,u})$ as:

$$d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,u-1}, \boldsymbol{m}_{\lambda,u}) = d_{\mathcal{M}}\big(\boldsymbol{m}_{\lambda,u-1}, \exp_{\boldsymbol{m}_{\lambda,u-1}}(-\lambda H(\boldsymbol{m}_{\lambda,u-1}, \boldsymbol{x}_{u-1}))\big)\,. \qquad (5.43)$$

Since the injectivity radius of the manifold is assumed to be globally bounded above by I, the condition $\lambda \leqslant \frac{I}{\rho}$ implies that $\|\lambda H(\boldsymbol{m}_{\lambda,u-1}, \boldsymbol{x}_{u-1})\| < I$. Thus, by proposition 10.22 of [Boumal 2023a],

$$\begin{aligned}
d_{\mathcal{M}}\big(\boldsymbol{m}_{\lambda,u-1}, \exp_{\boldsymbol{m}_{\lambda,u-1}}(-\lambda H(\boldsymbol{m}_{\lambda,u-1}, \boldsymbol{x}_{u-1}))\big) &= \lambda\|H(\boldsymbol{m}_{\lambda,u-1}, \boldsymbol{x}_{u-1})\| && (5.44)\\
&\leqslant \rho\lambda\,. && (5.45)
\end{aligned}$$

The term $\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-B-1}, \boldsymbol{m}_2^*)\}$ can be lower bounded as

$$\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-B-1}, \boldsymbol{m}_2^*)\} \geqslant d_{\mathcal{M}}(\boldsymbol{m}_1^*, \boldsymbol{m}_2^*) - \mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-B-1}, \boldsymbol{m}_1^*)\}\,, \qquad (5.46)$$

with $\boldsymbol{m}_1^*$ being the Karcher mean of distribution $P_1(\boldsymbol{x})$ of the data before the changepoint.

Knowing that the changepoint occurred at time $t-B$, and since the algorithms are assumed to have asymptotically converged before the changepoint happened (i.e., $t-B-1$ is large), we can upper bound $\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-B-1}, \boldsymbol{m}_1^*)\}$ in (5.46) using

Jensen's inequality, Theorem 5.2.1, and the fact $\left(\frac{\sqrt{a}+\sqrt{b}}{2}\right)^2 \leqslant \frac{a+b}{2}$ for nonnegative a and b, which gives us

$$\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-B-1}, \boldsymbol{m}_1^*)\} \leqslant \mathbb{E}\{d^2_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t-B-1}, \boldsymbol{x}')\}^{\frac{1}{2}} + \mathbb{E}\{d^2_{\mathcal{M}}(\boldsymbol{m}_1^*, \boldsymbol{x}')\}^{\frac{1}{2}} \tag{5.47}$$

$$\leqslant \left(f_{\text{bef}}(\boldsymbol{m}_1^*) + \frac{(1-\varepsilon)^{t-B-1}D^2}{2\lambda} + \frac{\lambda\sigma^2}{2\varepsilon}\right)^{\frac{1}{2}} + (f_{\text{bef}}(\boldsymbol{m}_1^*))^{\frac{1}{2}} \tag{5.48}$$

$$\leqslant \left(2f_{\text{bef}}(\boldsymbol{m}_1^*) + \frac{\lambda\sigma^2}{\varepsilon}\right)^{\frac{1}{2}}, \tag{5.49}$$

where $\boldsymbol{x}' \sim P_1(\boldsymbol{x})$ and the expectation above is now taken w.r.t. the distribution $P_1(\boldsymbol{x})$, before the changepoint; we used that fact $(1-\varepsilon)^{t-B-1} \to 0$ due to the large $t-B-1$, and $f_{\text{bef}}(\boldsymbol{m}) = \mathbb{E}_{\boldsymbol{x}\sim P_1(\boldsymbol{x})}\{d^2_{\mathcal{M}}(\boldsymbol{m}, \boldsymbol{x})\}$ denotes the Karcher mean loss function before the changepoint (i.e., where the expectation is defined w.r.t. $P_1(\boldsymbol{x})$), with solution $\boldsymbol{m}_1^*$.

Combining the bounds in (5.42), (5.45), (5.46), (5.49) leads to the following lower bound:

$$\mathbb{E}\{d_{\mathcal{M}}(\boldsymbol{m}_{\lambda,t}, \boldsymbol{m}_2^*)\} \geqslant d_{\mathcal{M}}(\boldsymbol{m}_1^*, \boldsymbol{m}_2^*) - \left(2f_{\text{bef}}(\boldsymbol{m}_1^*) + \frac{\lambda\sigma^2}{\varepsilon}\right)^{\frac{1}{2}} - \rho\lambda B\,. \tag{5.50}$$

Finally, combining the bounds (5.35), (5.36), (5.41) and (5.50), we obtain

$$\mathbb{P}(g_t > \xi) \geqslant \frac{1}{D-\xi}\Bigg[d_{\mathcal{M}}(\boldsymbol{m}_1^*, \boldsymbol{m}_2^*) - \left(2f_{\text{bef}}(\boldsymbol{m}_1^*) + \frac{\lambda\sigma^2}{\varepsilon}\right)^{\frac{1}{2}} - \rho\lambda B$$
$$- \left(2f_{\text{aft}}(\boldsymbol{m}_2^*) + \frac{(1-\varepsilon)^B D^2}{\Lambda} + \frac{\Lambda\sigma^2}{\varepsilon}\right)^{\frac{1}{2}} - \xi\Bigg], \tag{5.51}$$

which is the desired result. □

Theorem 5.2.3 shows that smaller values of ξ and larger values of $d_{\mathcal{M}}(\boldsymbol{m}_1^*, \boldsymbol{m}_2^*)$ make the lower bound on the detection rate tighter when a changepoint occurs. Moreover, the bound also gets tighter as λ gets smaller and Λ gets bigger, which is intuitive since, when B is not too large, a small λ assures $\boldsymbol{m}_{\lambda,t}$ will still be close to the Karcher mean of the data before the changepoint, whereas a large Λ means that $\boldsymbol{m}_{\Lambda,t}$ will converge faster to the Karcher mean of the data after the changepoint, their distance being thus more effective for change detection. However, one should note that λ being too small can hurt the adaptability of the method and its capability to detect multiple changepoints. Thus, the stepsizes should be selected to ensure a sufficiently fast speed of convergence for the desired application.

The increase in the number of samples B after a changepoint has a twofold effect on the lower bound to the detection rate in (5.33). On the one hand, the estimate of the Karcher means before the changepoint from the "slow" algorithm gets polluted by samples following the post-change distribution, causing the term $\psi(\lambda)$ to *increase*

Algorithm 7 Adaptive threshold selection

Input : $\{g_t\}$, forgetting factor α, quantile q.
Initialization : $\beta_t^g = g_1$, $\gamma_t^g = g_1^2$.
for $t = 1, 2, 3, \ldots$ **do**
$\beta_t^g = (1-\alpha)\beta_{t-1}^g + \alpha g_t$;
$\gamma_t^g = (1-\alpha)\gamma_{t-1}^g + \alpha g_t^2$;
$\hat{\xi}_t = \beta_t^g + \sqrt{\gamma_t^g - (\beta_t^g)^2}\sqrt{2}\text{erf}^{-1}(2q-1)$;

with B. On the other hand, the "fast" algorithm will converge to the Karcher means of the post-change data, causing the term $\phi(\Lambda)$ to *decrease* with B. The bound also gets larger as the Karcher variances of the data, the gradient noise, and the bound on the diameter of the domain decrease. Note that these quantities are the main sources of stochasticity in the proposed algorithm, and as the uncertainty decreases the theoretical detection performance of the algorithm improves. A similar behavior is also observed for the upper bound to the false alarm rate in (5.24).

5.2.4 Adaptive threshold selection

One challenge in applying CPD algorithms is the selection of the detection threshold ξ without prior knowledge of the data distribution. In real use cases, a simple yet effective procedure is to adjust ξ so as to achieve some desired performance in the absence of changepoints. A classical approach consists of adjusting ξ such that the algorithm achieves some desired performance under the null hypothesis (i.e., in the absence of changepoints), such as a given probability of false alarms [Keriven 2020]. For a false alarm rate of, e.g., 0.05, ξ can be set as the 95-th quantile of g_t. The performance of the algorithm under the null hypothesis can be computed using training data or based on a theoretical analysis, such as the result given in Theorem 5.2.2. However, threshold selection approaches based on theoretical analyses are hard to apply in practice as they require strong prior knowledge of the statistical distribution of the data, such as the Karcher variance $f(\boldsymbol{m})$ and gradient noise σ^2 in our case.

A more practical approach is to set ξ as an estimate of the q-th quantile of g_t obtained using a recursive algorithm. Although efficient algorithms have been proposed for recursive quantile estimation [Chen 2023], we use a simpler alternative by approximating g_t by a Gaussian distribution (the validity of this hypothesis illustrated empirically in FIGURE 5.3), as also done in [Keriven 2020]. This way, computing only its first two moments is sufficient to compute the q-th quantile, which is given by the mean plus the standard deviation multiplied by $\sqrt{2}\text{erf}^{-1}(2q-1)$, where erf is the Gauss error function. A simple recursive implementation of this strategy is shown in Algorithm 7, which is based on EWMAs of the first two moments of g_t. Experiments illustrating the validity of the Gaussian hypothesis over g_t and the performance of Algorithm 7 can be found in Subsection 5.4.1.

5.3 Application to specific manifolds

In this section, we tailor Algorithm 6 to two common instances of Riemannian manifolds for the case of the Karcher mean cost function $c = d^2_{\mathcal{M}}$, which will later be illustrated through numerical experiments in Section 5.4. The first one is the manifold of $p \times p$ SPD matrices, denoted by $\mathcal{S}_p^{++}$. The second is the Grassmann manifold, a set of k-dimensional linear subspaces of $\mathbb{R}^p$, denoted by $\mathcal{G}_p^k$. We refer the interested reader to [Boumal 2023a, Collas 2022] for more details. Note that although $\mathcal{G}_p^k$ is not a Hadamard manifold, Algorithm 6 still performs empirically well as will be presented in Section 5.4. In practice, these manifolds can appear as natural representations of the data (e.g., in diffusion tensor imaging) or as feature embeddings thereof. For computational simplicity, we will replace the exponential maps in the R-SGD updates (5.4) and (5.5) with approximate retractions $R_{\boldsymbol{m}_{\lambda,t}}$ and $R_{\boldsymbol{m}_{\Lambda,t}}$ as in [Bonnabel 2013]. A systematic study of retraction to understand the effect of this approximation on convergence rate and performance guarantees of our method is an important topic for future research.

5.3.1 The manifold of SPD matrices

The manifold $\mathcal{S}_p^{++}$ consists of the set of SPD matrices endowed with an appropriate metric. When considering the affine invariant metric, the geodesic distance between two SPD matrices $\boldsymbol{\Sigma}$ and $\boldsymbol{\Sigma}_t \in \mathcal{S}_p^{++}$ can be computed as [Pennec 2006b]:

$$d_{\mathcal{S}_p^{++}}(\boldsymbol{\Sigma}, \boldsymbol{\Sigma}_t) = \left\| \log(\boldsymbol{\Sigma}_t^{-\frac{1}{2}} \boldsymbol{\Sigma} \boldsymbol{\Sigma}_t^{-\frac{1}{2}}) \right\|_F , \tag{5.52}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. In this case, the Riemannian gradient $H(\boldsymbol{\Sigma}, \boldsymbol{\Sigma}_t)$ of the loss function $d^2_{\mathcal{S}_p^{++}}(\boldsymbol{\Sigma}, \boldsymbol{\Sigma}_t)$ at $\boldsymbol{\Sigma} \in \mathcal{S}_p^{++}$ is obtained by applying the transformation $\frac{1}{2}\boldsymbol{\Sigma}(\boldsymbol{G}^T + \boldsymbol{G})\boldsymbol{\Sigma}$ to its Euclidean gradient $\boldsymbol{G}$ [Bhatia 2009], which gives us :

$$H(\boldsymbol{\Sigma}, \boldsymbol{\Sigma}_t) = 2 \log(\boldsymbol{\Sigma}\boldsymbol{\Sigma}_t^{-1})\boldsymbol{\Sigma} . \tag{5.53}$$

Finally, a second-order retraction on $\mathcal{S}_p^{++}$ is given by:

$$R_{\boldsymbol{\Sigma}, \mathcal{S}_p^{++}}(\boldsymbol{\xi}) = \boldsymbol{\Sigma} + \boldsymbol{\xi} + \frac{1}{2}\boldsymbol{\xi}\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi} . \tag{5.54}$$

With $\{\boldsymbol{\Sigma}_t\}_{t \in \mathbb{N}}$ lying in $\mathcal{S}_p^{++}$ and the metric defined in (5.52), the Karcher means were estimated by minimizing the following objective function

$$f(\boldsymbol{\Sigma}) = \mathbb{E}_{\boldsymbol{\Sigma}_t \sim P(\boldsymbol{\Sigma})} \left\{ \left\| \log(\boldsymbol{\Sigma}_t^{-\frac{1}{2}} \boldsymbol{\Sigma} \boldsymbol{\Sigma}_t^{-\frac{1}{2}}) \right\|_F^2 \right\} , \tag{5.55}$$

Note this cost function is known to be geodecially strong convex and smooth as discussed in [Zhang 2016b]. The R-SGD algorithms in (5.4) and (5.5) with the stochastic gradient (5.53) and the retraction (5.54) were used to compute the online CPD statistic in (5.6).

5.3.2 The Grassmann manifold

We consider the Grassmann manifold $\mathcal{G}_p^k$ endowed with the canonical metric. The Grassmann manifold is typically characterized as a smooth quotient of the Stiefel manifold $\mathcal{S}_p^k = \{\boldsymbol{U} \in \mathbb{R}^{p\times k} : \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_k\}$. This way, by defining the surjective map $\pi : \mathcal{S}_p^k \to \mathcal{G}_p^k$ as follows: $\pi(\boldsymbol{U}) = \{\boldsymbol{U}\boldsymbol{O} : \boldsymbol{O} \in \mathbb{R}^{k\times k}, \boldsymbol{O}^T\boldsymbol{O} = \boldsymbol{I}_k\}$, every point $\pi(\boldsymbol{U}) \in \mathcal{G}_p^k$ can be equivalently represented by the orthonormal matrix $\boldsymbol{U}$ whose columns form its basis. We spare the reader of the technical details, which can be found in [Absil 2009, Boumal 2023a]. To proceed, let us first denote by $\boldsymbol{V}_1 \operatorname{Diag}(\boldsymbol{\theta}_t)\boldsymbol{V}_2^T$ the singular value decomposition (SVD) of $\boldsymbol{U}^T\boldsymbol{U}_t$. The geodesic distance between $\pi(\boldsymbol{U}) \in \mathcal{G}_p^k$ and $\pi(\boldsymbol{U}_t) \in \mathcal{G}_p^k$ can be defined as [Edelman 1998]:

$$d_{\mathcal{G}_p^k}(\boldsymbol{U}, \boldsymbol{U}_t) = \|\cos^{-1}(\boldsymbol{\theta}_t)\|_2 \,. \tag{5.56}$$

The Riemmanian gradient $H(\boldsymbol{U}, \boldsymbol{U}_t)$ of the loss function $d^2_{\mathcal{G}_p^k}(\boldsymbol{U}, \boldsymbol{U}_t)$ at $\pi(\boldsymbol{U}) \in \mathcal{G}_p^k$ can be computed by applying the transformation $(\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^T)\boldsymbol{G}$ to its Euclidean gradient $\boldsymbol{G}$. Using results from matrix calculus, this results in:

$$H(\boldsymbol{U}, \boldsymbol{U}_t) = -(\boldsymbol{I} - \boldsymbol{U}\boldsymbol{U}^T)\boldsymbol{U}_t\boldsymbol{V}_2 \operatorname{Diag}\left(2\left(1 - \boldsymbol{\theta}_t^2\right)^{-\frac{1}{2}}\right)\boldsymbol{V}_1^T. \tag{5.57}$$

Let $\boldsymbol{\xi} \in T_{\pi(\boldsymbol{U})}\mathcal{G}_p^k$, and let $\boldsymbol{X}\boldsymbol{\Upsilon}\boldsymbol{Y} = \boldsymbol{U} + \boldsymbol{\xi}$ be the thin SVD of $\boldsymbol{U} + \boldsymbol{\xi} \in \mathbb{R}^{n\times p}$. A second-order retraction on the Grassmann manifold is given by [Boumal 2023a]

$$R_{\pi(\boldsymbol{U})}(\boldsymbol{\xi}) = \pi\left(\boldsymbol{X}\boldsymbol{Y}^T\right). \tag{5.58}$$

With $\{\pi(\boldsymbol{U}_t)\}_{t\in\mathbb{N}}$ lying in $\mathcal{G}_p^k$ and the metric defined in (5.56), the Karcher means were estimated by minimizing the objective function

$$f(\pi(\boldsymbol{U})) = \mathbb{E}_{\pi(\boldsymbol{U}_t)\sim P(\pi(\boldsymbol{U}))}\left\{\|\cos^{-1}(\boldsymbol{\theta}_t)\|_2^2\right\}. \tag{5.59}$$

Accordingly, the R-SGD algorithms in (5.4) and (5.5) with the stochastic gradient (5.57) and the retraction (5.58) were used to compute the online CPD statistic in (5.6).

5.4 Experiments

In this section, we present numerical experiments using the manifolds $\mathcal{S}_p^{++}$ and $\mathcal{G}_p^k$ discussed in Section 5.3. Our method was implemented in Python using Pymanopt [Townsend 2016]. The selection of stepsizes based on theoretical analyses is not trivial to apply in our approach as they require strong prior knowledge of the statistical distribution of the data, such as the Karcher variance $f(\boldsymbol{m}^*)$ and gradient noise σ^2. Instead, we set the stepsizes of our method as $\lambda = 0.01$ and $\Lambda = 0.02$ with empirical evaluation. Open-source code to reproduce the results is publicly available at `https://github.com/xiuheng-wang/CPD_manifold_release`. Here we briefly describe the baselines and evaluation metrics.

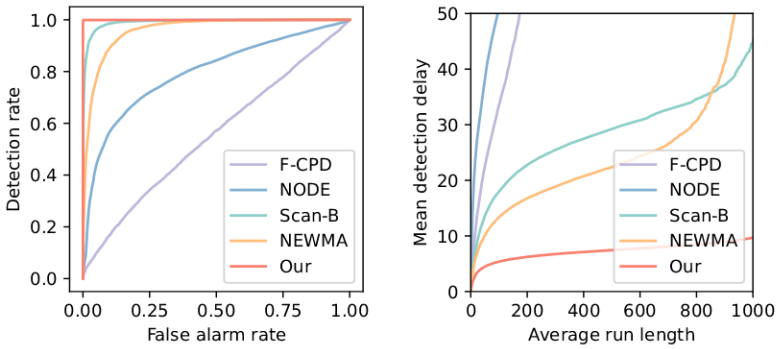

FIGURE 5.1 – ROC curves, ARL versus MDD for the compared algorithms on synthetic data on $\mathcal{S}_p^{++}$.

Baselines: We selected four CPD methods Scan-B [Li 2019], NEWMA [Keriven 2020], the Fréchet CPD (F-CPD) [Dubey 2020] and NODE [Wang 2023b] as baselines for comparison with our method. Scan-B, NEWMA, and NODE are online algorithms originally designed for Euclidean spaces but were adapted to the manifold setting in this study. We applied Scan-B, NEWMA, and NODE to the vectorization of the lower triangular portion of each SPD matrix $\boldsymbol{\Sigma}_t$ and to each entire matrix $\boldsymbol{U}_t$ for the SPD and Grassmann manifolds, respectively. In Scan-B, the number of reference blocks was set to 3. NEWMA was implemented with Random Fourier features using the Gaussian kernel. The window size of Scan-B and NEWMA were both set to 50. The reference and test window lengths of NODE were both set to 64. F-CPD was designed to operate on manifolds but can only detect a single changepoint and operates offline. To address these limitations, we computed statistics in F-CPD to compare data distributions in two consecutive sliding windows, each with 64 samples.

Metrics: To evaluate the performance of the methods, we considered three metrics : the Average Run Length (ARL), Mean Detection Delay (MDD), and Receiver Operating Characteristic (ROC) curves. ARL represents the expected time before incorrectly announcing a changepoint when none has occurred, and is related to the false alarm rate. MDD is the expected time the algorithm needs to flag a detection after a changepoint occurs, reflecting its sensitivity. The ROC curve is a graphical representation of the detection rate versus the false alarm rate.

5.4.1 Experiments with synthetic data

We first present results over sequences of i.i.d. synthetically generated data in $\mathcal{S}_p^{++}$ and $\mathcal{G}_p^k$.

Manifold $\mathcal{S}_p^{++}$: We sampled matrices $\boldsymbol{\Sigma}_t \in \mathcal{S}_p^{++}$ with $p = 8$ from a Wishart distribution with a randomly generated scaling matrix $\boldsymbol{V}$ and $p + 2$ degrees of

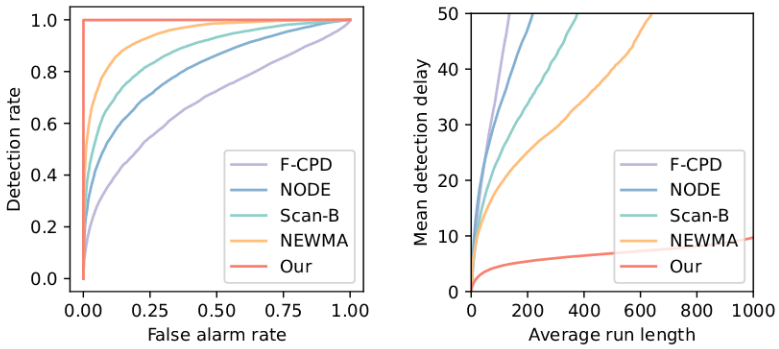

FIGURE 5.2 – ROC curves, ARL versus MDD for the compared algorithms on synthetic data on $\mathcal{G}_p^k$.

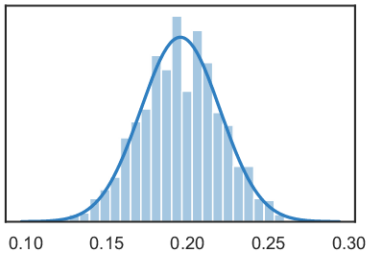

FIGURE 5.3 – Histogram of g_t under the null hypothesis for synthetic data on $\mathcal{S}_p^{++}$ and its Gaussian fit.

freedom. We generated 2000 samples and set a changepoint at $t_r = 1500$ where we randomly reset $\boldsymbol{V}$ from one random matrix to another.

Manifold $\mathcal{G}_p^k$: The data $\pi(\boldsymbol{U}_t) \in \mathcal{G}_p^k$ with $p = 15$, $k = 5$ was generated in two steps. First, we generated matrices $\boldsymbol{Z}_t$ following a matrix Gaussian distribution [Gupta 1999] with random mean and row/column covariance matrices. Then, the orthonormal matrices $\boldsymbol{U}_t$ were generated as the left singular vectors corresponding to the k largest singular values of $\boldsymbol{Z}_t$. We generated 2000 samples and set a changepoint at $t_r = 1500$ where we reset the mean of the matrix Gaussian distribution of $\boldsymbol{Z}_t$.

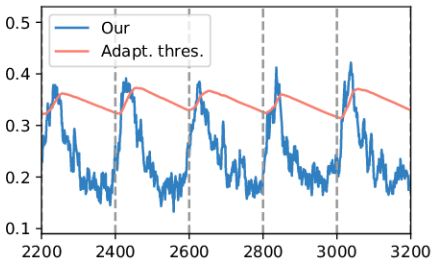

FIGURE 5.4 – Illustration of the adaptive threshold procedure. The dotted gray lines indicate changepoints.

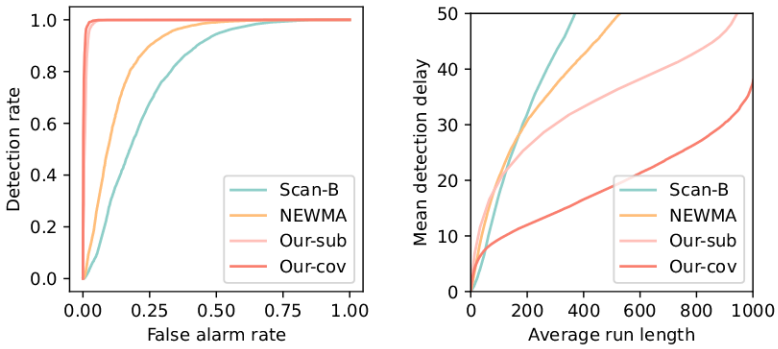

FIGURE 5.5 – ROC curves, ARL versus MDD for the compared algorithms on real data for voice activity detection.

Discussion: The ROCs, MDD as a function of ARL for all methods, averaged over 10^4 Monte Carlo runs, are depicted in FIGURE 5.1 and 5.2 for both manifolds. It is evident that the proposed method results in a significantly lower detection delay for a fixed ARL when compared to Euclidean methods Scan-B, NEWMA, and NODE, which does not consider manifold geometry, and F-CPD, which does not benefit from long time series through a recursive operation. The compared methods exhibited similar behavior in both manifolds, although the proposed method resulted in slightly lower MDDs for $\mathcal{G}_p^k$. This underscores the importance of accounting for manifold geometry and utilizing an efficient online estimation framework. Illustrations of the mean and standard deviation and further comparisons between histograms of the test statistics for all compared methods are provided in Subsection 5.4.5.

Histogram of the test statistic, Gaussian fit: To illustrate the validity of the Gaussian hypothesis of g_t, in FIGURE 5.3, we plot the histogram of g_t for 1000 Monte Carlo runs, computed based on samples of g_t under the null hypothesis when the algorithm is tested for the synthetic example on $\mathcal{S}_p^{++}$, after the algorithms converge (with stepsizes $\lambda = 0.01$ and $\Lambda = 0.02$). It can be observed that the histogram and its Gaussian fit are very close, which justifies the approximations in Algorithm 7.

Illustration of the adaptive threshold procedure: We illustrate the performance of Algorithm 7 with $\alpha = 0.005$ and $q = 0.95$ (5% of false alarms). We considered the same setup as in the synthetic example in $\mathcal{S}_p^{++}$, but here we added multiple changepoints, spaced by 200 samples to allow the algorithm to converge. The test statistic and the adaptive threshold are shown in FIGURE 5.4 (results are shown after the steady-state convergence of both the CPD and adaptive threshold selection algorithms), where it can be seen that the dynamic threshold can successfully adapt to detect multiple changepoints in a continuous run.

5.4.2 Voice activity detection

We now present results on real data on both $\mathcal{S}_p^{++}$ and $\mathcal{G}_p^k$ by considering the task of voice activity detection on audio signals. We first added 4 seconds of real speech extracted from the TIMIT database [Garofolo 1993] to 15 seconds of background noises in real street environments from the QUT-NOISE database [Dean 2010], with -3 dB Signal-to-Noise Ratio. The goal is to detect the speech segments in the noise background. Then, we used the Short Time Fourier Transform (STFT) [Cohen 1995] on a one-dimensional audio signal to extract on-the-fly frequency information and form a $d = 128$ dimensional time series $\boldsymbol{s}_t \in \mathbb{R}^d$. The two methods with the best performance in the experiments with synthetic data, Scan-B and NEWMA, were used as baselines in this experiment. They were directly applied on $\boldsymbol{s}_t$ as they are designed to operate on Euclidean spaces.

Manifold $\mathcal{S}_p^{++}$: We averaged the neighboring channels of $\boldsymbol{s}_t$ in the frequency domain to obtain its down-sampled version with 16 channels. We then generated data points $\boldsymbol{\Sigma}_t \in \mathcal{S}_p^{++}$ with $p = 16$ by computing the covariance matrices in sliding windows, each with 32 samples. The proposed method on such covariance descriptors is denoted as “Our-cov".

Manifold $\mathcal{G}_p^k$: We also applied the truncated SVD with $k = 1$ singular values to the samples in the same sliding windows to obtain orthonormal matrices $\boldsymbol{U}_t$ defining the subspaces $\pi(\boldsymbol{U}_t) \in \mathcal{G}_p^k$. We denote our method on these subspaces as “Our-sub".

Discussion: The ROCs and MDD as a function of ARL for all methods, averaged over 10^4 Monte Carlo runs, are depicted in FIGURE 5.5. It is important to note that the problem setting is challenging due to the complexity of real acoustic signals and the non-i.i.d. nature of the extracted features. Nevertheless, one can observe that the proposed strategy exhibits a higher detection rate for a given false alarm rate and better performance on MDD versus ARL when compared to both Scan-B and NEWMA, except for very small ARLs where Scan-B has a lower MDD. This behavior occurs since both the covariance and subspace descriptors are computed over a sliding window, which introduces a small detection delay in our method when the ARL is small [1]. However, its performance is significantly better for larger ARLs. This illustrates the superior performance of our method. Furthermore, the performance was slightly superior in the covariance descriptors on $\mathcal{S}_p^{++}$ compared to the subspace representations on $\mathcal{G}_p^k$. In this application, the performance improvement of our method is also due to the beneficial properties of the covariance matrix, which reduces the impact of vertical noise when detecting changepoints in spectrograms.

1. Although a shorter sliding window is preferred to introduce a smaller delay, the window length has to be long enough to provide enough samples for an accurate estimation of these statistical descriptors.

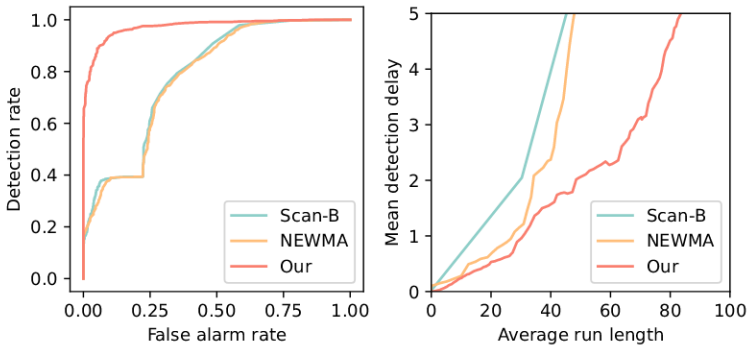

FIGURE 5.6 – ROC curves, ARL versus MDD for the compared algorithms on real data for skeleton-based action recognition.

5.4.3 Skeleton-based action recognition

We also present results on real data on the $\mathcal{S}_p^{++}$ manifold by considering the problem of detecting changepoints in skeleton-based action recognition using the HDM05 motion capture database [Müller 2007]. In this database, we identified action categories and preprocessed the data as described in [Huang 2017] to generate data points $\mathbf{\Sigma}_t \in \mathcal{S}_p^{++}$ with $p = 93$ by computing the joint covariance descriptor [Hussein 2013] of 3D coordinates of the 31 joints. The aim is to flag a changepoint at the border of two different action categories. We randomly selected the sequences corresponding to action categories containing more than 200 samples and then concatenated them. The parameters of the compared algorithms were appropriately re-adjusted for this example since there were fewer samples between changepoints, requiring a faster convergence.

Discussion: The ROC and MDD versus ARL curves of the compared methods (Scan-B, NEWMA, and our algorithm), averaged over 10^3 Monte Carlo runs, can be seen in FIGURE 5.6. Note that the problem setting is challenging due to the high data dimension. It can be seen that our method achieves a significantly higher detection rate compared to the Scan-B and NEWMA, which had very similar ROCs [2]. Moreover, for ARLs smaller than 40 samples, the proposed method and NEWMA obtained similar MDDs. However, when the ARL was higher, the proposed method performed significantly better. This further illustrates the effectiveness of our method.

5.4.4 Computational complexity

The computational complexity of our method consists mainly of the cost of implementing the two R-SGD algorithms used to estimate the generalized Karcher means. The R-SGD algorithm is a first-order method that is computationally efficient compared to other manifold optimization algorithms. It comprises two main steps:

2. The mean and standard deviation of the test statistic of all methods for this example can also be seen in Subsection 5.4.5.

1) computation of the Riemannian gradient of the loss function, and 2) computing the exponential or retraction to map the gradient back to the manifold.

The computational complexity involved with these steps depends on the choice of the manifold as it affects both the loss function (and therefore the gradient) and the retraction/exponential map. However, for many manifolds of great practical interest, including the SPD and the Grassmann, computing these operations is relatively efficient, and for these two manifolds, we can compute the complexity explicitly.

Complexity for the SPD manifold: The operations involved in implementing the R-SGD on the manifold of $p \times p$ SPD matrices consist of five matrix multiplications, a matrix inverse, and a matrix logarithm. Thus, the computational cost is given by $\mathcal{O}(p^3)$ operations.

Complexity for the Grassmann manifold: The operations involved in implementing the R-SGD on the Grassmann manifold of k-dimensional subspaces in $\mathbb{R}^p$ consists of two SVDs, five matrix products, and the evaluation of $\mathcal{O}(k)$ arithmetic functions. Thus, the computational cost is given by $\mathcal{O}(p^2 k)$.

Comparison to baselines: We briefly compare the complexity with respect to the baselines F-CPD [Dubey 2020] and NEWMA [Keriven 2020]. F-CPD is an offline method designed to operate on manifolds and detects a changepoint based on a two-sample test. For every candidate changepoint, the test statistic is computed as a function of the Karcher means and variances of the data before and after the candidate changepoint, which is computationally very intensive to implement. NEWMA, on the other hand, is an online method designed to operate on Euclidean spaces, by comparing exponentially weighted moving averages of generalized moments of the data computed based on the random features framework. Thus, the cost of NEWMA is dominated by the cost of computing the random features [Keriven 2020]. For random Fourier features [Rahimi 2007], the computation complexity scales as $O(Sd)$ operations, where d is the dimension of the input data, and S is the number of random samples (the dimension of the feature space), which are sampled from a probability measure related to the kernel. Thus, depending on the choice of kernel and the feature dimension NEWMA can be efficient, although it does not take the manifold geometry into account.

5.4.5 Additional results

In this subsection, we provide some supplementary results to enhance comprehension of the methodologies, though they are not necessarily required to validate the performance.

5.4.5.1 Mean and standard deviation of the test statistics

In FIGURE 5.7, we plot the mean and standard deviation of the test statistics of all the compared algorithms for the examples with synthetic data. It can be seen that for the synthetic example the test statistic of the proposed strategy required approximately 200 samples to converge after a changepoint occurs. The algorithm achieves good performance for detecting multiple changepoints as long as the interval between them is sufficiently large compared to the time it requires to converge. By comparison, we also plot in FIGURE 5.9 the test statistic for the compared methods for the skeleton-based action recognition example, in which the parameters of the algorithms had to be readjusted to achieve faster convergence since the number of samples between changepoints is smaller. It can be observed that the algorithms converge significantly faster (requiring only approximately 80 samples). However, the variances of the test statistic, particularly after the changepoint, are also much higher. This illustrates the trade-off between detection performance and adaptability of the proposed method.

5.4.5.2 Comparisons between the histograms of the test statistics on synthetic data

To get a deeper insight into the behavior of the ROC curves in the examples with synthetic data (FIGUREs 4.2 and 5.2), where our method had an area under curve close to one, we compared the histograms of the test statistics of all methods under the null hypothesis and at their peak value after a changepoint. The result can be seen in FIGURE 5.8. One can observe that different from the competing methods, the histogram of the test statistic of our method under the null hypothesis shows almost no overlap with its counterpart at peak value after a changepoint. This explains the good detection performance of the proposed detector confirmed by ROC curves.

5.5 Conclusion

This chapter presented a general approach for non-parametric online CPD on Riemannian manifolds. An adaptive test statistic was computed using stochastic Riemannian optimization to monitor the generalized Karcher mean of data streams. Performance guarantees for detection and false alarm rates were established based on a theoretical analysis of the non-asymptotic convergence of the R-SGD algorithm. Experimental results on the manifold of SPD matrices and the Grassmann manifold demonstrated the superiority of the proposed algorithm on synthetic and real-world datasets. We also identify the main limitations of our work:

— The number of samples needs to be large enough for the “slow" algorithm to converge to the Karcher mean of the data before a new changepoint occurs to perform well. This is a limitation of our method and also of other recursive algorithms.

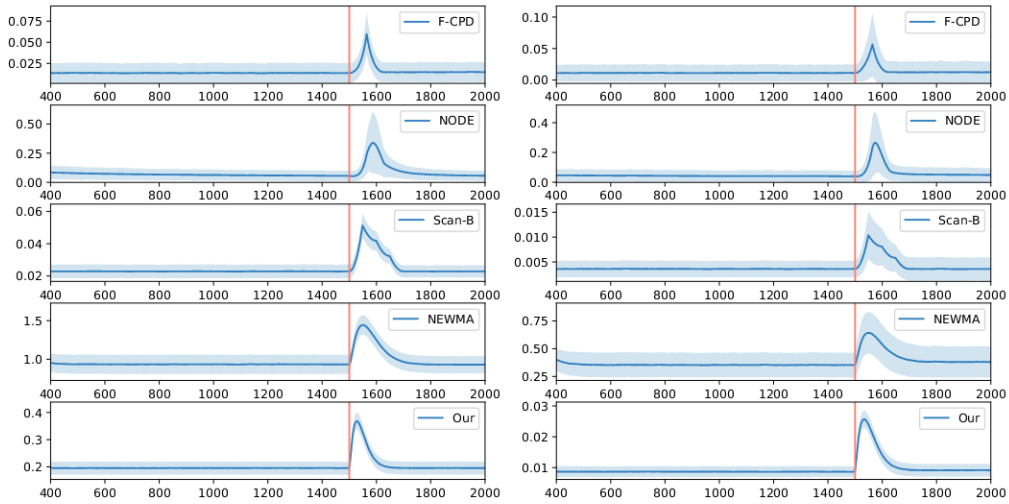

FIGURE 5.7 – Illustration of the mean and standard deviation of all the compared detection statistics for the experiments on synthetic data on both $\mathcal{S}_p^{++}$ (left) and $\mathcal{G}_p^k$ (right). The red line indicates the changepoint.

— Although $\mathcal{S}_p^{++}$ and $\mathcal{G}_p^k$ are selected to illustrate our approach, our framework is more general and can indeed be applied to other manifolds. The main possible hurdle is related to the convergence rate of the R-SGD algorithm affected by the manifold curvature, being slower for higher curvature values. This in turn can negatively impact the detection delay.

— For the theoretical analysis, we make additional assumptions on the manifold (e.g., Hadamard). Although this does not limit the practical applicability to other manifolds, manifolds with complex geometries can introduce additional challenges such as non-convexity of the cost function.

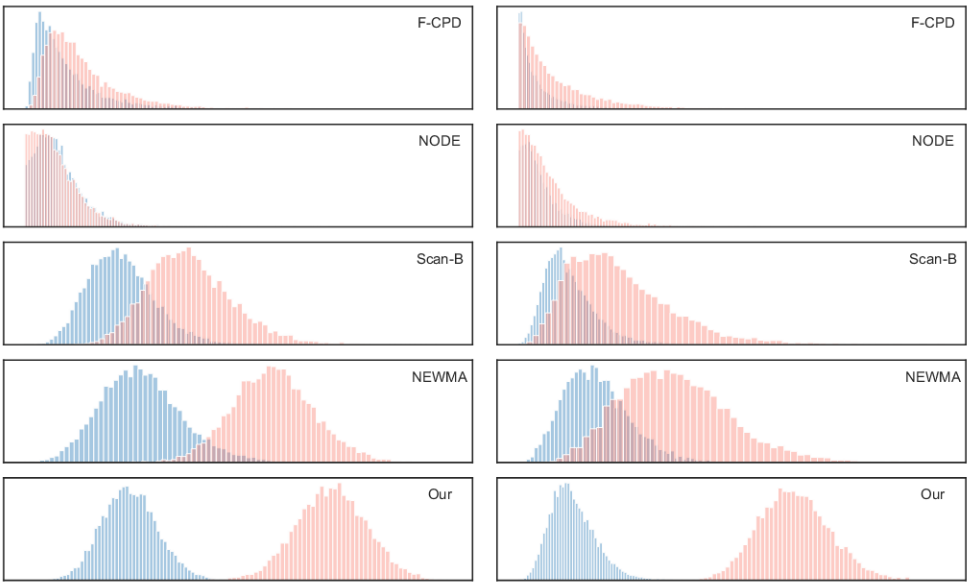

FIGURE 5.8 – Histograms of all the compared detection statistics for the experiments on synthetic data on both $\mathcal{S}_p^{++}$ (left) and $\mathcal{G}_p^k$ (right). The blue histograms are under the null hypothesis and the pink histograms are at their peak values after the changepoint.

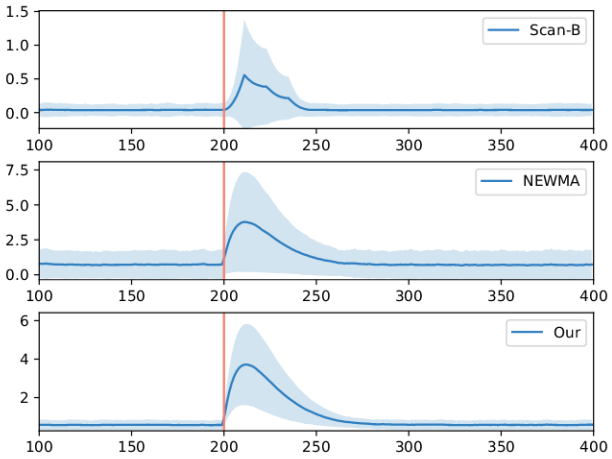

FIGURE 5.9 – Illustration of the mean and standard deviation of the compared detection statistics for the experiments on real data for skeleton-based action recognition. The red line indicates the changepoint.

Chapter 6

Distributed CPD in manifold-valued signals over graphs

Contents

6.1 Introduction

CPD aims to identify time instants when the probability distribution of a stochastic process or a time series changes. Recently, one trend that has been growing in popularity is to detect changepoints in time series measured over the nodes of a network while taking the network topology into account [Sharpnack 2013, Ferrari 2019, Ferrari 2020b]. This problem is of great interest for many applications as diverse as network security, environmental monitoring and neuroimaging.

The changepoints in these problems often occur in groups of highly connected nodes (i.e., communities) of networks, represented as graphs. Graph signal processing tools, including spectral analysis [Ng 2001, Von Luxburg 2007] and filtering [Shuman 2011, Segarra 2015, Loukas 2015, Isufi 2017], are indicated to combine such graph topology information with measurements collected at each node. In [Sharpnack 2013], the authors introduce the Graph Fourier Scan Statistic (GFSS) and a low-pass filter based on graph Fourier transform to detect anomalies over graph signals. The work in [Ferrari 2019] proposes an online CPD algorithm with a fully distributed and adaptive GFSS to monitor for changepoints in large-scale networks. This algorithm was applied for CPD in multi-channel image sequences in [Borsoi 2021f]. An online and distributed strategy based on likelihood ratio estimation with kernel machinery

is also described in [Ferrari 2020b] to detect changepoints over graphs with few assumptions on the data distribution.

All the distributed CPD frameworks listed above are limited to real- or vector-valued time series in Euclidean spaces. To the best of our knowledge, there is no generalization of CPD techniques over graphs where the streaming data collected at each node belongs to a Riemannian manifold. A representative application is the detection of changepoints in videos with a sequence of spatially localized covariance descriptors [Wang 2023a]. In this application, the changes usually affect multiple related regions in the image. However, this information is not taken into account by algorithms that process each region separately. One possibility to leverage this information is to design a graph that describes the relationship between regions, and then perform the detection of changepoints cooperatively. To devise algorithms for manifold-valued data, it is important to take the geometry of the data space into account by exploiting, e.g., an appropriate Riemannian metric [Pennec 2006b, Boumal 2023b].

This chapter introduces a distributed framework for detecting changepoints on manifold-valued signals collected over a network. The proposed method is built upon a test statistic derived for streaming data on a Riemannian manifold which accounts for the geometry of the data, and a fully distributed graph filter that exploits the network topology information to enhance the detection of anomalies localized in unknown communities of nodes. Simulation results show that taking manifold geometry and graph topology into account can significantly improve the detection performance.

6.2 Problem formulation

We consider an undirected graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$ with N vertices in $\mathcal{N} = \{1, \dots, N\}$ and M edges in $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$ such that $(i, j) \in \mathcal{E}$ iff nodes i and j are connected. The graph $\mathcal{G}$ is associated with an $N \times N$ weighted adjacency matrix $\boldsymbol{W}$. Each entry $W_{i,j} \geqslant 0$ is the connection strength between nodes i and j, with non-zero value iff. $(i, j) \in \mathcal{E}$. A community $\mathcal{C} \subset \mathcal{N}$ in $\mathcal{G}$ is a subset of nodes that are densely connected.

At each time instant $t \in \mathbb{N}$, we observe a signal over the graph $\mathcal{X}_t = \{\boldsymbol{x}_t(n)\}_{n=1}^N$, where $\boldsymbol{x}_t(n) \in \mathcal{M}$ denotes the measurement collected at node n, that lies on a Riemannian manifold $(\mathcal{M}, g)$. In this chapter, the objective is to detect an abrupt change in the graph signal $\mathcal{X}_t$ that might occur at an unknown time t_r, called a *changepoint*. In particular, we assume that the changes occur in an unknown community $\mathcal{C}^*$ of $\mathcal{G}$, which means that :

$$\begin{aligned} t < t_r &: \ \boldsymbol{x}_t(n) \sim P_{0,n} \,, \\ t \geqslant t_r &: \ \boldsymbol{x}_t(n) \sim P_{1,n} \,, \end{aligned} \tag{6.1}$$

with

$$\begin{aligned} &\forall n \in \mathcal{C}^*, \ \ P_{0,n} \neq P_{1,n} \,, \\ &\forall n \notin \mathcal{C}^*, \ \ P_{0,n} = P_{1,n} \,. \end{aligned} \tag{6.2}$$

where $P_{0,n}$ and $P_{1,n}$ denote probability measures on $\mathcal{M}$ that represent the distribution of the signal $\boldsymbol{x}_t(n)$ before and after the changepoint t_r. For ease of notation, (6.1) considers only a single changepoint. However, the algorithm presented hereafter can handle multiple changepoints.

6.3 Methodology

Distributed CPD strategies [Sharpnack 2013, Ferrari 2019, Ferrari 2020b] initially designed to handle time series signals in an Euclidean space cannot handle streaming data that lies on a manifold. In this work, we aim to design a new framework to detect changepoints in streaming manifold-valued signals over graphs. First, we consider an online CPD strategy on Riemannian manifolds to take the data geometry into account. Second, we leverage the graph topology by graph-filtering test statistics computed at each node, without compromising the manifold interpretation of the signals. Finally, the centralized graph filter is implemented in a fully distributed way, to provide an efficient CPD method for large-scale networks.

6.3.1 CPD in streaming manifold-valued signals

Some recent algorithms have been investigated for detecting changepoints in streaming manifold-valued data. For instance, an online and parametric CPD algorithm in [Bouchard 2020] was specifically designed for the compound Gaussian distribution. An offline and non-parametric technique can be found in [Dubey 2020]. In this subsection, we consider an online and non-parametric algorithm presented in Chapter 5, which detects changepoints by monitoring for abrupt changes in the *Karcher means* [Karcher 1977] of the streaming data. Consider a random signal $\boldsymbol{x}$ in $\mathcal{M}$ distributed according to P. As already discussed in Chapter 5, its Karcher mean is defined as:

$$\boldsymbol{m}^* = \underset{\boldsymbol{m}\in\mathcal{M}}{\arg\min} \left\{ f(\boldsymbol{m}) \triangleq \int d^2_{\mathcal{M}}(\boldsymbol{m}, \boldsymbol{x})\, dP(\boldsymbol{x}) \right\}. \tag{6.3}$$

Recall that the algorithm in Chapter 5 allows us to detect changepoints by comparing two Karcher means estimated using two stochastic Riemannian optimization methods, one rapidly approaching the information of new data, and another steadily progressing to emphasize a long-term trend. This method can be applied to detect changepoints in the distribution of $\boldsymbol{x}_t(n)$ at each node n by estimating its Karcher mean. Specifically, for each node n, using two Riemannian stochastic gradient descent (SGD) algorithms [Bonnabel 2013] with two distinct stepsizes $0 < \lambda < \Lambda$, two Karcher means are estimated recursively as:

$$\boldsymbol{m}_{\lambda,t}(n) = \exp_{\boldsymbol{m}_{\lambda,t-1}(n)} \big(-\lambda H(\boldsymbol{m}_{\lambda,t-1}(n), \boldsymbol{x}_t(n)) \big), \tag{6.4}$$

$$\boldsymbol{m}_{\Lambda,t}(n) = \exp_{\boldsymbol{m}_{\Lambda,t-1}(n)} \big(-\Lambda H(\boldsymbol{m}_{\Lambda,t-1}(n), \boldsymbol{x}_t(n)) \big), \tag{6.5}$$

where $\exp_{\boldsymbol{m}}$ is the exponential map at $\boldsymbol{m}$, and $H(\boldsymbol{m}, \boldsymbol{x})$ denotes the Riemannian gradient of the loss function $f(\boldsymbol{m})$. The convergence rates of (6.4) and (6.5) are

Algorithm 8 CPD in streaming manifold-valued signals

Input: $\{\boldsymbol{x}_t(n)\}_{n=1}^N$, stepsizes λ, Λ, threshold ξ.
Initialize $\boldsymbol{m}_{\lambda,0}(n) = \boldsymbol{m}_{\Lambda,0}(n) = \boldsymbol{x}_0(n)$
for $t = 1, 2, 3, \ldots$ **do**
  for $n = 1, \ldots, N$ **do**
    Update $\boldsymbol{m}_{\lambda,t}(n)$ and $\boldsymbol{m}_{\Lambda,t}(n)$ using (6.4) and (6.5)
    Compute $d_t(n)$ using (6.6)
  if $\exists\ n \in \mathcal{N} : d_t(n) > \xi$ **then**
    Flag t as a changepoint

directly affected by λ and Λ. Constraint $\lambda < \Lambda$ implies that $\boldsymbol{m}_{\Lambda,t}(n)$ is more adaptive to new data while $\boldsymbol{m}_{\lambda,t}(n)$ has a longer memory. The exponential maps in (6.4) and (6.5) can also be replaced by a computationally simpler retraction $R_{\boldsymbol{m}_{\lambda,t-1}(n)}$ and $\boldsymbol{m}_{\Lambda,t-1}(n)$, respectively.

By assessing the disparity between $\boldsymbol{m}_{\lambda,t}(n)$ and $\boldsymbol{m}_{\Lambda,t}(n)$ through the geodesic distance on $\mathcal{M}$, an adaptive CPD statistic $d_t(n)$ for each node n can be computed as:

$$d_t(n) = d_{\mathcal{M}}\big(\boldsymbol{m}_{\lambda,t}(n), \boldsymbol{m}_{\Lambda,t}(n)\big)\,. \tag{6.6}$$

CPD at each node can then be performed by comparing $d_t(n)$ to a threshold ξ. The corresponding CPD procedure on manifolds is summarized in Algorithm 8.

6.3.2 Community CPD over graphs

Consider $\boldsymbol{d}_t = [d_t(1), \ldots, d_t(N)]^\top$. The CPD statistic computed in (6.6) at each node does not take into account the graph topology. To improve the localization of the communities that might contain a changepoint, we consider the graph filter $\boldsymbol{g}$ introduced in [Sharpnack 2013] for computing the GFSS. The GFSS aims to test if a graph signal with scalar measurements at each node is zero-mean against the hypothesis that there is a community of well-connected nodes where signals have a mean that differs from zero. We propose to apply the GFSS to the node-level test statistics $\boldsymbol{d}_t$ defined in (6.6) rather than original signals $\boldsymbol{x}_t$ to avoid loss of the manifold interpretation of problem (6.1).

Let us denote the normalized graph Laplacian of $\mathcal{G}$ by $\boldsymbol{L}$. Let $\boldsymbol{u}_n$ for $n = 1, \ldots, N$ be the set of orthonormal eigenvectors of $\boldsymbol{L}$ with μ_n being the associated eigenvalues. Given the node-level test statistics $\boldsymbol{d}_t$, the GFSS is defined as:

$$t_{\text{GFSS}}(\boldsymbol{d}_t) = \|\boldsymbol{g}_{\boldsymbol{d}_t}\|_2\,, \tag{6.7}$$

$$\boldsymbol{g}_{\boldsymbol{d}_t} = \sum_{n=2}^{N} h^*(\mu_n)(\boldsymbol{u}_n^\top \boldsymbol{d}_t)\boldsymbol{u}_n\,, \tag{6.8}$$

where $\boldsymbol{g}_{\boldsymbol{d}_t}$ is the graph-filtered statistics, and $h^*(\mu)$ is the frequency response of the filter defined as [Sharpnack 2013]:

$$h^*(\mu) = \min\left\{1, \sqrt{\frac{\gamma}{\mu}}\right\},\ \mu > 0\,, \tag{6.9}$$

Algorithm 9 Distributed CPD in streaming manifold-valued signals over graphs

Input : $\{\mathcal{X}_t\}$, stepsizes λ, Λ, threshold ξ
Initialize $\boldsymbol{y}_{\ell,-1} = \mathbf{0}$
for $t = 1, 2, 3, \ldots$ **do**
  $\forall\, n \in \mathcal{N}$, compute $d_t(n)$ as in Algorithm 8
  Set $\boldsymbol{d}_t = [d_t(1), \ldots, d_t(N)]^\top$
  for $\ell = 1, \ldots, K$ **do**
    $\boldsymbol{y}_{\ell,t} = \psi_\ell \boldsymbol{L} \boldsymbol{y}_{\ell,t-1} + \varphi_\ell \boldsymbol{d}_t$
    $\hat{\boldsymbol{g}}_{\boldsymbol{d}_t} = \sum_{\ell=1}^{K} \boldsymbol{y}_{\ell,t} + c\boldsymbol{d}_t$
  if $\exists\, n \in \mathcal{N} : \hat{\boldsymbol{g}}_{\boldsymbol{d}_t}(n) > \xi$ **then**
    Flag t as a changepoint

where $\gamma > 0$ being a tuning parameter. Adjusting the parameter γ is crucial for the GFSS performance, as γ controls the bandwidth of the low-pass graph filter. The practical adjustment of γ is discussed in Section 6.4.

To get more insight into the filtering procedure in (6.8), let us recall the role of the eigenvectors $\boldsymbol{u}_n$ of the graph Laplacian matrix $\boldsymbol{L}$ in spectral clustering [Ng 2001, Von Luxburg 2007]. Consider the ideal case of a graph with $1 < K < N$ disconnected clusters of densely connected nodes. We denote by $\mathcal{C}_n$ the set of nodes in cluster n, with $n = 1, \cdots, K$. Each $\boldsymbol{u}_n$ is proportional to the indicator function of $\mathcal{C}_n$, and $\boldsymbol{u}_n^\top \boldsymbol{d}_t$ is therefore proportional to the sum of the $d_t(n)$'s in $\mathcal{C}_n$. This means that $(\boldsymbol{u}_n^\top \boldsymbol{d}_t)\boldsymbol{u}_n$ in (6.8) assigns the average value of the $d_t(n)$'s in $\mathcal{C}_n$ to each node in $\mathcal{C}_n$. As the number of communities K is unknown, the filter response in (6.9) is designed to penalize large numbers of clusters in (6.8). This assumption is also a cornerstone of spectral clustering methods [Ng 2001, Von Luxburg 2007].

6.3.3 Distributed implementation

The filtering operation as defined in (6.8) requires the eigen-decomposition of the normalized graph Laplacian matrix $\boldsymbol{L}$. This is computationally expensive and hence cannot be scaled to large networks. A strategy to make our community CPD algorithm scalable is to substitute the filter in (6.8)–(6.9) with a distributed filter that can be implemented locally at each graph vertex [Shuman 2011, Segarra 2015]. In contrast to these finite impulse response filters, an autoregressive moving average (ARMA) graph filter has been proposed in [Loukas 2015, Isufi 2017]. This filter recursively aggregates signals in the neighborhood of each node, which therefore requires low computation and memory costs.

In the context of online community CPD, we propose to apply the parallel ARMA_K graph filter [Isufi 2017], an approximation of the GFSS filter $\boldsymbol{g}$ defined in (6.8), to the streaming statistics $\boldsymbol{d}_t$ in (6.6), which leads to:

$$\boldsymbol{y}_{\ell,t} = \psi_\ell \boldsymbol{L} \boldsymbol{y}_{\ell,t-1} + \varphi_\ell \boldsymbol{d}_t,\ \boldsymbol{y}_{\ell,-1} = \mathbf{0},\ \forall \ell = 1 \ldots K\,, \tag{6.10}$$

$$\hat{\boldsymbol{g}}_{\boldsymbol{d}_t} = \sum_{\ell=1}^{K} \boldsymbol{y}_{\ell,t} + c\boldsymbol{d}_t\,. \tag{6.11}$$

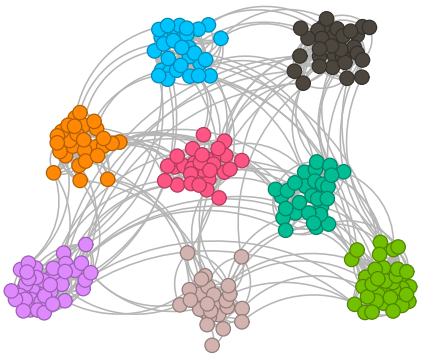

FIGURE 6.1 – Graph topology with colored communities $\mathcal{C}_i$.

The operation $\boldsymbol{L}\boldsymbol{y}_{\ell,t-1}$ is a graph-shift which is performed locally at each node n by linearly combining the statistics as follows: $\sum_{k\in\mathcal{N}_p} L_{p,k}\, y_{\ell,t-1,k}$, where $\mathcal{N}_p$ is the neighborhood of node p, including p itself, $y_{\ell,t-1,k}$ is the k-th entry of $\boldsymbol{y}_{\ell,t-1}$ and $L_{p,k}$ the (p,k)-th entry of $\boldsymbol{L}$. This operation plays a central role in the fully distributed graph-filtering procedure of streaming statistics $\boldsymbol{d}_t$ as it only involves the values of the neighboring nodes over graphs. Note that there exists a series of appropriate parameters c and $\{(\psi_\ell, \varphi_\ell)\}_{\ell=1\ldots K}$ so that $h(\mu)$ closely approximates $h^*(\mu)$ in (6.9). The practical computation problems of parameters c and $(\psi_\ell, \varphi_\ell)$ for $\ell = 1, \ldots, K$ are discussed in Section 6.4. The fully distributed CPD procedure for streaming manifold-valued signals over graphs is described in Algorithm 9. In practice, the adaptive threshold selection strategy described in Section 5.2.4 can be used to determine ξ in algorithms 8 and 9.

6.4 Simulations

We shall now illustrate the performance of the proposed approach using graph signals $\boldsymbol{x}_t(n)$ over the manifold of SPD matrices $\mathcal{S}_p^{++}$. The topology of the graph $\mathcal{G}$ [1] used for simulations is illustrated in FIGURE. 6.1. It contains $p = 250$ nodes and $m = 2508$ edges, and 8 communities. These communities $\mathcal{C}_i$ have been unfolded using [Blondel 2008] and colored for visualization. We generated $\mathcal{X}_t$ as in (6.1) with a changepoint in community $\mathcal{C}^*$. Its nodes are colored in orange in FIGURE. 6.1. The synthetic matrices $\boldsymbol{\Sigma}_t \in \mathcal{S}_p^{++}$ with $d = 6$ were sampled from a Wishart distribution with the scaling matrix $\boldsymbol{V}$ and degrees of freedom d. We generated 800 independent samples and inserted the changepoint at $t_r = 500$ in (6.1) where we randomly reset $\boldsymbol{V}$ from one random matrix to another..

With $\{\boldsymbol{\Sigma}_t\}_{t\in\mathbb{N}}$ lying on $\mathcal{S}_p^{++}$ and the metric defined in (5.52), the Karcher means were estimated by minimizing the objective function (5.55) using the Riemannian SGD algorithms in (6.4) and (6.5) with the stochastic gradient (5.53) and the retraction (5.54). The empirical evaluation was used to find the optimal value of

1. The topology of the graph can be downloaded from https://github.com/andferrari/icassp20.

stepsizes as $\lambda = 0.01$ and $\Lambda = 0.02$ to compute the online statistic in (6.6). Given a filter order K, as discussed in [Ferrari 2019], the filter parameters can be computed by minimizing

$$J(\boldsymbol{a}, \boldsymbol{b}) = \sum_i [B(x_i) - h(x_i)A(x_i)]^2 \tag{6.12}$$

w.r.t. $(\boldsymbol{a}, \boldsymbol{b})$ where $A(x) = 1+\sum_{\ell=1}^{K} \boldsymbol{a}(\ell)x^\ell = \prod_{\ell=1}^{K}(1-\psi_\ell x)$, and $B(x) = \sum_{\ell=0}^{K} \boldsymbol{b}(\ell)x^\ell$, over a uniform grid x_i on the interval $(0, 2)$. According to [Ferrari 2019], this quadratic problem must be solved by the following linear constraints w.r.t. $\boldsymbol{a} : |A(x_i)| < \beta$ for all x_i on the grid, with β a parameter to be set by the user. Finally, initial variables c and $\{(\phi_\ell, \psi_\ell)\}$ were estimated from $(\boldsymbol{a}, \boldsymbol{b})$ by a partial fraction expansion of $B(x)/A(x)$. The empirical evaluation was used to set $\gamma = 0.03$, $K = 4$, and $\beta = 0.1$. Subsequently, the resulting filter $h^*(\mu)$ was checked to be stable.

To compare the detection performance of these algorithms, Monte Carlo simulations were performed to estimate the mean detection delay, average run length, and false alarm rate for daGFSS, $\boldsymbol{d}_t$ and $\hat{\boldsymbol{g}}_{\boldsymbol{d}_t}$. Considering $\hat{\boldsymbol{g}}_{\boldsymbol{d}_t}$ for illustration purposes, these metrics are defined as follows:

$$T_{\text{mdd}} = \inf\{t - t_r : \hat{\boldsymbol{g}}_{\boldsymbol{d}_t}(n) > \xi \,|\, n \in \mathcal{C}^*\}, \tag{6.13}$$

$$T_{\text{arl}} = \inf\{t : \hat{\boldsymbol{g}}_{\boldsymbol{d}_t}(n) > \xi \,|\, n \notin \mathcal{C}^*\}, \tag{6.14}$$

$$P_{\text{fa}} = P\big(\hat{\boldsymbol{g}}_{\boldsymbol{d}_t}(n) > \xi \,|\, t > t_r, n \notin \mathcal{C}^*\big). \tag{6.15}$$

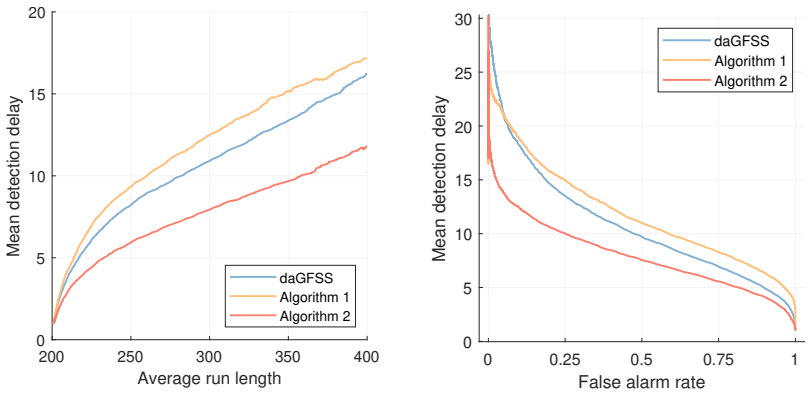

FIGURE 6.2 – Average run lengths versus mean detection delays for all compared algorithms (left). Mean detection delays versus false alarm rates for all compared algorithms (Right). Algorithm 1: $\boldsymbol{d}_t$, Algorithm 2: $\hat{\boldsymbol{g}}_{\boldsymbol{d}_t}$.

To illustrate the advantage of exploiting both manifold geometry and graph topology, we compared our Algorithm 9 to two baselines. The first baseline is daGFSS [Ferrari 2019], originally designed for Euclidean data. We applied daGFSS to the vectorization of the lower triangular and diagonal parts of $\boldsymbol{\Sigma}_t$. The second baseline is the Karcher means-based CPD method on manifolds detailed in Algorithm 8, performed node-by-node without cooperation.

FIGURE. 6.2 (left) and FIGURE. 6.2 (right) show the mean detection delays versus average run lengths and false alarm rates, respectively, of all detectors consi-

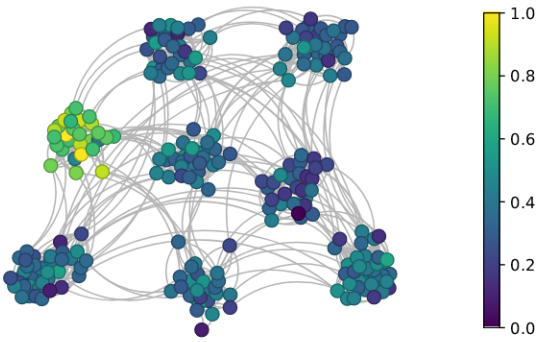

FIGURE 6.3 – Normalized $\hat{\boldsymbol{g}}_{\boldsymbol{d}_t}$ at $t_r + 20$.

dered in this chapter. As expected, Algorithm 9 clearly benefits from $\boldsymbol{d}_t$ compared to daGFSS, and $\hat{\boldsymbol{g}}$ in Algorithm 8. FIGURE. 6.3 illustrates the test statistics $\hat{\boldsymbol{g}}_{\boldsymbol{d}_t}$, normalized here only for illustration convenience. This figure shows the ability of the proposed algorithm to localize the community where changepoints occur.

6.5 Conclusion

This chapter introduced a distributed algorithm for detecting changepoints in streaming manifold-valued signals within an unknown community of a graph. The algorithm is online, non-parametric, and fully distributed across the graph nodes. Simulation results validate its effectiveness in leveraging data manifold geometry and graph topology for improved performance.

Chapter 7

Conclusions and perspectives

Contents

7.1 Conclusions

This thesis investigated new approaches integrating both physical modeling and machine learning strategies. The composition of these two methodologies has attracted considerable attention during the past few years, since they provide clear interpretation with specific domain knowledge and meanwhile achieve superior performance with powerful learning ability. The work of this thesis explored problems related to this composed methodology under two important signal processing contexts, solving inverse problems in hyperspectral imaging and detecting changepoints in time series.

The first part of the thesis concerned the joint modeling and learning approaches for hyperspectral imaging and explored a novel idea to design Plug-and-Play methods for two distinct inverse problems in hyperspectral imaging. In Chapter 2, we started by introducing a tuning-free hypersepctral image deconvolution method utilizing the Plug-and-Play framework. Instead of relying on manually crafted priors, we developed a blind B3DDN denoiser leveraging deep learning to capture spectral-spatial information from the data directly by substituting steps in an ADMM-based optimizer. The hyperparameters are automatically learned using the B3DDN and a measure of 3D residual whiteness. Chapter 3 investigated an unsupervised deep learning method for hypersepctral and multispectral image deconvolution with inter-image variability. Initially, we devised a novel imaging model incorporating both joint and image-specific priors of the two latent high-resolution images. Inter-image variability was characterized using a hyper-Laplacian distribution, while the image-specific priors for the latent high-resolution images were implicitly learned through deep denoising engines. To address the non-convex cost function, we explored an iteratively reweighted scheme. A lightweight, image-specific CNN-based denoiser

was designed with a zero-shot training strategy. During the optimization process, the network parameters were iteratively updated to adapt to the variability of the estimated high-resolution images as convergence was achieved. The proposed methods in the first part of this thesis achieved superior experimental performance in both the image deconvolution and fusion inverse problems in hyperspectral imaging when compared to state-of-the-art approaches.

The second part of the thesis addressed the problem of detecting changepoints in time series and investigated the use of machine learning techniques to learn objectives supported by physics-based models. In Chapter 4, we proposed a novel strategy for online changepoint detection that leverages the powerful learning ability of neural networks to estimate density-ratio in a non-parametric manner. A continual learning framework was exploited to devise an adaptive detection algorithm that retains past information. Chapter 5 presented a general approach for non-parametric online changepoint detection on Riemannian manifolds. We computed an adaptive test statistic by employing stochastic Riemannian optimization to track the generalized Karcher mean of data streams. By conducting a theoretical analysis of the non-asymptotic convergence of the stochastic Riemannian gradient descent algorithm, we established performance guarantees for both detection and false alarm rates and then applied this method to two typical instances of manifolds. To extend this method to detect changepoints in streaming manifold-valued signals within an unknown community of a graph, Chapter 6 incorporated a local test statistic at each node to handle the inherent geometry of data lying on a manifold, along with a fully distributed graph filter that incorporates network topology information. Experimental and simulation results validated the effectiveness of the methods proposed in the second part of this thesis to detect changepoints in streaming Euclidean data, data lying on Riemannian manifolds, and manifold-valued data over graphs.

7.2 Perspectives

This thesis proposed novel frameworks integrating both physics-based and machine learning approaches to problems in hyperspectral image and time series analysis. There are several related problems which remain to be further investigated. We list some of these future research directions below.

7.2.1 CPD in multi-temporal hyperspectral data

The increasing availability of multi-temporal hyperspectral devices allows for a detailed analysis of the evolution of a scene over time [Borsoi 2021c]. In this thesis, the problems of hyperspectral imaging and changepoint detection were studied in two individual parts, one research direction with great potential is to investigate online changepoint detection in multi-temporal hyperspectral data. This can potentially benefit various applications, ranging from agricultural and forestry monitoring to natural disaster and urban landscape analysis [Borsoi 2021f].

In addition, it is also promising to develop an online changepoint detection algorithm to process the hyperspectral data cubes acquired slice by slice in push-broom imaging systems [Song 2019]. This can be compatible with real-time processing in industrial applications of hyperspectral imaging.

7.2.2 CPD over graphs with neural networks

We considered a new online changepoint detection strategy based on neural density-ratio estimation to process data in sliding windows in Chapter 4. An open problem is how to extend this algorithm to detect changepoints in streaming data over graphs. One may compute test statistics at each node, followed by leveraging graph filtering techniques to aggregate node-level statistics, as discussed in Chapter 6. However, this needs to implement neural networks at each node and design the graph filter beforehand. A more efficient alternative is to design graph neural networks to take graph topology into account and process the data in an end-to-end manner.

7.2.3 Distributed optimization on Riemannian manifolds

In Chapter 6, a distributed method was presented considering a graph filter to process statistics proposed in Chapter 5 for streaming data on general manifolds, but it was only designed for the changepoint detection task.

In a more general setting, distributed optimization recently gained considerable attention. It aims to solve the multi-agent optimization problem considering *consensus* on a Riemannian manifold $\mathcal{M}$:

$$\min_{\boldsymbol{w}\in\mathcal{M}} \sum_{k=1}^{K} J_k(\boldsymbol{w}) \tag{7.1}$$

with $J_k : \mathcal{M} \to \mathbb{R}$ a local risk function defined for each agent as $J_k(\boldsymbol{w}) = \mathbb{E}_{\boldsymbol{x}_k}\{Q(\boldsymbol{w};\boldsymbol{x}_k)\}$ in terms of some loss function $Q(\boldsymbol{w};\boldsymbol{x}_k)$. The expectation is computed over the unknown distribution of the data $\boldsymbol{x}_k$, which makes it necessary to use a stochastic approximation based on the set of independent realizations $\boldsymbol{x}_{k,t}$, observed sequentially over time. A wide range of applications in machine learning and signal processing can be written in the form of (7.1), including dictionary learning, principal component analysis, and low-rank matrix completion [Boumal 2023a].

To solve (7.1), an effective way is to develop distributed optimization on manifolds, which directly operates on $\mathcal{M}$ by exploiting the inherent geometry. In the exploratory work [Wang 2024b], we introduced two general Riemannian diffusion adaptation strategies and considered an application for online distributed principal component analysis. The theoretical behavior of this algorithm as well as its application to different problems is an interesting direction to be further investigated.

Bibliography

[Absil 2009] P-A Absil, Robert Mahony and Rodolphe Sepulchre. Optimization algorithms on matrix manifolds. Princeton University Press, 2009. (Cited on pages 78, 80 and 95.)

[Afsari 2011] Bijan Afsari. *Riemannian ℓ^p center of mass : existence, uniqueness, and convexity.* Proceedings of the American Mathematical Society, vol. 139, no. 2, pages 655–673, 2011. (Cited on page 82.)

[Aiazzi 2006] B Aiazzi, L Alparone, S Baronti, A Garzelli and M Selva. *MTF-tailored multiscale fusion of high-resolution MS and Pan imagery.* Photogrammetric Engineering & Remote Sensing, vol. 72, no. 5, pages 591–596, 2006. (Cited on pages 37 and 52.)

[Akhtar 2015] Naveed Akhtar, Faisal Shafait and Ajmal Mian. *Bayesian sparse representation for hyperspectral image super resolution.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3631–3640, 2015. (Cited on page 38.)

[Almeida 2013] Mariana SC Almeida and Mário AT Figueiredo. *Parameter estimation for blind and non-blind deblurring using residual whiteness measures.* IEEE Trans. Image Process, vol. 22, no. 7, pages 2751–2763, 2013. (Cited on page 16.)

[Aminikhanghahi 2017] Samaneh Aminikhanghahi and Diane J Cook. *A survey of methods for time series change point detection.* Knowledge and information systems, vol. 51, no. 2, pages 339–367, 2017. (Cited on pages 2 and 4.)

[Ammanouil 2014] Rita Ammanouil, André Ferrari, Cédric Richard and David Mary. *Blind and fully constrained unmixing of hyperspectral images.* IEEE Transactions on Image Processing, vol. 23, no. 12, pages 5510–5518, 2014. (Cited on page 45.)

[Arlot 2019] Sylvain Arlot, Alain Celisse and Zaid Harchaoui. *A kernel multiple change-point algorithm via model selection.* Journal of machine learning research, vol. 20, no. 162, pages 1–56, 2019. (Cited on page 79.)

[Atashgahi 2022] Zahra Atashgahi, Decebal Constantin Mocanu, Raymond Veldhuis and Mykola Pechenizkiy. *Memory-free Online Change-point Detection : A Novel Neural Network Approach.* arXiv preprint arXiv :2207.03932, 2022. (Cited on page 68.)

[Bai 1998] Jushan Bai and Pierre Perron. *Estimating and Testing Linear Models with Multiple Structural Changes.* Econometrica, vol. 66, no. 1, pages 47–78, 1998. (Cited on page 4.)

[Bhatia 2009] Rajendra Bhatia. Positive definite matrices. Princeton university press, 2009. (Cited on page 94.)

[Bioucas-Dias 2013] José M Bioucas-Dias, Antonio Plaza, Gustavo Camps-Valls, Paul Scheunders, Nasser Nasrabadi and Jocelyn Chanussot. *Hyperspectral remote sensing data analysis and future challenges.* IEEE Geosci. Remote Sens. Mag., vol. 1, no. 2, pages 6–36, 2013. (Cited on page 2.)

[Blondel 2008] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte and Etienne Lefebvre. *Fast unfolding of communities in large networks.* Journal of Statistical Mechanics : Theory and Experiment, vol. 2008, no. 10, page P10008, 2008. (Cited on page 110.)

[Bongard 2011] S Bongard, F Soulez, Éric Thiébaut and É Pecontal. *3D deconvolution of hyper-spectral astronomical data.* Mon. Not. Roy. Astron. Soc., vol. 418, no. 1, pages 258–270, 2011. (Cited on page 15.)

[Bonnabel 2013] Silvere Bonnabel. *Stochastic gradient descent on Riemannian manifolds.* IEEE Transactions on Automatic Control, vol. 58, no. 9, pages 2217–2229, 2013. (Cited on pages 78, 80, 82, 94 and 107.)

[Borsoi 2018] Ricardo Augusto Borsoi, Guilherme Holsbach Costa and José Carlos Moreira Bermudez. *A new adaptive video super-resolution algorithm with improved robustness to innovations.* IEEE Transactions on Image Processing, vol. 28, no. 2, pages 673–686, 2018. (Cited on page 45.)

[Borsoi 2020] R. A. Borsoi, T. Imbiriba and J. C. M. Bermudez. *Super-Resolution for Hyperspectral and Multispectral Image Fusion Accounting for Seasonal Spectral Variability.* IEEE Transactions on Image Processing, vol. 29, no. 1, pages 116–127, 2020. (Cited on pages 38, 39, 42 and 52.)

[Borsoi 2021a] Ricardo A Borsoi, Clémence Prévost, Konstantin Usevich, David Brie, José CM Bermudez and Cédric Richard. *Coupled Tensor Models Accounting for Inter-image Variability.* In 2021 55th Asilomar Conference on Signals, Systems, and Computers, pages 1586–1590. IEEE, 2021. (Cited on pages 39 and 42.)

[Borsoi 2021b] Ricardo Augusto Borsoi. *Spectral variability in hyperspectral unmixing : Multiscale, tensor, and neural network-based approaches.* PhD thesis, Université Côte d'Azur ; Universidade federal de Santa Catarina (Brésil), 2021. (Cited on page 2.)

[Borsoi 2021c] Ricardo Augusto Borsoi, Tales Imbiriba, José Carlos Moreira Bermudez and Cédric Richard. *Fast unmixing and change detection in multitemporal hyperspectral data.* IEEE Transactions on Computational Imaging, vol. 7, pages 975–988, 2021. (Cited on pages 2 and 114.)

[Borsoi 2021d] Ricardo Augusto Borsoi, Tales Imbiriba, José Carlos Moreira Bermudez, Cédric Richard, Jocelyn Chanussot, Lucas Drumetz, Jean-Yves Tourneret, Alina Zare and Christian Jutten. *Spectral Variability in Hyperspectral Data Unmixing : A Comprehensive Review.* IEEE Geoscience and Remote Sensing Magazine, vol. 9, no. 4, pages 223–270, 2021. (Cited on pages 38, 42 and 44.)

[Borsoi 2021e] Ricardo Augusto Borsoi, Clémence Prévost, Konstantin Usevich, David Brie, José Carlos Moreira Bermudez and Cédric Richard. *Coupled*

Tensor Decomposition for Hyperspectral and Multispectral Image Fusion with Inter-image Variability. IEEE Journal of Selected Topics in Signal Processing, vol. 15, no. 3, pages 702–717, 2021. (Cited on pages 38, 39, 42, 43, 52 and 61.)

[Borsoi 2021f] Ricardo Augusto Borsoi, Cédric Richard, André Ferrari, Jie Chen and José Carlos M Bermudez. *Online graph-based change point detection in multiband image sequences.* In 2020 28th European Signal Processing Conference (EUSIPCO), pages 850–854. IEEE, 2021. (Cited on pages 2, 4, 105 and 114.)

[Bottou 2010] Léon Bottou. *Large-scale machine learning with stochastic gradient descent.* In Proceedings of COMPSTAT'2010 : 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers, pages 177–186. Springer, 2010. (Cited on page 1.)

[Bottou 2018] Léon Bottou, Frank E Curtis and Jorge Nocedal. *Optimization methods for large-scale machine learning.* Siam Review, vol. 60, no. 2, pages 223–311, 2018. (Cited on pages 1 and 6.)

[Bouchard 2020] Florent Bouchard, Ammar Mian, Jialun Zhou, Salem Said, Guillaume Ginolhac and Yannick Berthoumieu. *Riemannian geometry for compound Gaussian distributions : Application to recursive change detection.* Signal Processing, vol. 176, page 107716, 2020. (Cited on pages 78, 79 and 107.)

[Boumal 2023a] Nicolas Boumal. An introduction to optimization on smooth manifolds. Cambridge University Press, 2023. (Cited on pages 78, 80, 86, 88, 91, 94, 95 and 115.)

[Boumal 2023b] Nicolas Boumal. An introduction to optimization on smooth manifolds. Cambridge University Press, 2023. (Cited on page 106.)

[Boyd 2004] Stephen Boyd, Stephen P Boyd and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2004. (Cited on pages 13, 19 and 20.)

[Boyd 2011] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein*et al. Distributed optimization and statistical learning via the alternating direction method of multipliers.* Found. Trends Mach. Learn., vol. 3, no. 1, pages 1–122, 2011. (Cited on page 13.)

[Brezini 2021] Salah Eddine Brezini, Moussa Sofiane Karoui, Fatima Zohra Benhalouche, Yannick Deville and Abdelaziz Ouamri. *Hypersharpening by an NMF-Unmixing-Based Method Addressing Spectral Variability.* IEEE Geoscience and Remote Sensing Letters, vol. 19, pages 1–5, 2021. (Cited on pages 39, 42 and 43.)

[Brifman 2016] Alon Brifman, Yaniv Romano and Michael Elad. *Turning a denoiser into a super-resolver using plug and play priors.* In Proc. IEEE Int. Conf. Image Process. (ICIP), pages 1404–1408, 2016. (Cited on pages 16 and 17.)

[Camacho 2022] Ariolfo Camacho, Edwin Vargas and Henry Arguello. *Hyperspectral and multispectral image fusion addressing spectral variability by an augmented*

linear mixing model. International Journal of Remote Sensing, vol. 43, no. 5, pages 1577–1608, 2022. (Cited on pages 39, 42 and 43.)

[Chang 2020] Yi Chang, Luxin Yan, Xi-Le Zhao, Houzhang Fang, Zhijun Zhang and Sheng Zhong. *Weighted low-rank tensor recovery for hyperspectral image restoration*. IEEE Trans. Cybern., vol. 50, no. 11, pages 4558–4572, 2020. (Cited on pages 16 and 29.)

[Chen 2019] Hao Chen. *Sequential change-point detection based on nearest neighbors*. The Annals of Statistics, vol. 47, no. 3, pages 1381–1407, 2019. (Cited on pages 68 and 72.)

[Chen 2020] Meiya Chen, Yi Chang, Shuning Cao and Luxin Yan. *Learning Blind Denoising Network for Noisy Image Deblurring*. In Proc. IEEE Int. Conf. on Acoust, Speech, Signal Process (ICASSP), pages 2533–2537. IEEE, 2020. (Cited on pages 16, 17 and 23.)

[Chen 2022] Jie Chen, Min Zhao, Xiuheng Wang, Cédric Richard and Susanto Rahardja. *Integration of physics-based and data-driven models for hyperspectral image unmixing*. IEEE Signal Process. Mag., 2022. (Cited on pages 9, 16, 38 and 49.)

[Chen 2023] Likai Chen, Georg Keilbar and Wei Biao Wu. *Recursive Quantile Estimation : Non-Asymptotic Confidence Bounds*. Journal of Machine Learning Research, vol. 24, no. 91, pages 1–25, 2023. (Cited on page 93.)

[Cheng 2021] Xiuyuan Cheng and Yao Xie. *Neural tangent kernel maximum mean discrepancy*. Advances in Neural Information Processing Systems, vol. 34, pages 6658–6670, 2021. (Cited on page 79.)

[Cohen 1995] Leon Cohen. Time-frequency analysis, volume 778. Prentice hall New Jersey, 1995. (Cited on page 99.)

[Collas 2022] Antoine Collas. *Riemannian geometry for statistical estimation and learning : application to remote sensing*. PhD thesis, université Paris-Saclay, 2022. (Cited on page 94.)

[Costa 2006] AFB Costa and MA Rahim. *A single EWMA chart for monitoring process mean and process variance*. Quality Technology & Quantitative Management, vol. 3, no. 3, pages 295–305, 2006. (Cited on pages 79 and 81.)

[Cranmer 2015] Kyle Cranmer, Juan Pavez and Gilles Louppe. *Approximating likelihood ratios with calibrated discriminative classifiers*. arXiv preprint arXiv :1506.02169, 2015. (Cited on pages 69 and 70.)

[Daubechies 2010] Ingrid Daubechies, Ronald DeVore, Massimo Fornasier and C Sinan Güntürk. *Iteratively reweighted least squares minimization for sparse recovery*. Communications on Pure and Applied Mathematics : A Journal Issued by the Courant Institute of Mathematical Sciences, vol. 63, no. 1, pages 1–38, 2010. (Cited on page 46.)

[De Lange 2021] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh and Tinne Tuytelaars. *A continual*

learning survey : Defying forgetting in classification tasks. IEEE transactions on pattern analysis and machine intelligence, vol. 44, no. 7, pages 3366–3385, 2021. (Cited on page 68.)

[De Ryck 2021] Tim De Ryck, Maarten De Vos and Alexander Bertrand. *Change point detection in time series data using autoencoders with a time-invariant representation.* IEEE Transactions on Signal Processing, vol. 69, pages 3513–3524, 2021. (Cited on page 68.)

[Dean 2010] David Dean, Sridha Sridharan, Robert Vogt and Michael Mason. *The QUT-NOISE-TIMIT corpus for evaluation of voice activity detection algorithms.* In Proceedings of the 11th Annual Conference of the International Speech Communication Association, pages 3110–3113. International Speech Communication Association, 2010. (Cited on page 99.)

[Desobry 2005] Frédéric Desobry, Manuel Davy and Christian Doncarli. *An online kernel change detection algorithm.* IEEE Transactions on Signal Processing, vol. 53, no. 8, pages 2961–2974, 2005. (Cited on pages 68 and 70.)

[Dian 2020] Renwei Dian, Shutao Li and Xudong Kang. *Regularizing hyperspectral and multispectral image fusion by CNN denoiser.* IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 3, pages 1124–1135, 2020. (Cited on pages 38, 44, 49 and 50.)

[Ding 2020] Meng Ding, Xiao Fu, Ting-Zhu Huang, Jun Wang and Xi-Le Zhao. *Hyperspectral super-resolution via interpretable block-term tensor modeling.* IEEE Journal of Selected Topics in Signal Processing, vol. 15, no. 3, pages 641–656, 2020. (Cited on page 38.)

[Dobigeon 2013] Nicolas Dobigeon, Jean-Yves Tourneret, Cédric Richard, José Carlos M Bermudez, Stephen McLaughlin and Alfred O Hero. *Nonlinear unmixing of hyperspectral images : Models and algorithms.* IEEE Signal processing magazine, vol. 31, no. 1, pages 82–94, 2013. (Cited on page 38.)

[Donoho 1994] David L Donoho and Jain M Johnstone. *Ideal spatial adaptation by wavelet shrinkage.* biometrika, vol. 81, no. 3, pages 425–455, 1994. (Cited on page 51.)

[Duan 2019] Xiaomin Duan, Huafei Sun and Xinyu Zhao. *A Matrix Information-Geometric Method for Change-Point Detection of Rigid Body Motion.* Entropy, vol. 21, no. 5, page 531, 2019. (Cited on page 79.)

[Dubey 2020] Paromita Dubey and Hans-Georg Müller. *Fréchet change-point detection.* The Annals of Statistics, vol. 48, no. 6, pages 3312–3335, 2020. (Cited on pages 78, 79, 83, 96, 101 and 107.)

[Durkan 2020] Conor Durkan, Iain Murray and George Papamakarios. *On contrastive learning for likelihood-free inference.* In International Conference on Machine Learning, pages 2771–2781. PMLR, 2020. (Cited on pages 69 and 70.)

[Edelman 1998] Alan Edelman, Tomás A Arias and Steven T Smith. *The geometry of algorithms with orthogonality constraints.* SIAM journal on Matrix Analysis and Applications, vol. 20, no. 2, pages 303–353, 1998. (Cited on page 95.)

[Ferrari 2019] André Ferrari, Cédric Richard and Louis Verduci. *Distributed change detection in streaming graph signals.* In 2019 IEEE 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), pages 166–170. IEEE, 2019. (Cited on pages 105, 107 and 111.)

[Ferrari 2020a] André Ferrari and Cédric Richard. *Non-parametric community change-points detection in streaming graph signals.* In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5545–5549. IEEE, 2020. (Cited on page 79.)

[Ferrari 2020b] André Ferrari and Cédric Richard. *Non-parametric community change-points detection in streaming graph signals.* In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5545–5549. IEEE, 2020. (Cited on pages 105, 106 and 107.)

[Ferrari 2022] André Ferrari, Cédric Richard, Anthony Bourrier and Ikram Bouchikhi. *Online change-point detection with kernels.* Pattern Recognition, page 109022, 2022. (Cited on pages 68, 69, 72, 73 and 79.)

[Fréchet 1948] Maurice Fréchet. *Les éléments aléatoires de nature quelconque dans un espace distancié.* In Annales de l'institut Henri Poincaré, volume 10, pages 215–310, 1948. (Cited on page 81.)

[Fu 2021] Xiyou Fu, Sen Jia, Meng Xu, Jun Zhou and Qingquan Li. *Fusion of hyperspectral and multispectral images accounting for localized inter-image changes.* IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pages 1–18, 2021. (Cited on pages 39, 42, 43, 52 and 61.)

[Gajic 2015] Dragoljub Gajic, Zeljko Djurovic, Jovan Gligorijevic, Stefano Di Gennaro and Ivana Savic-Gajic. *Detection of epileptiform activity in EEG signals based on time-frequency and non-linear analysis.* Frontiers in computational neuroscience, vol. 9, page 38, 2015. (Cited on page 4.)

[Galatsanos 1989] Nikolas P Galatsanos and Roland T Chin. *Digital restoration of multichannel images.* IEEE Trans. Audio, Speech, Language Process., vol. 37, no. 3, pages 415–421, 1989. (Cited on page 15.)

[Galatsanos 1991] Nikolas P Galatsanos, Aggelos K Katsaggelos, Roland T Chin and Allen D Hillery. *Least squares restoration of multichannel images.* IEEE Trans. Signal Process., vol. 39, no. 10, pages 2222–2236, 1991. (Cited on page 15.)

[Garofolo 1993] John S Garofolo. *Timit acoustic phonetic continuous speech corpus.* Linguistic Data Consortium, 1993, 1993. (Cited on page 99.)

[Gaucel 2006] J-M Gaucel, Mireille Guillaume and Salah Bourennane. *Adaptive-3D-Wiener for hyperspectral image restoration : Influence on detection strategy.* In Proc. EUSIPCO, pages 1–5, 2006. (Cited on page 15.)

[Geman 1995] Donald Geman and Chengda Yang. *Nonlinear image recovery with half-quadratic regularization.* IEEE Trans. Image Process., vol. 4, no. 7, pages 932–946, 1995. (Cited on pages 13 and 47.)

[Glasner 2009] Daniel Glasner, Shai Bagon and Michal Irani. *Super-resolution from a single image.* In 2009 IEEE 12th International Conference on Computer Vision, pages 349–356. IEEE, 2009. (Cited on page 50.)

[Golub 1979] Gene H Golub, Michael Heath and Grace Wahba. *Generalized cross-validation as a method for choosing a good ridge parameter.* Technometrics, vol. 21, no. 2, pages 215–223, 1979. (Cited on page 16.)

[Gong 2012] Dian Gong, Gérard Medioni, Sikai Zhu and Xuemei Zhao. *Kernelized temporal cut for online temporal segmentation and recognition.* In Computer Vision–ECCV 2012 : 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part III 12, pages 229–243. Springer, 2012. (Cited on page 79.)

[Gretton 2006] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf and Alex Smola. *A kernel method for the two-sample-problem.* Advances in neural information processing systems, vol. 19, 2006. (Cited on pages 79 and 81.)

[Guo 2021] Lin Guo, Xi-Le Zhao, Xian-Ming Gu, Yong-Liang Zhao, Yu-Bang Zheng and Ting-Zhu Huang. *Three-dimensional fractional total variation regularized tensor optimized model for image deblurring.* Appl. Math. Comput., vol. 404, page 126224, 2021. (Cited on page 29.)

[Gupta 1999] Arjun K Gupta and Daya K Nagar. Matrix variate distributions, volume 104. CRC Press, 1999. (Cited on page 97.)

[Gupta 2022] Muktesh Gupta, Rajesh Wadhvani and Akhtar Rasool. *Real-time Change-Point Detection : A deep neural network-based adaptive approach for detecting changes in multivariate time series data.* Expert Systems with Applications, vol. 209, page 118260, 2022. (Cited on page 68.)

[Gustafsson 1996] Fredrik Gustafsson. *The marginalized likelihood ratio test for detecting abrupt changes.* IEEE Transactions on automatic control, vol. 41, no. 1, pages 66–78, 1996. (Cited on pages 67 and 79.)

[Hansen 1992] Per Christian Hansen. *Analysis of discrete ill-posed problems by means of the L-curve.* SIAM Rev., vol. 34, no. 4, pages 561–580, 1992. (Cited on page 16.)

[Harchaoui 2008] Zaid Harchaoui, Eric Moulines and Francis Bach. *Kernel change-point analysis.* Advances in neural information processing systems, vol. 21, 2008. (Cited on page 79.)

[He 2015] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. *Delving deep into rectifiers : Surpassing human-level performance on imagenet classification.* In Proc. IEEE Int. Conf. Comput. Vis. (ICCV), pages 1026–1034, 2015. (Cited on pages 28 and 53.)

[Henrot 2012] Simon Henrot, Charles Soussen and David Brie. *Fast positive deconvolution of hyperspectral images.* IEEE Trans. Image Process., vol. 22, no. 2, pages 828–833, 2012. (Cited on pages 16, 19 and 29.)

[Howard 2017] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto and Hartwig Adam. *Mobilenets : Efficient convolutional neural networks for mobile vision applications*. arXiv preprint arXiv :1704.04861, 2017. (Cited on page 49.)

[Huang 2015] Wen Huang, Kyle A Gallivan and P-A Absil. *A Broyden class of quasi-Newton methods for Riemannian optimization*. SIAM Journal on Optimization, vol. 25, no. 3, pages 1660–1685, 2015. (Cited on page 80.)

[Huang 2017] Zhiwu Huang and Luc Van Gool. *A Riemannian network for SPD matrix learning*. In Thirty-First AAAI Conference on Artificial Intelligence, 2017. (Cited on page 100.)

[Hunt 1984] B Hunt and Olaf Kubler. *Karhunen-Loeve multispectral image restoration, part I : Theory*. IEEE Trans. Audio, Speech, Language Process., vol. 32, no. 3, pages 592–600, 1984. (Cited on page 15.)

[Hussein 2013] Mohamed E Hussein, Marwan Torki, Mohammad A Gowayyed and Motaz El-Saban. *Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations*. In Twenty-third international joint conference on artificial intelligence, 2013. (Cited on page 100.)

[Ienco 2014] Dino Ienco, Albert Bifet, Bernhard Pfahringer and Pascal Poncelet. *Change detection in categorical evolving data streams*. In 29th annual ACM symposium on applied computing, pages 792–797, 2014. (Cited on page 79.)

[Imamura 2019] Ryuji Imamura, Tatsuki Itasaka and Masahiro Okuda. *Zero-shot hyperspectral image denoising with separable image prior*. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, pages 0–0, 2019. (Cited on page 50.)

[Imbiriba 2018] Tales Imbiriba, Ricardo Augusto Borsoi and José Carlos Moreira Bermudez. *Generalized linear mixing model accounting for endmember variability*. In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1862–1866, Calgary, Canada, 2018. (Cited on page 42.)

[Inclan 1994] Carla Inclan and George C Tiao. *Use of cumulative sums of squares for retrospective detection of changes of variance*. Journal of the American Statistical Association, vol. 89, no. 427, pages 913–923, 1994. (Cited on page 67.)

[Isufi 2017] Elvin Isufi, Andreas Loukas, Andrea Simonetto and Geert Leus. *Autoregressive Moving Average Graph Filtering*. IEEE Transactions on Signal Processing, vol. 65, no. 2, pages 274–288, 2017. (Cited on pages 105 and 109.)

[Kacem 2018] Anis Kacem, Mohamed Daoudi, Boulbaba Ben Amor, Stefano Berretti and Juan Carlos Alvarez-Paiva. *A novel geometric framework on gram matrix trajectories for human behavior understanding*. IEEE transactions on pattern analysis and machine intelligence, vol. 42, no. 1, pages 1–14, 2018. (Cited on page 78.)

[Kadambi 2023] Achuta Kadambi, Celso de Melo, Cho-Jui Hsieh, Mani Srivastava and Stefano Soatto. *Incorporating physics into data-driven computer vision.* Nature Machine Intelligence, pages 1–9, 2023. (Cited on page 1.)

[Kanatsoulis 2018] Charilaos I Kanatsoulis, Xiao Fu, Nicholas D Sidiropoulos and Wing-Kin Ma. *Hyperspectral super-resolution : A coupled tensor factorization approach.* IEEE Transactions on Signal Processing, vol. 66, no. 24, pages 6503–6517, 2018. (Cited on pages 38, 44 and 52.)

[Karcher 1977] Hermann Karcher. *Riemannian center of mass and mollifier smoothing.* Communications on pure and applied mathematics, vol. 30, no. 5, pages 509–541, 1977. (Cited on page 107.)

[Kawahara 2007] Yoshinobu Kawahara, Takehisa Yairi and Kazuo Machida. *Change-point detection in time-series data based on subspace identification.* In Seventh IEEE International Conference on Data Mining (ICDM 2007), pages 559–564. IEEE, 2007. (Cited on page 67.)

[Kawakami 2011] Rei Kawakami, Yasuyuki Matsushita, John Wright, Moshe Ben-Ezra, Yu-Wing Tai and Katsushi Ikeuchi. *High-resolution hyperspectral imaging via matrix factorization.* In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2329–2336, Colorado Springs, CO, USA, 2011. IEEE. (Cited on page 38.)

[Kendall 1990] Wilfrid S Kendall. *Probability, convexity, and harmonic maps with small image I : uniqueness and fine existence.* Proceedings of the London Mathematical Society, vol. 3, no. 2, pages 371–406, 1990. (Cited on page 82.)

[Keriven 2020] Nicolas Keriven, Damien Garreau and Iacopo Poli. *NEWMA : a new method for scalable model-free online change-point detection.* IEEE Transactions on Signal Processing, vol. 68, pages 3515–3528, 2020. (Cited on pages 68, 72, 79, 83, 93, 96 and 101.)

[Keshava 2002] Nirmal Keshava and John F Mustard. *Spectral unmixing.* IEEE signal processing magazine, vol. 19, no. 1, pages 44–57, 2002. (Cited on pages 38 and 42.)

[Khan 2019] Haidar Khan, Lara Marcuse and Bülent Yener. *Deep density ratio estimation for change point detection.* arXiv preprint arXiv :1905.09876, 2019. (Cited on pages 68 and 69.)

[Kingma 2014a] Diederik P Kingma and Jimmy Ba. *Adam : A method for stochastic optimization.* arXiv preprint arXiv :1412.6980, 2014. (Cited on pages 28, 53 and 72.)

[Kingma 2014b] Diederik P. Kingma and Max Welling. *Auto-Encoding Variational Bayes.* In Yoshua Bengio and Yann LeCun, editeurs, Proc. 2nd International Conference on Learning Representations (ICLR), Banff, AB, Canada, 2014. (Cited on page 72.)

[Krishnan 2009a] Dilip Krishnan and Rob Fergus. *Fast image deconvolution using hyper-Laplacian priors.* In Proc. Neural. Inf Process. Systems. (NIPS)., pages 1033–1041, 2009. (Cited on page 29.)

[Krishnan 2009b] Dilip Krishnan and Rob Fergus. *Fast image deconvolution using hyper-Laplacian priors.* Advances in Neural Information Processing Systems, vol. 22, pages 1033–1041, 2009. (Cited on page 45.)

[Krishnan 2011] Dilip Krishnan, Terence Tay and Rob Fergus. *Blind deconvolution using a normalized sparsity measure.* In Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pages 233–240. IEEE, 2011. (Cited on page 34.)

[Krizhevsky 2012] Alex Krizhevsky, Ilya Sutskever and Geoffrey E Hinton. *Imagenet classification with deep convolutional neural networks.* In Proc. NIPS., pages 1097–1105, 2012. (Cited on page 1.)

[Lanaras 2015] Charis Lanaras, Emmanuel Baltsavias and Konrad Schindler. *Hyperspectral super-resolution by coupled spectral unmixing.* In Proceedings of the IEEE International Conference on Computer Vision, pages 3586–3594, 2015. (Cited on page 38.)

[Lanza 2018] Alessandro Lanza, Serena Morigi, Federica Sciacchitano and Fiorella Sgallari. *Whiteness constraints in a unified variational framework for image restoration.* J. Math Imaging. Vis, vol. 60, no. 9, pages 1503–1526, 2018. (Cited on page 21.)

[Lanza 2020] Alessandro Lanza, Monica Pragliola and Fiorella Sgallari. *RESIDUAL WHITENESS PRINCIPLE FOR PARAMETER-FREE IMAGE RESTORATION.* Electron. Trans. Numer. Anal., vol. 53, pages 329–352, 2020. (Cited on pages 16 and 21.)

[LeCun 2015] Yann LeCun, Yoshua Bengio and Geoffrey Hinton. *Deep learning.* nature, vol. 521, no. 7553, pages 436–444, 2015. (Cited on page 1.)

[Levin 2009] Anat Levin, Yair Weiss, Fredo Durand and William T Freeman. *Understanding and evaluating blind deconvolution algorithms.* In Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pages 1964–1971, 2009. (Cited on page 28.)

[Li 2018] Shutao Li, Renwei Dian, Leyuan Fang and José M Bioucas-Dias. *Fusing Hyperspectral and Multispectral Images via Coupled Sparse Tensor Factorization.* IEEE Transactions on Image Processing, vol. 27, no. 8, pages 4118–4130, 2018. (Cited on page 38.)

[Li 2019] Shuang Li, Yao Xie, Hanjun Dai and Le Song. *Scan B-statistic for kernel change-point detection.* Sequential Analysis, vol. 38, no. 4, pages 503–544, 2019. (Cited on pages 79 and 96.)

[Li 2022] Jiaxin Li, Danfeng Hong, Lianru Gao, Jing Yao, Ke Zheng, Bing Zhang and Jocelyn Chanussot. *Deep Learning in Multimodal Remote Sensing Data Fusion : A Comprehensive Review.* arXiv preprint arXiv :2205.01380, 2022. (Cited on page 38.)

[Liu 2000] JG Liu. *Smoothing filter-based intensity modulation : A spectral preserve image fusion technique for improving spatial details.* International Journal of Remote Sensing, vol. 21, no. 18, pages 3461–3472, 2000. (Cited on page 37.)

[Liu 2013] Song Liu, Makoto Yamada, Nigel Collier and Masashi Sugiyama. *Change-point detection in time-series data by relative density-ratio estimation.* Neural Networks, vol. 43, pages 72 – 83, 2013. (Cited on pages 68 and 69.)

[Liu 2019a] Sicong Liu, Daniele Marinelli, Lorenzo Bruzzone and Francesca Bovolo. *A review of change detection in multitemporal hyperspectral images : Current techniques, applications, and challenges.* IEEE Geoscience and Remote Sensing Magazine, vol. 7, no. 2, pages 140–158, 2019. (Cited on pages 38, 42 and 44.)

[Liu 2019b] Wei Liu and Joonwhoan Lee. *A 3-D atrous convolution neural network for hyperspectral image denoising.* IEEE Trans. Geosci. Remote Sens., vol. 57, no. 8, pages 5701–5715, 2019. (Cited on page 24.)

[Liu 2022] Jianjun Liu, Zebin Wu, Liang Xiao and Xiao-Jun Wu. *Model Inspired Autoencoder for Unsupervised Hyperspectral Image Super-Resolution.* IEEE Transactions on Geoscience and Remote Sensing, 2022. (Cited on page 38.)

[Loncan 2015] Laetitia Loncan, Luis B de Almeida, José M Bioucas-Dias, Xavier Briottet, Jocelyn Chanussot, Nicolas Dobigeon, Sophie Fabre, Wenzhi Liao, Giorgio A Licciardi, Miguel Simoes*et al. Hyperspectral pansharpening : A review.* IEEE Geoscience and remote sensing magazine, vol. 3, no. 3, pages 27–46, 2015. (Cited on page 37.)

[Loukas 2015] Andreas Loukas, Andrea Simonetto and Geert Leus. *Distributed Autoregressive Moving Average Graph Filters.* IEEE Signal Processing Letters, vol. 22, no. 11, pages 1931–1935, 2015. (Cited on pages 105 and 109.)

[Lu 2014] Zhaosong Lu. *Iterative reweighted minimization methods for ℓ_p regularized unconstrained nonlinear programming.* Mathematical Programming, vol. 147, no. 1, pages 277–307, 2014. (Cited on pages 45 and 46.)

[Madrid Padilla 2023] Carlos Misael Madrid Padilla, Haotian Xu, Daren Wang, Oscar Hernan Madrid Padilla and Yi Yu. *Change point detection and inference in multivariate non-parametric models under mixing conditions.* Advances in Neural Information Processing Systems, vol. 36, 2023. (Cited on page 79.)

[Menon 2016] Aditya Menon and Cheng Soon Ong. *Linking losses for density ratio and class-probability estimation.* In International Conference on Machine Learning, pages 304–313. PMLR, 2016. (Cited on pages 69 and 70.)

[Moreno-Muñoz 2020] Pablo Moreno-Muñoz, David Ramírez and Antonio Artés-Rodríguez. *Continual learning for infinite hierarchical change-point detection.* In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3582–3586. IEEE, 2020. (Cited on page 68.)

[Müller 2007] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger and A. Weber. *Documentation Mocap Database HDM05.* Rapport technique CG-2007-2, Universität Bonn, June 2007. (Cited on page 100.)

[Neelamani 2004] Ramesh Neelamani, Hyeokho Choi and Richard Baraniuk. *ForWaRD : Fourier-wavelet regularized deconvolution for ill-conditioned systems.* IEEE Trans. Signal Process., vol. 52, no. 2, pages 418–433, 2004. (Cited on page 15.)

[Ng 2001] Andrew Ng, Michael Jordan and Yair Weiss. *On spectral clustering : Analysis and an algorithm.* Advances in neural information processing systems, vol. 14, 2001. (Cited on pages 105 and 109.)

[Nguyen 2018] Cuong V. Nguyen, Yingzhen Li, Thang D. Bui and Richard E. Turner. *Variational Continual Learning.* In International Conference on Learning Representations, 2018. (Cited on pages 68 and 71.)

[Page 1954] Ewan S Page. *Continuous inspection schemes.* Biometrika, vol. 41, no. 1/2, pages 100–115, 1954. (Cited on page 79.)

[Palsson 2017] Frosti Palsson, Johannes R Sveinsson and Magnus O Ulfarsson. *Multispectral and hyperspectral image fusion using a 3-D-convolutional neural network.* IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 5, pages 639–643, 2017. (Cited on page 38.)

[Peng 2021] Yidong Peng, Weisheng Li, Xiaobo Luo and Jiao Du. *Hyperspectral Image Superresolution Using Global Gradient Sparse and Nonlocal Low-Rank Tensor Decomposition With Hyper-Laplacian Prior.* IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pages 5453–5469, 2021. (Cited on page 45.)

[Pennec 2004] Xavier Pennec. *Probabilities and statistics on Riemannian manifolds : A geometric approach.* Rapport technique 5093, INRIA, 2004. (Cited on pages 81 and 82.)

[Pennec 2006a] Xavier Pennec. *Intrinsic statistics on Riemannian manifolds : Basic tools for geometric measurements.* Journal of Mathematical Imaging and Vision, vol. 25, pages 127–154, 2006. (Cited on page 82.)

[Pennec 2006b] Xavier Pennec, Pierre Fillard and Nicholas Ayache. *A Riemannian framework for tensor computing.* International Journal of computer vision, vol. 66, no. 1, pages 41–66, 2006. (Cited on pages 78, 94 and 106.)

[Prévost 2019] Clémence Prévost, Konstantin Usevich, Pierre Comon and David Brie. *Coupled tensor low-rank multilinear approximation for hyperspectral super-resolution.* In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5536–5540, Brighton, U.K., 2019. IEEE. (Cited on page 44.)

[Prévost 2020] Clémence Prévost, Konstantin Usevich, Pierre Comon and David Brie. *Hyperspectral super-resolution with coupled Tucker approximation : Recoverability and SVD-based algorithms.* IEEE Transactions on Signal Processing, vol. 68, pages 931–946, 2020. (Cited on pages 38 and 52.)

[Prévost 2022] Clémence Prévost, Ricardo A. Borsoi, Konstantin Usevich, David Brie, José C. M. Bermudez and Cédric Richard. *Hyperspectral Super-resolution Accounting for Spectral Variability : Coupled Tensor LL1-Based Recovery and Blind Unmixing of the Unknown Super-resolution Image.* SIAM Journal on Imaging Sciences, vol. 15, no. 1, pages 110–138, 2022. (Cited on pages 39, 42 and 43.)

[Qu 2018] Ying Qu, Hairong Qi and Chiman Kwan. *Unsupervised sparse Dirichlet-net for hyperspectral image super-resolution.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2511–2520, 2018. (Cited on page 38.)

[Rahimi 2007] Ali Rahimi and Benjamin Recht. *Random features for large-scale kernel machines.* Advances in neural information processing systems, vol. 20, 2007. (Cited on page 101.)

[Rasti 2021] Behnood Rasti, Yi Chang, Emanuele Dalsasso, Loic Denis and Pedram Ghamisi. *Image restoration for remote sensing : Overview and toolbox.* IEEE Geoscience and Remote Sensing Magazine, vol. 10, no. 2, pages 201–230, 2021. (Cited on page 2.)

[Reeves 1994] Stanley J Reeves. *Optimal space-varying regularization in iterative image restoration.* IEEE Trans. Image Process., vol. 3, no. 3, pages 319–324, 1994. (Cited on page 16.)

[Reeves 2007] Jaxk Reeves, Jien Chen, Xiaolan Wang, Robert Lund and Qi Qi Lu. *A Review and Comparison of Changepoint Detection Techniques for Climate Data.* Journal of applied meteorology and climatology, vol. 46, no. 6, pages 900 – 915, 2007. (Cited on page 4.)

[Roger 1996] R. E. Roger and J. F. Arnold. *Reliably estimating the noise in AVIRIS hyperspectral images.* Int J. Remote Sens., vol. 17, no. 10, pages 1951–1962, 1996. (Cited on pages 34 and 52.)

[Romano 2017] Yaniv Romano, Michael Elad and Peyman Milanfar. *The little engine that could : Regularization by denoising (RED).* SIAM Journal on Imaging Sciences, vol. 10, no. 4, pages 1804–1844, 2017. (Cited on pages 13, 39 and 48.)

[Rybach 2009] David Rybach, Christian Gollan, Ralf Schluter and Hermann Ney. *Audio segmentation for speech recognition using segment features.* In IEEE International Conference on Acoustics, Speech and Signal Processing, pages 4197–4200, 2009. (Cited on page 4.)

[Sarder 2006] Pinaki Sarder and Arye Nehorai. *Deconvolution methods for 3-D fluorescence microscopy images.* IEEE Signal Process. Mag., vol. 23, no. 3, pages 32–45, 2006. (Cited on page 15.)

[Schötz 2019] Christof Schötz. *Convergence rates for the generalized Fréchet mean via the quadruple inequality.* Electronic Journal of Statistics, vol. 13, pages 4280–4345, 2019. (Cited on page 81.)

[Segarra 2015] Santiago Segarra, Antonio G. Marques and Alejandro Ribeiro. *Distributed implementation of linear network operators using graph filters.* In Proc. Annual Allerton Conference on Communication, Control, and Computing, 2015. (Cited on pages 105 and 109.)

[Sharpnack 2013] James Sharpnack, Aarti Singh and Alessandro Rinaldo. *Changepoint Detection over Graphs with the Spectral Scan Statistic.* In Proc. International Conference on Artificial Intelligence and Statistics, volume 31, pages 545–553, 2013. (Cited on pages 105, 107 and 108.)

[Shaw 2003] Gary A Shaw and Hsiao-hua K Burke. *Spectral imaging for remote sensing.* Lincoln Laboratory Journal, vol. 14, no. 1, pages 3–28, 2003. (Cited on pages 2 and 37.)

[Shiga 1984] Kiyoshi Shiga. *Hadamard manifolds.* Geometry of Geodesics and Related Topics, vol. 3, pages 239–282, 1984. (Cited on page 82.)

[Shlezinger 2023] Nir Shlezinger, Jay Whang, Yonina C Eldar and Alexandros G Dimakis. *Model-based deep learning.* Proceedings of the IEEE, 2023. (Cited on page 1.)

[Shocher 2018] Assaf Shocher, Nadav Cohen and Michal Irani. *"zero-shot" super-resolution using deep internal learning.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3118–3126, 2018. (Cited on pages 40 and 50.)

[Shuman 2011] David I Shuman, Pierre Vandergheynst and Pascal Frossard. *Chebyshev polynomial approximation for distributed signal processing.* In Proc. IEEE International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011. (Cited on pages 105 and 109.)

[Sidiropoulos 2017] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis and C. Faloutsos. *Tensor decomposition for signal processing and machine learning.* IEEE Transactions on Signal Processing, vol. 65, no. 13, pages 3551–3582, 2017. (Cited on page 43.)

[Simões 2015] Miguel Simões, José Bioucas-Dias, Luis B Almeida and Jocelyn Chanussot. *A convex formulation for hyperspectral image superresolution via subspace-based regularization.* IEEE Trans. Geosci. Remote Sens., vol. 53, no. 6, pages 3373–3388, 2015. (Cited on pages 34, 38, 44 and 52.)

[Sinn 2012] Mathieu Sinn, Ali Ghodsi and Karsten Keller. *Detecting change-points in time series by maximum mean discrepancy of ordinal pattern distributions.* In Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence, pages 786–794, 2012. (Cited on page 79.)

[Smith 1994] Steven Smith. *Optimization Techniques on Riemannian Manifolds.* Fields Institute Communications, vol. 3, 1994. (Cited on page 80.)

[Song 2016] Yingying Song, David Brie, El-Hadi Djermoune and Simon Henrot. *Regularization parameter estimation for non-negative hyperspectral image deconvolution.* IEEE Trans. Image Process., vol. 25, no. 11, pages 5316–5330, 2016. (Cited on page 16.)

[Song 2019] Yingying Song, El-Hadi Djermoune, Jie Chen, Cédric Richard and David Brie. *Online deconvolution for industrial hyperspectral imaging systems.* SIAM J. Imaging Sci., vol. 12, no. 1, pages 54–86, 2019. (Cited on pages 16 and 115.)

[Sreehari 2016] Suhas Sreehari, S Venkat Venkatakrishnan, Brendt Wohlberg, Gregery T Buzzard, Lawrence F Drummy, Jeffrey P Simmons and Charles A Bouman. *Plug-and-play priors for bright field electron tomography and sparse interpolation.* IEEE Trans. Comput. Imaging, vol. 2, no. 4, pages 408–423, 2016. (Cited on page 32.)

[Stein 1981] Charles M Stein. *Estimation of the mean of a multivariate normal distribution*. Ann Stat., pages 1135–1151, 1981. (Cited on page 16.)

[Sugiyama 2007] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul Buenau and Motoaki Kawanabe. *Direct importance estimation with model selection and its application to covariate shift adaptation*. Advances in neural information processing systems, vol. 20, 2007. (Cited on page 68.)

[Sugiyama 2012] Masashi Sugiyama, Taiji Suzuki and Takafumi Kanamori. Density ratio estimation in machine learning. Cambridge University Press, 2012. (Cited on page 68.)

[Tartakovsky 2014] Alexander Tartakovsky, Igor Nikiforov and Michele Basseville. Sequential analysis : Hypothesis testing and changepoint detection. CRC press, 2014. (Cited on page 79.)

[Tekalp 1990] A Murat Tekalp and Gordana Pavlović. *Multichannel image modeling and Kalman filtering for multispectral image restoration*. Signal Process., vol. 19, no. 3, pages 221–232, 1990. (Cited on page 15.)

[Teodoro 2017] Afonso M Teodoro, José M Bioucas-Dias and Mário AT Figueiredo. *Scene-adapted plug-and-play algorithm with convergence guarantees*. In Proc. IEEE Int. Workshop. Mach Learn. Signal Process. (MLSP), pages 1–6. IEEE, 2017. (Cited on page 32.)

[Thiébaut 2005] Eric Thiébaut. *Introduction to image reconstruction and inverse problems*. In Optics in Astrophysics, pages 397–422. Springer, 2005. (Cited on page 15.)

[Thompson 1991] Alan M. Thompson, John C. Brown, Jim W Kay and D Michael Titterington. *A study of methods of choosing the smoothing parameter in image restoration by regularization*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 13, no. 04, pages 326–339, 1991. (Cited on page 16.)

[Townsend 2016] James Townsend, Niklas Koep and Sebastian Weichwald. *Pymanopt : A Python Toolbox for Optimization on Manifolds using Automatic Differentiation*. Journal of Machine Learning Research, vol. 17, no. 137, page 1–5, 2016. (Cited on page 95.)

[Truong 2020] Charles Truong, Laurent Oudre and Nicolas Vayatis. *Selective review of offline change point detection methods*. Signal Processing, vol. 167, page 107299, 2020. (Cited on pages 2, 4, 67, 68 and 77.)

[Tuzel 2008] Oncel Tuzel, Fatih Porikli and Peter Meer. *Pedestrian detection via classification on Riemannian manifolds*. IEEE transactions on pattern analysis and machine intelligence, vol. 30, no. 10, pages 1713–1727, 2008. (Cited on page 78.)

[Ulyanov 2018] Dmitry Ulyanov, Andrea Vedaldi and Victor Lempitsky. *Deep image prior*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 9446–9454, 2018. (Cited on pages 38 and 39.)

[Van De Ville 2011] Dimitri Van De Ville and Michel Kocher. *Nonlocal means with dimensionality reduction and SURE-based parameter selection.* IEEE Trans. Image Process., vol. 20, no. 9, pages 2683–2690, 2011. (Cited on page 16.)

[Veganzones 2016] Miguel A Veganzones, Miguel Simoes, Giorgio Licciardi, Naoto Yokoya, José M Bioucas-Dias and Jocelyn Chanussot. *Hyperspectral super-resolution of locally low rank images from complementary multisource data.* IEEE Transactions on Image Processing, vol. 25, no. 1, pages 274–288, 2016. (Cited on page 38.)

[Venkatakrishnan 2013] Singanallur V Venkatakrishnan, Charles A Bouman and Brendt Wohlberg. *Plug-and-play priors for model based reconstruction.* In Proc. IEEE Glob. Conf. Signal. Inf Process, pages 945–948, 2013. (Cited on pages 13, 16, 39 and 48.)

[Vivone 2018] Gemine Vivone, Rocco Restaino and Jocelyn Chanussot. *Full scale regression-based injection coefficients for panchromatic sharpening.* IEEE Transactions on Image Processing, vol. 27, no. 7, pages 3418–3431, 2018. (Cited on page 37.)

[Vogel 1996] Curtis R Vogel. *Non-convergence of the L-curve regularization parameter selection method.* Inverse Probl., vol. 12, no. 4, page 535, 1996. (Cited on page 16.)

[Von Luxburg 2007] Ulrike Von Luxburg. *A tutorial on spectral clustering.* Statistics and computing, vol. 17, pages 395–416, 2007. (Cited on pages 105 and 109.)

[Wald 2000] Lucien Wald. *Quality of high resolution synthesised images : Is there a simple criterion ?* In Proc. 3rd conf. “Fusion of Earth Data", pages 99–103. SEE/URISCA, 2000. (Cited on pages 29 and 54.)

[Wang 2002] Zhou Wang and Alan C Bovik. *A universal image quality index.* IEEE Signal Processing Letters, vol. 9, no. 3, pages 81–84, 2002. (Cited on page 55.)

[Wang 2004] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli*et al.* *Image quality assessment : from error visibility to structural similarity.* IEEE Trans. Image Process., vol. 13, no. 4, pages 600–612, 2004. (Cited on page 29.)

[Wang 2020a] Xiuheng Wang, Jie Chen, Cédric Richard and David Brie. *Learning spectral-spatial prior via 3DDnCNN for hyperspectral image deconvolution.* In Proc. IEEE Int. Conf. on Acoust, Speech, Signal Process (ICASSP), pages 2403–2407. IEEE, 2020. (Cited on pages 49 and 50.)

[Wang 2020b] Xiuheng Wang, Min Zhao and Jie Chen. *Hyperspectral Unmixing Via Plug-And-Play Priors.* In Proc. IEEE Int. Conf. Image Process. (ICIP), pages 1063–1067. IEEE, 2020. (Cited on pages 16 and 17.)

[Wang 2020c] Zhengjue Wang, Bo Chen, Ruiying Lu, Hao Zhang, Hongwei Liu and Pramod K Varshney. *FusionNet : An unsupervised convolutional variational network for hyperspectral and multispectral image fusion.* IEEE Transactions on Image Processing, vol. 29, pages 7565–7577, 2020. (Cited on page 38.)

[Wang 2021] Xiuheng Wang, Jie Chen, Qi Wei and Cédric Richard. *Hyperspectral Image Super-Resolution via Deep Prior Regularization with Parameter Estimation*. IEEE Trans. Circuits Syst, Video Technol., 2021. (Cited on pages 16, 25, 28, 38, 44 and 51.)

[Wang 2022a] Xiuheng Wang, Ricardo Augusto Borsoi, Cédric Richard and Jie Chen. *Deep Image Fusion Accounting for Inter-Image Variability*. In 2022 56th Asilomar Conference on Signals, Systems, and Computers, pages 645–649. IEEE, 2022. (Cited on pages 7 and 9.)

[Wang 2022b] Xiuheng Wang, Jie Chen and Cédric Richard. *Hyperspectral image super-resolution with deep priors and degradation model inversion*. In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2814–2818. IEEE, 2022. (Cited on pages 9 and 44.)

[Wang 2023a] Xiuheng Wang, Ricardo Augusto Borsoi and Cédric Richard. *Online change point detection on Riemannian manifolds with Karcher mean estimates*. In 2023 31st European Signal Processing Conference (EUSIPCO), pages 2033–2037. EURASIP, 2023. (Cited on pages 8, 9, 78, 80, 82 and 106.)

[Wang 2023b] Xiuheng Wang, Ricardo Augusto Borsoi, Cédric Richard and Jie Chen. *Change Point Detection with Neural Online Density-Ratio Estimator*. In ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1–5. IEEE, 2023. (Cited on pages 8, 9, 79 and 96.)

[Wang 2023c] Xiuheng Wang, Ricardo Augusto Borsoi, Cédric Richard and Jie Chen. *Deep Hyperspectral and Multispectral Image Fusion with Inter-image Variability*. IEEE Transactions on Geoscience and Remote Sensing, 2023. (Cited on pages 7 and 9.)

[Wang 2023d] Xiuheng Wang, Ricardo Augusto Borsoi, Cédric Richard and André Ferrari. *Détection de changements dans des signaux sur graphe à valeur dans des variétés riemanniennes*. In 29° COLLOQUE SUR LE TRAITEMENT DU SIGNAL ET DES IMAGES (GRETSI 2023), 2023. (Cited on pages 8 and 9.)

[Wang 2023e] Xiuheng Wang, Ricardo Augusto Borsoi, Cédric Richard and André Ferrari. *Distributed Change Point Detection in Streaming Manifold-valued Signals over Graphs*. In 2023 57th Asilomar Conference on Signals, Systems, and Computers. IEEE, 2023. (Cited on pages 8 and 9.)

[Wang 2023f] Xiuheng Wang, Jie Chen and Cédric Richard. *Tuning-free plug-and-play hyperspectral image deconvolution with deep priors*. IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pages 1–13, 2023. (Cited on pages 7 and 9.)

[Wang 2024a] Xiuheng Wang, Ricardo Augusto Borsoi and Cédric Richard. *Non-parametric Online Change Point Detection on Riemannian Manifolds*. In

International Conference on Machine Learning (ICML). PMLR, 2024. (Cited on pages 8 and 9.)

[Wang 2024b] Xiuheng Wang, Ricardo Augusto Borsoi and Cédric Richard. *Riemannian Diffusion Adaptation over Graphs with Application to Online Distributed PCA*. In ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2024. (Cited on pages 8, 9 and 115.)

[Wei 2015a] Qi Wei, José Bioucas-Dias, Nicolas Dobigeon and Jean-Yves Tourneret. *Hyperspectral and multispectral image fusion based on a sparse representation.* IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 7, pages 3658–3668, 2015. (Cited on page 38.)

[Wei 2015b] Qi Wei, Nicolas Dobigeon and Jean-Yves Tourneret. *Fast fusion of multi-band images based on solving a Sylvester equation.* IEEE Transactions on Image Processing, vol. 24, no. 11, pages 4109–4121, 2015. (Cited on pages 38 and 44.)

[Wei 2016] Qi Wei, José Bioucas-Dias, Nicolas Dobigeon, Jean-Yves Tourneret, Marcus Chen and Simon Godsill. *Multiband image fusion based on spectral unmixing.* IEEE Transactions on Geoscience and Remote Sensing, vol. 54, no. 12, pages 7236–7249, 2016. (Cited on page 49.)

[Wei 2020a] Kaixuan Wei, Angelica Aviles-Rivero, Jingwei Liang, Ying Fu, Carola-Bibiane Schönlieb and Hua Huang. *Tuning-free plug-and-play proximal algorithm for inverse imaging problems.* In Proc. Int. Conf. Mach Learn. (ICML)., pages 10158–10169. PMLR, 2020. (Cited on pages 16 and 17.)

[Wei 2020b] Wei Wei, Jiangtao Nie, Lei Zhang and Yanning Zhang. *Unsupervised recurrent hyperspectral imagery super-resolution using pixel-aware refinement.* IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pages 1–15, 2020. (Cited on pages 38 and 52.)

[Wei 2022] Song Wei and Yao Xie. *Online kernel CUSUM for change-point detection.* arXiv preprint arXiv :2211.15070, 2022. (Cited on page 79.)

[Wen 2023] Bihan Wen, Saiprasad Ravishankar, Zhizhen Zhao, Raja Giryes and Jong Chul Ye. *Physics-Driven Machine Learning for Computational Imaging [From the Guest Editor].* IEEE Signal Processing Magazine, vol. 40, no. 1, pages 28–30, 2023. (Cited on pages 1 and 6.)

[Xie 2020] Qi Xie, Minghao Zhou, Qian Zhao, Zongben Xu and Deyu Meng. *MHF-net : An interpretable deep network for multispectral and hyperspectral image fusion.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020. (Cited on pages 28 and 38.)

[Yamada 2013] Makoto Yamada, Akisato Kimura, Futoshi Naya and Hiroshi Sawada. *Change-point detection with feature selection in high-dimensional time-series data.* In Twenty-Third International Joint Conference on Artificial Intelligence, 2013. (Cited on pages 68 and 70.)

[Yan 2021] Longbin Yan, Min Zhao, Xiuheng Wang, Yuge Zhang and Jie Chen. *Object Detection in Hyperspectral Images*. IEEE Signal Process. Lett., vol. 28, pages 508–512, 2021. (Cited on page 2.)

[Yao 2020] Jing Yao, Danfeng Hong, Jocelyn Chanussot, Deyu Meng, Xiaoxiang Zhu and Zongben Xu. *Cross-attention in coupled unmixing nets for unsupervised hyperspectral super-resolution*. In European Conference on Computer Vision, pages 208–224. Springer, 2020. (Cited on page 38.)

[Yasuma 2010] Fumihito Yasuma, Tomoo Mitsunaga, Daisuke Iso and Shree K Nayar. *Generalized assorted pixel camera : postcapture control of resolution, dynamic range, and spectrum*. IEEE Trans. Image Process., vol. 19, no. 9, pages 2241–2253, 2010. (Cited on page 26.)

[Yokoya 2012] Naoto Yokoya, Takehisa Yairi and Akira Iwasaki. *Coupled nonnegative matrix factorization unmixing for hyperspectral and multispectral data fusion*. IEEE Transactions on Geoscience and Remote Sensing, vol. 50, no. 2, pages 528–537, 2012. (Cited on pages 37, 38, 44 and 52.)

[Yokoya 2016] N. Yokoya and A. Iwasaki. *Airborne hyperspectral data over Chikusei*. Rapport technique SAL-2016-05-27, Space Application Laboratory, University of Tokyo, Japan, May 2016. (Cited on page 26.)

[Yokoya 2017a] Naoto Yokoya, Claas Grohnfeldt and Jocelyn Chanussot. *Hyperspectral and multispectral data fusion : A comparative review of the recent literature*. IEEE Geoscience and Remote Sensing Magazine, vol. 5, no. 2, pages 29–56, 2017. (Cited on page 2.)

[Yokoya 2017b] Naoto Yokoya, Claas Grohnfeldt and Jocelyn Chanussot. *Hyperspectral and Multispectral Data Fusion : A comparative review of the recent literature*. IEEE Geoscience and Remote Sensing Magazine, vol. 5, no. 2, pages 29–56, 2017. (Cited on pages 38 and 42.)

[Zeng 2020] Linglin Zeng, Brian D Wardlow, Daxiang Xiang, Shun Hu and Deren Li. *A review of vegetation phenological metrics extraction using time-series, multispectral satellite data*. Remote Sensing of Environment, vol. 237, page 111511, 2020. (Cited on page 4.)

[Zhang 2016a] Hongyi Zhang, Sashank J Reddi and Suvrit Sra. *Riemannian SVRG : Fast stochastic optimization on Riemannian manifolds*. In Advances in Neural Information Processing Systems, pages 4592–4600, 2016. (Cited on page 80.)

[Zhang 2016b] Hongyi Zhang and Suvrit Sra. *First-order methods for geodesically convex optimization*. In Conference on Learning Theory, pages 1617–1638, 2016. (Cited on pages 78, 80, 84, 86, 87, 88 and 94.)

[Zhang 2017a] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng and Lei Zhang. *Beyond a gaussian denoiser : Residual learning of deep cnn for image denoising*. IEEE Trans. Image Process., vol. 26, no. 7, pages 3142–3155, 2017. (Cited on pages 24, 25, 49 and 50.)

[Zhang 2017b] Kai Zhang, Wangmeng Zuo, Shuhang Gu and Lei Zhang. *Learning deep CNN denoiser prior for image restoration*. In Proc. IEEE Conf. Comput.

Vis. Pattern Recognit. (CVPR), pages 3929–3938, 2017. (Cited on pages 16, 17 and 23.)

[Zhang 2018a] Jingzhao Zhang, Hongyi Zhang and Suvrit Sra. *R-spider : A fast Riemannian stochastic optimization algorithm with curvature independent rate*. arXiv preprint arXiv :1811.04194, 2018. (Cited on page 80.)

[Zhang 2018b] Lei Zhang, Wei Wei, Chengcheng Bai, Yifan Gao and Yanning Zhang. *Exploiting clustering manifold structure for hyperspectral imagery super-resolution*. IEEE Transactions on Image Processing, vol. 27, no. 12, pages 5969–5982, 2018. (Cited on page 38.)

[Zhang 2018c] Shaoquan Zhang, Jun Li, Heng-Chao Li, Chengzhi Deng and Antonio Plaza. *Spectral–spatial weighted sparse regression for hyperspectral image unmixing*. IEEE Transactions on Geoscience and Remote Sensing, vol. 56, no. 6, pages 3265–3276, 2018. (Cited on page 46.)

[Zhang 2020a] Lei Zhang, Jiangtao Nie, Wei Wei, Yong Li and Yanning Zhang. *Deep blind hyperspectral image super-resolution*. IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 6, pages 2388–2400, 2020. (Cited on page 38.)

[Zhang 2020b] Lei Zhang, Jiangtao Nie, Wei Wei, Yanning Zhang, Shengcai Liao and Ling Shao. *Unsupervised adaptation learning for hyperspectral imagery super-resolution*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3073–3082, 2020. (Cited on page 38.)

[Zhang 2021] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool and Radu Timofte. *Plug-and-play image restoration with deep denoiser prior*. IEEE Trans. Pattern Anal. Mach. Intell., 2021. (Cited on pages 16, 17 and 32.)

[Zhao 2016] Hang Zhao, Orazio Gallo, Iuri Frosio and Jan Kautz. *Loss functions for image restoration with neural networks*. IEEE Trans. Comput. Imaging, vol. 3, no. 1, pages 47–57, 2016. (Cited on pages 25 and 51.)

[Zhao 2019] Min Zhao, Jie Chen and Zhe He. *A laboratory-created dataset with ground truth for hyperspectral unmixing evaluation*. IEEE J. Sel. Top. Appl. Earth Observat. Remote Sens., vol. 12, no. 7, pages 2170–2183, 2019. (Cited on page 34.)

[Zhao 2021] Min Zhao, Xiuheng Wang, Jie Chen and Wei Chen. *A Plug-and-Play Priors Framework for Hyperspectral Unmixing*. IEEE Trans. Geosci. Remote Sens., 2021. (Cited on pages 16 and 17.)

[Zhou 2019] Pan Zhou, Xiaotong Yuan, Shuicheng Yan and Jiashi Feng. *Faster First-Order Methods for Stochastic Non-Convex Optimization on Riemannian Manifolds*. IEEE transactions on pattern analysis and machine intelligence, vol. 43, no. 2, pages 459–472, 2019. (Cited on page 80.)