

EXPLOITING STRUCTURE IN DYNAMICAL SYSTEMS FOR TRACKING AND DIMENSIONALITY REDUCTION

A Dissertation
Presented to
The Academic Faculty

By

Nicholas P. Bertrand

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
in
Electrical and Computer Engineering



School of Electrical and Computer Engineering
Georgia Institute of Technology
December 2019

Copyright © Nicholas P. Bertrand 2019

**EXPLOITING STRUCTURE IN DYNAMICAL SYSTEMS FOR TRACKING AND
DIMENSIONALITY REDUCTION**

Approved by:

Dr. Christopher J. Rozell, Advisor
Associate Professor, School of ECE
Georgia Institute of Technology

Dr. Mark A. Davenport
Associate Professor, School of ECE
Georgia Institute of Technology

Dr. Adam S. Charles
Assistant Professor, Department of
Biomedical Engineering
Johns Hopkins University

Dr. Omer Inan
Associate Professor, School of ECE
Georgia Institute of Technology

Dr. Samuel Shapero
Research Engineer, Electro-Optical
Systems Laboratory
Georgia Tech Research Institute

Date Approved: October 30, 2019

ACKNOWLEDGEMENTS

The opportunity to interact with the fun and incredibly talented people at Georgia Tech was one of the most rewarding aspects of my graduate school experience. I want to express my deepest gratitude to my thesis advisor, Chris Rozell, who went so far beyond his duties as an academic advisor and has been my unwavering advocate, mentor and role model. I sincerely appreciate his patience, understanding, generosity, transparency and dedication to his students. I also want to thank the members of my thesis committee: Mark Davenport, Omer Inan, Adam Charles and Sam Shapero, each of whom I have been extremely fortunate to work with over the years.

I am grateful to be surrounded by wonderful friends in lab whose camaraderie and support helped keep me sane through the ups and downs of grad student life: Abbie, Adam, Aurèle, Ayse, Greg, Han Lun, John, Kyle, Marissa, Matt, Mehgnchen, Pavel, Stefano, Andy, Chris, Darryl, Michael, Nishant, Namrata, Rakshith, and Steve, and many more. I am also thankful for my friends outside of lab who have made Atlanta feel like home: Fate, Hao, Mohit, Suzi, Thy and Tim.

Finally, I thank my family for their love and support even during the times when I was too busy or stressed to visit home. Above all, thank you to my loving wife Muriel who constantly reminds me of what is most important in life.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	iii
LIST OF FIGURES	vi
SUMMARY	viii
CHAPTER 1: INTRODUCTION	1
1.1 Contributions	2
1.2 Background: optimal transport	3
CHAPTER 2: EMD REGULARIZED DYNAMIC FILTERING	8
2.1 Introduction	9
2.2 Background	11
2.2.1 Dynamic filtering	11
2.3 Earth mover’s distance dynamic filtering	14
2.3.1 EMD-DF for nonnegative signals	15
2.3.2 EMD-DF for complex-valued signals	16
2.3.3 EMD computational complexity	18
2.4 Results	20
2.4.1 Target tracking	20
2.4.2 Tracking traveling waves	24
2.4.3 Frequency tracking	25
2.4.4 Tracking neural oscillations	30
2.4.5 Computational scalability	33
2.5 Summary and future work	34
CHAPTER 3: APPLICATIONS IN OT-REGULARIZED RPCA	37
3.1 Infrared search and track with optimal transport regularization	38
3.1.1 Introduction	38
3.1.2 Background	40
3.1.3 Optimal transport regularization for IRST	41

3.1.4	Results	47
3.1.5	Summary	53
3.2	Compressive RPCA for natural images	54
3.2.1	Background: compressive RPCA	54
3.2.2	Results: RPCA+UOT-DF with natural images	54
CHAPTER 4: EFFICIENT RANDOMIZED DIMENSIONALITY REDUCTION		58
4.1	Introduction	59
4.2	Methods	60
4.2.1	Stable manifold embeddings	60
4.2.2	Preservation of inner products	63
4.2.3	Isometry constant estimation	64
4.2.4	Estimation with complex-valued measurements: real-valued noise	65
4.2.5	Estimation with complex-valued measurements: complex-valued noise	68
4.2.6	Application: time series estimation from reduced measurements	69
4.2.7	Application: classification from reduced measurements	71
4.3	Results	72
4.3.1	Datasets	73
4.3.2	Distance preservation in practice	76
4.3.3	Application: neural imaging	78
4.3.4	Application: fluid flow classification	79
4.4	Discussion	81
CHAPTER 5: CONCLUSION		84
5.1	Summary	84
5.2	Limitations	85
5.3	Conclusions and future work	86
REFERENCES		88

LIST OF FIGURES

2.1	Stylized tracking scenario.	14
2.2	Example state sequence used in target tracking simulations.	21
2.3	Example state recovery in target tracking simulations.	21
2.4	Recovery performance over time in target tracking simulations.	23
2.5	Target tracking state recovery performance scaling behavior.	24
2.6	Performance in a detection task as a function of the number of compressive measurements, M	25
2.7	Algorithm performance as a function of target speed in target tracking simulations for different levels of noise variance, σ^2	26
2.8	Single step recovery of wavefronts from a 40×40 Kuramoto oscillator array via linear Gaussian measurements.	27
2.9	Examples of error computation used in frequency tracking simulations.	29
2.10	Mean spectral estimate error in frequency tracking simulations.	31
2.11	Mean frequency recovery error as a function of window length for simulated neural oscillation signals.	32
2.12	Time-frequency plots for a single channel of tetrode data recorded from the rat hippocampus.	33
2.13	Demonstration of computational speed up with EMD-DF Beckmann formulation.	35
3.1	Example frame of simulated infrared video and representative recovery of sparse targets.	50
3.2	Aggregate infrared tracking performance with noise, power, and ROC curve sweeps.	51

3.3	Performance of algorithms for simultaneous alignment and low-rank plus sparse recovery.	53
3.4	Separation of a moving subject from its environment using compressive measurements of a real video clip.	56
4.1	Explanatory illustration of RF algorithm.	61
4.2	The isometry constant δ for RF vs LPF on an array of diverse datasets. . . .	77
4.3	Calcium imaging simulation imaging apparatus, example data, and sweep results.	80
4.4	Vorticity simulation imaging apparatus, example data, and sweep results. . .	82

SUMMARY

The goal of this work is to leverage the underlying structure in observations from dynamical systems to improve tracking performance and efficiently perform dimensionality reduction. First, we propose the use of the earth mover's distance (EMD) as a dynamics regularizer for sparse signal tracking. Traditional tracking algorithms such as the Kalman filter use the ℓ_p -norm to evaluate similarity between the signal estimate and prediction from the dynamics model. However, the ℓ_p -norm does not effectively exploit the geometric structure or ordering present in the coefficients in many applications such as imaging and frequency estimation. The EMD is a natural alternative dynamics regularizer which is inherently aware of the structure between elements by way of a user-defined cost matrix. In this work, we formulate an EMD-based tracking algorithm and evaluate its performance in imaging, wavefront, and frequency tracking scenarios with applications to electrophysiology. Next, we utilize optimal transport formulations to build on other types of structure by regularizing the sparse plus low rank problem in robust principle components analysis. This approach is validated through simulations on natural and infrared video sequences. Finally, we study an efficient dimensionality reduction scheme based on random projections for observations from a dynamical system which has converged to a low-dimensional attractor manifold. Performance is evaluated via tasks on synthetic neural imaging and fluid flow data.

CHAPTER 1

INTRODUCTION

The task of extracting salient information from time-varying data is a central problem in statistical signal processing and its applications permeate science and engineering. Neuroscientists are interested in determining the precise frequencies at which neurons in the brain oscillate using recordings from electrode arrays; aircraft pilots rely sensor data to pinpoint the locations of incoming threats; physicists classify fluid flow using images from high-speed cameras. Each of these examples poses unique challenges such as low signal-to-noise ratio (SNR), indirect measurements, and real-time or low-power running requirements. In order to address these challenges, it is crucial that signal processing algorithms can leverage a-priori information to compensate for noise and missing data. For example, a target tracking algorithm should leverage the fact that a target is unlikely to teleport across the scene instantaneously. In this thesis, we explore methods for leveraging two types of general structure to improve performance in signal processing tasks: the geometric structure among signal elements and generic low-dimensional (manifold) structure for dynamical systems evolving on an attractor.

The first type of structure that we will study is the geometric relationship between elements in streaming measurements of a time varying signal. We assume that the signal of interest is sparse (i.e., that it may be represented using only a small number of atoms in a larger dictionary), and that it evolves according to a dynamics model. The goal in this problem is to estimate the unknown signal from measurements by leveraging the measurement and dynamics models. Many current methods use the ℓ_p -norm as a metric with which to compare candidate signal estimates to the prediction. The ℓ_p -norm however is unaware of the geometric relationship between signal elements. Consider, for example, an

imaging scenario where we wish to track a single pixel moving through a scene. An ℓ_p -norm based regularizer assigns equal penalties to any prediction in which the target is not precisely in the correct location regardless of how far away the erroneous pixel is. Similarly, when tracking sparse frequency targets, an ℓ_p -norm based regularizer on frequencies is agnostic to how similar the estimated frequency is to the frequency of interest.

The second type of structure studied in this thesis is that of a dynamical system which has converged to a low dimensional attractor manifold. Manifolds may be intuitively understood as generalizations of surfaces (in two-dimensions) to higher dimensions, and have proven to be effective in describing systems whose state depends on a relatively small set of parameters. For example, [1] studies the manifold structure of images of objects at different angles. Technological developments over the last few decades have resulted the ubiquity of sensors which operate at high temporal and spatial resolution, leading to massive amounts of data from which scientists wish to draw meaningful conclusions. In fact, the amount of data which we are now able to collect is quickly outpacing our ability to store, process, and transmit. Dimensionality reduction is the crucial task of finding a lower dimensional representation of data recorded in a high-dimensional ambient space while leaving the information of interest intact. Manifold learning one type of dimensionality reduction which aims to discover the structure of a manifold from data and a host of algorithms exist toward this end. However, the majority of manifold learning algorithms are computationally expensive and require batches of data.

1.1 Contributions

In Chapter 2, we propose the earth mover's distance (EMD) as an alternative regularizer for tracking time-varying sparse signals and introduce a new causal sparse tracking algorithm: earth mover's distance dynamic filtering (EMD-DF). In essence, the EMD measures the amount of energy required to transform one signal into another, allowing the algorithm to account for relevant geometric relationships of the sparse coefficient space. The proposed

EMD-DF is therefore most appropriate in situations where online estimation is necessary (e.g., closed-loop systems) and where there is a natural geometry to the coefficient space that should be respected (e.g., meaningful ordering).

In Chapter 3, we explore the infrared search and track problem in which the goal is to locate a small target from infrared imagery captured from a moving platform. This is a particularly difficult problem due to low SNR, small target size, and the presence of non-stationary background clutter. In this aim, we extend the methods described in Chapter 2 to incorporate additional types of structure, such as low-rank, in order to jointly remove noise and clutter from the images and track targets of interest. We also develop algorithm variants which allow for affine transformations of the observed data which results in improved robustness to instability of the imaging sensor.

Finally, in Chapter 4, we adapt theoretical results for stable manifold embeddings via random projections to address the problem of efficient dimensionality reduction for dynamical systems. We study an algorithm which we call randomized filtering which is an analog for manifold-modeled data of the fast Johnson-Lindenstrauss (JL) transform for point cloud data. We demonstrate through a series of simulations on synthetic neural imaging and fluid flow data that randomized filtering retains enough information to perform important domain specific tasks like spike recovery and classification directly in the reduced space.

1.2 Background: optimal transport

Optimal transport (OT) is a body of literature that grew out of the seminal work by Monge [2]. OT techniques may be used to compare signals whose elements share a geometric structure, for example Euclidean distance between pixels in an image. Consider for example the problem of tracking a target in pixel space. If two images \mathbf{X}_1 and \mathbf{X}_2 containing non-overlapping targets are compared with the commonly used ℓ_p norm via $d(\mathbf{X}_1, \mathbf{X}_2) = \|\mathbf{X}_1 - \mathbf{X}_2\|_p^p$, then the distance will be the same regardless of how far away the targets are

from each other. In contrast, an OT-based distance function would correctly reflect the relative distance of the targets.

One well-known OT formulation is the EMD. Intuitively, if we visualize the first signal as being composed of piles of dirt and the second as holes, the EMD computes the minimum amount of *work* needed to fill the holes with dirt. The EMD has recently been increasingly used in a variety of applications such as image and histogram comparison [3, 4, 5], as well as for sparse inverse problems [6, 7, 8]. We consider here the EMD for discrete signals which is defined for two nonnegative input signals \mathbf{x} and \mathbf{y} by the optimization program

$$\begin{aligned}
 d_{\text{emd}}(\mathbf{x}, \mathbf{y}) &= \min_{\mathbf{F}} \sum_{ij} R_{ij} F_{ij} \\
 &\text{subject to } F_{ij} \geq 0, \\
 &\sum_j F_{ij} \leq \mathbf{x}[i], \quad \sum_i F_{ij} \leq \mathbf{y}[j], \\
 &\sum_{ij} F_{ij} = \min(\|\mathbf{x}\|_1, \|\mathbf{y}\|_1), \tag{1.1}
 \end{aligned}$$

where mass flow from the i -th element of \mathbf{x} to the j -th element of \mathbf{y} , represented by the elements of the matrix $\mathbf{F} = (F_{ij})$, incurs a cost given by R_{ij} . Often, this cost is defined as $R_{ij} = d(\mathbf{c}_i, \mathbf{c}_j)$ where d is a distance metric and \mathbf{c}_i represents the discretized support coordinates of location i . The EMD is the cost associated with the minimum cost flow under four constraints. The first constraint specifies that flows must be positive. The second and third constraints enforce conservation of mass (e.g., the total mass flowing out of the i -th position of \mathbf{x} is bounded by $\mathbf{x}[i]$). The final constraint states that the total amount of flow must be equal to the total mass of the smaller operand signal. This prevents the trivial solution where no mass flows, resulting in a cost of zero. Although traditional formulations of the EMD require distribution-like signals (i.e., ones with equal mass), the formulation in (1.1) allows for signals of unequal mass via the min function in the last constraint. A key property to note is that the EMD is inherently aware, by way of the distance matrix (R_{ij}) , of

the geometric relationship between elements in its operands. With non-trivial choices for this distance matrix, the cost to move mass to nearby elements is different than the cost to move mass over a longer distance. This is in stark contrast to ℓ_p metrics, and is the primary motivation for its use as a tracking regularizer.

The traditional EMD formulation presented here involves solving for $\mathcal{O}(N^2)$ flow variables, which has the potential to be computationally prohibitive for large problems. While recent computational advances exploiting entropic regularization [9] enable fast numerical approaches to EMD problems (even in variational settings) [10, 11], these algorithms only approximate the EMD calculation and we defer consideration of these advances in tracking problems for future work. However, for applications whose distance cost d is restricted to be the standard Euclidean distance (e.g., video), geometric structure can be exploited to also reduce the optimization variable complexity in exact EMD solutions from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$. In particular, the EMD problem can be reinterpreted as a fluid dynamics flux problem known as the *Beckmann problem* [12]. This problem searches for the optimal flux configuration of a fluid flowing between a source and a sink (i.e., the input arguments of the EMD problem). The enormous reduction in variables is therefore a result of physical fluid constraints that, by virtue of its representation, restricts point masses from “teleporting” across space. We note that this formulation has recently been applied [13] for computing the EMD between large-scale images.

We will consider the discretization of a D -dimensional flux field $\mathbf{M} \in \mathbb{R}^{N \times D}$, whose support is evenly gridded. For example, for images when $D = 2$, the columns of a 2-dimensional flux field (with equally gridded horizontal and vertical spacings) may be reorganized into two flux matrices $\mathbf{M}_x, \mathbf{M}_y \in \mathbb{R}^{n_x \times n_y}$, each one representing the flux field in each direction and where $N = n_x n_y$. The notion of how much each point in the flux field

is a source or a sink is mathematically described by a linear divergence operator, defined as

$$\begin{aligned} \operatorname{div}(\mathbf{M})[i, j] &= (M_x[i, j] - M_x[i - 1, j]) \\ &+ (M_y[i, j] - M_y[i, j - 1]), \end{aligned} \quad (1.2)$$

where zero-flux boundary conditions are enforced (i.e., $M[i, j] = 0$ whenever i or j falls outside the support). Now, we can re-express the EMD definition in (1.1) as the Beckmann problem:

$$d_{\text{emd}}(\mathbf{x}, \mathbf{y}) = \min_{\mathbf{M}} \|\mathbf{M}\|_{2,1} \quad \text{subject to} \quad \operatorname{div}(\mathbf{M}) + \mathbf{y} - \mathbf{x} = \mathbf{0}, \quad (1.3)$$

where the rows of \mathbf{M} contain points in a D -dimensional vector field and $\|\mathbf{M}\|_{2,1} := \sum_{i=1}^N \|\mathbf{m}_i\|_2$ denotes the sum of their Euclidean norms. This optimization searches for the minimal vector field configuration \mathbf{M} whose inward and outward flux contributions are defined by \mathbf{x} and \mathbf{y} respectively.

While dramatically reducing the number of optimization variables for exact EMD calculation under Euclidean distance cost, a key limitation of the Beckmann formulation is that applications are limited to inputs that to lie in the probability simplex (i.e., vectors that sum to one). For example, in radar tracking, targets can spontaneously pop in and out, therefore the total energy is not constant over time. The trivial solution of normalizing their total energy (to fit these formulations) is a poor choice because individual signal energies will scale arbitrarily. To exploit the efficiency of (1.3), we will require a reformulation to adopt the type of constraints in (1.1) that more gracefully allows it to handle applications where the total energy changes with time.

Other recent works [14] incorporate optimal transport regularizers in inverse problems using the Sinkhorn algorithm [15, 9]. However, the work presented here is distinct in several subtle but important ways. First, we propose a partial transport regularizer instead of the balanced transport regularizer found in the literature [11, 14, 16]. This is important in

real-world tracking applications where the total signal magnitude varies in time, compared to signals whose mass remains static. Second, compared to Sinkhorn approaches, our proposed Beckmann formulation provides an alternative numerical approach that is more efficient (by virtue of the significantly smaller optimization space, i.e., $\mathcal{O}(N^2)$ versus $\mathcal{O}(N)$). Sinkhorn approaches are well known to trade off accuracy with speed via entropy-regularization. In contrast, the Beckmann formulation reflects the true optimal transport distance (subject only to negligible discretization errors).

CHAPTER 2

EMD REGULARIZED DYNAMIC FILTERING

In this chapter, we propose the use of the earth mover’s distance (EMD) as an alternative tracking regularizer for causal tracking when there is a natural geometry to the coefficient space that should be respected (e.g., meaningful ordering) ¹. In this chapter, we assume a sparse signal model due to its state-of-the-art performance in a variety of application domains. Our proposed earth mover’s distance dynamic filtering (EMD-DF) algorithm is a causal approach to tracking time-varying sparse signals that includes two variants: one which uses the traditional EMD as a tracking regularizer for sparse nonnegative signals, and a relaxation which allows for complex-valued signals. In addition, we present a computationally efficient formulation of EMD-DF (based on optimal transport (OT) theory), improving computational scalability for large state vectors. Through a series of simulations, we demonstrate the advantages of EMD-DF compared to existing methods on tracking sparse targets in state vectors with multiple dimensions and tracking sparse frequencies in time-series estimation. In the context of frequency tracking, we illustrate the advantages of EMD-DF in tracking neural oscillations in electrophysiology recordings from rodent brains. We demonstrate that EMD-DF causally produces representations that achieve much higher time-frequency resolution than traditional causal linear methods such as the short time Fourier transform (STFT).

¹The work here was presented in [17, 18, 19] and is in collaboration with Dr. Adam Charles, Dr. John Lee, and Pavel Dunn. NPB, ASC, and JL contributed equally. Specifically, ASC posed the idea of using the EMD as a dynamics regularizer and coded the early toy imaging simulations; NPB adapted the formulation to allow for practical considerations such as signals with complex-valued elements and developed solver code for these variations. NPB also ran extensive simulations in imaging, wavefront, and frequency tracking applications. JL developed a partial transport formulation based on the Beckmann problem for improved computational efficiency and provided corresponding solver code; PD provided code for generating early versions of the synthetic frequency data.

2.1 Introduction

Tracking algorithms (also called dynamic filtering) aim to improve the performance of statistical inference procedures for time series by incorporating information from a dynamics model which describes how the signal evolves. For example, the widely used Kalman filter [20] efficiently produces optimal estimates from linear measurements under additive Gaussian noise in the measurement and dynamics models. However, in contrast to these classic models, sparsity models are extremely non-Gaussian and have become increasingly popular due to their state-of-the-art performance in a variety of problems (for example in image processing [21] and compressive sensing [22]). Sparse inference problems with static data vectors have been studied in depth, resulting in many algorithmic advances and performance guarantees [23, 22, 24, 25, 26].

In the spirit of the Kalman filter, sparse tracking algorithms have also been introduced for dynamic filtering when the sparse signals are time-varying and have shown utility in practice [27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39]. basis pursuit denoising with dynamic filtering (BPDN-DF) [38] is one example of a recent algorithm which penalizes differences between the data and the prediction with an ℓ_p -norm to incorporate a dynamics model into the regularization. However, in many applications with discretized domains, such ℓ_p -norm regularizers disproportionately penalize predictions with slight mismatch in the signal support because they do not incorporate any knowledge of meaningful geometry (when it exists) into the penalty. Consider, for example, an imaging scenario where we wish to track a single pixel moving through a scene. An ℓ_p -norm based regularizer assigns equal penalties to any prediction in which the target is not precisely in the correct location regardless of how far away the erroneous pixel is. Similarly, when tracking sparse frequency targets, an ℓ_p -norm based regularizer on frequencies is agnostic to how similar the estimated frequency is to the frequency of interest.

In this work, we propose the earth mover’s distance (EMD) as an alternative regularizer

for tracking time-varying sparse signals and introduce a new causal sparse tracking algorithm: earth mover’s distance dynamic filtering (EMD-DF). In essence, the EMD measures the amount of energy required to transform one signal into another, allowing the algorithm to account for relevant geometric relationships of the sparse coefficient space. The proposed EMD-DF is therefore most appropriate in situations where online estimation is necessary (e.g., closed-loop systems) and where there is a natural geometry to the coefficient space that should be respected (e.g., meaningful ordering). The main contribution of this chapter is the introduction of EMD-DF at the algorithmic level, the casting of various versions (e.g., nonnegative coefficients, complex-coefficients) of the problem into tractable numerical optimizations, and the formulation of highly-efficient approaches that reduce computational complexity for large-scale problems. Compared to a recent method [14] employing *balanced* EMD-regularization based on the celebrated *Sinkhorn distances* [9], we propose to use the richer *partial* transport model [40] based on *Beckmann’s formulation* [12, 41], to allow greater numerical efficiency without compromising solution accuracy. Through a series of simulations, we demonstrate the advantages of EMD-DF compared to existing methods on tracking sparse targets in large state vectors with multiple dimensions and tracking sparse frequencies in time-series estimation. In the context of frequency tracking, we illustrate the advantages of EMD-DF in tracking neural oscillations in electrophysiology recordings from rodent brains. We demonstrate that EMD-DF causally produces representations that achieve much higher time-frequency resolution than traditional causal linear methods such as the short time Fourier transform (STFT). Specifically, this work advances preliminary introduction and limited demonstration of this approach [17, 18] to include novel formulations to dramatically reduce computational complexity, more thorough characterization on synthetic datasets, and application to real electrophysiology data.

2.2 Background

2.2.1 Dynamic filtering

Dynamic filtering is the problem of recovering a time varying signal from noisy measurements with the aid of a dynamics model. Here, we consider the linear observation model

$$\mathbf{y}_n = \mathbf{A}_n \mathbf{x}_n + \sigma \epsilon_n, \quad (2.1)$$

where for each time step n , \mathbf{x}_n is the underlying signal, \mathbf{A}_n is a linear observation operator, $\sigma \epsilon_n$ is Gaussian measurement noise with variance σ^2 , and \mathbf{y}_n is the resulting measurement vector. We model the signal as evolving according to a dynamics function g as

$$\mathbf{x}_{n+1} = g(\mathbf{x}_n) + \boldsymbol{\eta}_n, \quad (2.2)$$

where $\boldsymbol{\eta}_n$ is a noise vector called the *innovations* that accounts for inaccurate modeling of the dynamics. When g is linear and the signal, observation noise and innovations are Gaussian distributed, the classical Kalman filter provides an efficient way to compute the optimal (i.e., minimum expected ℓ_2 error) estimate taking into account all measurements up to the current time step. The estimate produced by the Kalman filter may be expressed as

$$\hat{\mathbf{x}}_n = \operatorname{argmin}_{\mathbf{x}} \|\mathbf{y}_n - \mathbf{A}_n \mathbf{x}\|_{\mathbf{R}_n}^2 + \|\mathbf{x} - \mathbf{G}_n \hat{\mathbf{x}}_{n-1}\|_{(\mathbf{Q}_n + \mathbf{G}_n \mathbf{P}_{n-1} \mathbf{G}_n^T)^{-1}}, \quad (2.3)$$

where \mathbf{G}_n is the linear dynamics operator, \mathbf{R}_n and \mathbf{Q}_n are covariance matrices of the noise and innovations, and $\hat{\mathbf{x}}_{n-1}$ and \mathbf{P}_{n-1} are the previous signal estimate and its covariance. Here, we use $\|\cdot\|_{\mathbf{B}}$ to denote the norm induced by the positive-definite matrix \mathbf{B} (i.e., $\|\mathbf{a}\|_{\mathbf{B}}^2 = \mathbf{a}^T \mathbf{B} \mathbf{a}$). Thus, the Kalman filter may be interpreted as the solution to a least-squares problem which is regularized by the dynamics model. The Kalman filter and its extensions [42, 43] have been used exhaustively in many applications throughout science

and engineering.

Sparsity models have also received much attention from the research community in recent years. A vector $\mathbf{x} \in \mathbb{C}^N$ is said to be sparse if only a few of its elements are non-zero (i.e., $\|\mathbf{x}\|_0 \ll N$, where $\|\cdot\|_0$ indicates the number of non-zero elements in the operand). Suppose \mathbf{y} contains noisy observations of \mathbf{x} through a linear measurement operator $\mathbf{A} \in \mathbb{C}^{M \times N}$. For example, results in the field of compressed sensing show that under certain conditions on \mathbf{A} , \mathbf{x} may be recovered from \mathbf{y} even when $M \ll N$. Of the many sparse inverse algorithms that exist (e.g., [44, 24, 25, 45, 46, 47, 48]), one popular optimization-based approach is basis pursuit denoising (BPDN) [23]:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1. \quad (2.4)$$

We may interpret this as the solution to a least-squares problem with the ℓ_1 -norm as a sparsifying regularizer with parameter $\lambda > 0$ controlling the trade-off between measurement fidelity and signal sparsity. Reweighted- ℓ_1 (RWL1) is one notable variation of BPDN that uses a hierarchical model called a Laplacian Scale Mixture to impose separate distributions on individual coefficients [49, 50]. Expectation maximization is then employed to estimate the signal, resulting in an iterative refinement of the signal via

$$\hat{\mathbf{x}}^k = \underset{\mathbf{x}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda_0 \sum_{i=1}^N \boldsymbol{\lambda}^{(k)}[i] |\mathbf{x}[i]|, \quad (2.5)$$

where k represents the algorithmic iteration, and $\boldsymbol{\lambda}^{k+1}[i] = \frac{\beta}{|\hat{\mathbf{x}}^k[i]| + \eta}$. RWL1 can yield sparser solutions than BPDN, however this improved performance comes at the cost of solving multiple optimization programs.

While the sparse recovery techniques discussed above infer static sparse vectors, recent work has also extended these ideas to dynamic filtering for sparse time-varying signals. Early work in this area included batch (i.e., non-causal) approaches [51, 52, 53, 54, 55, 56, 57] and modifications to the causal Kalman filter [29, 30]. More recent causal approaches

include BPDN-DF which provides theoretical convergence guarantees, and reweighted- ℓ_1 dynamic filtering (RWL1-DF) which was found to be more robust to model mismatch [38]. We describe each of these algorithms in more detail as they provide intuition for the contributions in this chapter. Furthermore, they offer state-of-the-art performance for causal algorithms, thus serving as points of comparison in the experiments to be described later.

BPDN-DF modifies standard BPDN with the addition of a tracking regularizer:

$$\hat{\mathbf{x}}_n = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y}_n - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 + \gamma \|\mathbf{x} - \tilde{\mathbf{x}}_n\|_2^2, \quad (2.6)$$

where $\tilde{\mathbf{x}}_n = g(\hat{\mathbf{x}}_{n-1})$ is the prediction produced using the dynamics function g . This additional term encourages solutions which adhere to the dynamics model. Similarly, RWL1-DF modifies RWL1 by injecting dynamics into the recovery process via an additional term in the denominator of the weight update equation:

$$\lambda_n^{k+1}[i] = \frac{\xi}{\beta |\hat{\mathbf{x}}_n^k[i]| + |\tilde{\mathbf{x}}_n[i]| + \eta}, \quad (2.7)$$

where $\hat{\mathbf{x}}_n^k$ is the estimate from the previous iteration of reweighting. Both BPDN-DF and RWL1-DF show improved performance in the recovery of time varying signals, and RWL1-DF tends to be more robust to model mismatch in the dynamics [38]. However, because the reweighting is done pointwise, RWL1-DF fails to capture the geometric relationship between neighboring signal elements. One might attempt to resolve this problem by blurring the prediction as a crude approximation to favor nearby vector indices (e.g., as is done in [58]), however this introduces an undesirable trade-off between tolerance to support mismatch and retention of dynamics information. At one extreme, using a narrow blurring kernel only predicts static signals. Conversely, for wide blurring kernels, detail in the prediction is washed out and dynamics information is lost. Furthermore, this approach introduces additional algorithm parameters to tune and lowers the sparsity penalty in patches of elements around the active elements in the prediction, potentially resulting in a less sparse

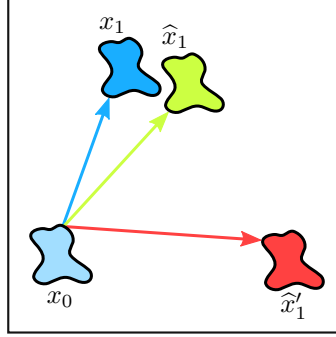


Figure 2.1: Stylized tracking scenario. The regions labeled x_0 and x_1 denote the ground truth signal for time steps $n = 0, 1$, and \hat{x}_1 and \hat{x}'_1 denote two candidate estimates. A tracking regularizer should favor \hat{x}_1 which is spatially closer to the ground truth than \hat{x}'_1 . However, ℓ_p -norm based tracking regularizers do not reflect this, since $\|x_1 - \hat{x}_1\|_p^p = \|x_1 - \hat{x}'_1\|_p^p$.

solution.

2.3 Earth mover's distance dynamic filtering

One drawback of existing tracking algorithms is a lack of robustness to small mismatches in the locations of the active signal coefficients, which is problematic when there is a geometric relationship or ordering among the coefficients. For instance, consider the image tracking scenario depicted in Figure 2.1. We should encourage signal estimates with active pixels geometrically close to the ground truth (even if the locations do not match exactly) and we should penalize estimates with active pixels that are far away. Unfortunately, each candidate estimate looks equally plausible when the error is measured with an ℓ_p -norm on the difference vector (e.g., mean-squared error). Similarly, in the problem of tracking a set of time varying frequencies, the ordering of the frequencies in the discrete Fourier transform (DFT) matrix results in a geometric relationship among the DFT coefficients which is not effectively utilized with ℓ_p -norm regularizers. The EMD is a natural alternative regularizer in both of these scenarios.

We propose a new tracking algorithm, EMD-DF, where the causal estimate of the signal

at time n is given by:

$$\hat{\mathbf{x}}_n = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y}_n - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 + \gamma d_{\text{emd}}(\mathbf{x}, \tilde{\mathbf{x}}_n), \quad (2.8)$$

where $\tilde{\mathbf{x}}_n = g(\hat{\mathbf{x}}_{n-1})$ is the prediction from the previous time step. EMD-DF has a similar structural form as BPDN-DF at first glance, but the use of an EMD penalty instead of an ℓ_2 dynamics regularizer is non-trivial because the evaluation of the EMD itself requires the solution of an optimization program. Incorporating the EMD into a dynamic filtering algorithm for common signals of interest presents three challenges that require technical innovation: the traditional formulation of the EMD 1) operates exclusively on nonnegative vectors, 2) operates on real-valued vectors, and 3) requires a prohibitive computational complexity for inclusion inside an optimization program. We address each of these issues in the following subsections.

2.3.1 EMD-DF for nonnegative signals

For the case where the signal of interest is nonnegative, we can substitute the definition of the EMD into (2.8) and arrive at the following joint optimization over the signal estimate and the EMD flow variables \mathbf{F} :

$$\begin{aligned} \hat{\mathbf{x}}_n &= \underset{\mathbf{x}, \mathbf{F}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y}_n - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 + \gamma \sum_{ij} R_{ij} F_{ij} \\ &\text{subject to } F_{ij} \geq 0, \\ &\sum_j F_{ij} \leq \mathbf{x}[i], \quad \sum_i F_{ij} \leq \tilde{\mathbf{x}}_n[j], \\ &\sum_{ij} F_{ij} = \min(\|\mathbf{x}\|_1, \|\tilde{\mathbf{x}}_n\|_1). \end{aligned} \quad (2.9)$$

Here, we adopt the notation $\operatorname{argmin}_{\mathbf{x}, \mathbf{F}} h(\mathbf{x}, \mathbf{F}) = \operatorname{argmin}_{\mathbf{x}} [\min_{\mathbf{F}} h(\mathbf{x}, \mathbf{F})]$. The last constraint is non-linear and thus complicates the evaluation of the optimization program. To

address this challenge, we replace the nonlinear equality constraint by introducing a slack variable as follows:

$$\begin{aligned}
\hat{\mathbf{x}}_n &= \underset{\mathbf{x}, \mathbf{F}, u}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y}_n - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 + \gamma \sum_{ij} R_{ij} F_{ij} - \mu u \\
&\text{subject to } F_{ij} \geq 0, \\
&\sum_j F_{ij} \leq \mathbf{x}[i], \quad \sum_i F_{ij} \leq \tilde{\mathbf{x}}_n[j], \\
&\sum_{ij} F_{ij} = u, \quad \|\mathbf{x}\|_1 \geq u, \quad \|\tilde{\mathbf{x}}_n\|_1 \geq u.
\end{aligned} \tag{2.10}$$

The additional term in the objective function encourages the slack variable u to be as large as possible, while the additional constraints force u to be bounded above by $\|\mathbf{x}\|_1$ and $\|\tilde{\mathbf{x}}_n\|_1$. Hence, for an appropriate value of μ , u will be equal to $\min\{\|\mathbf{x}\|_1, \|\tilde{\mathbf{x}}_n\|_1\}$, as desired. If the chosen value of μ is too small, the reduction in the objective function by increasing u is outweighed by the additional mass transport cost so the solution of u may be smaller than the mass in each of the EMD operands. Thus, μ should be chosen to be sufficiently large such that u is equal to $\min(\|\mathbf{x}\|_1, \|\tilde{\mathbf{x}}_n\|_1)$. Further increasing μ has little effect on the solution aside from numerical errors that arise when the magnitude of μu is severely unbalanced with the other terms in the objective function. We observed experimentally that choosing μ to be larger than $10 \max(R_{ij})$ works well in practice since the mass transport term is bounded by $\min(\|\mathbf{x}\|_1, \|\tilde{\mathbf{x}}_n\|_1) \max(R_{ij})$.

2.3.2 EMD-DF for complex-valued signals

In some applications such as tracking in the frequency domain (e.g., DFT coefficients), the signal of interest is complex-valued. We now expand the formulation of EMD-DF for nonnegative signals from the previous section to deal with complex-valued signals. Several modifications to the traditional EMD formulation for nonnegative inputs have been proposed to allow for signed inputs [59, 60, 61]. A natural model for the purposes of EMD-DF would

be to simply ignore the signal phase and constrain flows based on the complex magnitude of each element. That is, we would like to solve

$$\begin{aligned}
\hat{\mathbf{z}}_n &= \underset{\mathbf{z}, \mathbf{F}, u}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y}_n - \mathbf{A}\mathbf{z}\|_2^2 + \lambda \|\mathbf{z}\|_1 + \gamma \sum_{ij} R_{ij} F_{ij} - \mu u \\
&\text{subject to } F_{ij} \geq 0, \\
&\sum_j F_{ij} \leq |\mathbf{z}[i]|, \quad \sum_i F_{ij} \leq |\tilde{\mathbf{z}}_n[j]|, \\
&\sum_{ij} F_{ij} = u, \quad \|\mathbf{z}\|_1 \geq u, \quad \|\tilde{\mathbf{z}}_n\|_1 \geq u.
\end{aligned} \tag{2.11}$$

Unfortunately, the second constraint causes the program to become nonconvex and thus more difficult to solve using standard tools. Here, we formulate a relaxation which is convex and easily solved via an off-the-shelf optimization package (e.g., CVX [62, 63] or templates for first-order conic solvers (TFOCS) [64]). First, we decompose both the real and imaginary parts of \mathbf{z} into positive and negative components such that

$$\mathbf{z} = (\mathbf{z}_{\text{re}}^+ - \mathbf{z}_{\text{re}}^-) + i (\mathbf{z}_{\text{im}}^+ - \mathbf{z}_{\text{im}}^-),$$

where $\mathbf{z}_{\text{re}}^+, \mathbf{z}_{\text{re}}^-, \mathbf{z}_{\text{im}}^+, \mathbf{z}_{\text{im}}^- \in \mathbb{R}_+^N$. Ideally, we would like a decomposition in which the positive and negative components do not overlap, such as

$$\mathbf{z}_{\text{re}}^+[i] \mathbf{z}_{\text{re}}^-[i] = \mathbf{z}_{\text{im}}^+[i] \mathbf{z}_{\text{im}}^-[i] = 0, \quad i = 1, \dots, N. \tag{2.12}$$

In this case, the magnitude of the real and imaginary parts may be evaluated simply by adding the corresponding positive and negative component vectors. We can then approximate the magnitude of each element as $\mathbf{z}_{\text{re}}^+[i] + \mathbf{z}_{\text{re}}^-[i] + \mathbf{z}_{\text{im}}^+[i] + \mathbf{z}_{\text{im}}^-[i]$. When the positive and negative components do not overlap, this is equivalent to replacing the magnitude (i.e., the ℓ_2 distance in the complex plane) with the ℓ_1 distance. Since $a^2 + b^2 \leq (a + b)^2 \leq 2(a^2 + b^2)$ for all $a, b \in \mathbb{R}^+$, this approximation is accurate within a factor of $\sqrt{2}$. This ideal decomposition

always exists; for example, it may be computed using the map $a^+ = \max(a, 0)$ and $a^- = -\min(a, 0)$. The resulting convex relaxation of (2.11) is then given by

$$\begin{aligned} \widehat{\mathbf{z}}_n &= \operatorname{argmin}_{\mathbf{z}', \mathbf{F}, u} \frac{1}{2} \|\mathbf{y}_n - \mathbf{A}' \mathbf{z}'\|_2^2 + \lambda \|\mathbf{z}'\|_1 + \gamma \sum_{ij} R_{ij} F_{ij} - \mu u \\ &\text{subject to } F_{ij} \geq 0, \\ &\sum_j F_{ij} \leq \mathbf{z}_{\text{re}}^+[i] + \mathbf{z}_{\text{re}}^-[i] + \mathbf{z}_{\text{im}}^+[i] + \mathbf{z}_{\text{im}}^-[i], \quad \sum_i F_{ij} \leq |\widetilde{\mathbf{z}}_n[j]|, \\ &\sum_{ij} F_{ij} = u, \quad \|\mathbf{z}'\|_1 \geq u, \quad \|\widetilde{\mathbf{z}}_n\|_1 \geq u, \end{aligned} \quad (2.13)$$

where,

$$\mathbf{A}' = \begin{bmatrix} \mathbf{A} & -\mathbf{A} & i\mathbf{A} & -i\mathbf{A} \end{bmatrix},$$

and \mathbf{z}' is the concatenation of the decomposed real and imaginary parts of \mathbf{z} . Note that the decomposition produced by this optimization is not guaranteed to satisfy (2.12). However, the ℓ_1 regularizer serves to discourage solutions containing energy in overlapping elements of the positive and negative components.

2.3.3 EMD computational complexity

For general cost distances, the optimization program (2.10) involves solving N signal variables and an additional N^2 flow variables. Thus, the addition of the EMD regularizer potentially incurs a prohibitive increase in computational complexity compared to algorithms such as BPDN or RWL1. For general distance costs, note that when only K elements of $\widetilde{\mathbf{x}}_n$ are non-zero the conditions $F_{ij} \geq 0$ and $\sum_i F_{ij} \leq \widetilde{\mathbf{x}}_n[i]$ imply that all but K columns of \mathbf{F} contain only zeros. Hence, regardless of the distance cost, we need only solve for NK flow variables for sparse signal tracking, resulting in significant savings in computational cost when $K \ll N$.

Furthermore, in the common case when the distance cost d is Euclidean, we can exploit Beckmann's formulation of the EMD (1.3) to reduce the number of EMD variables from

$\mathcal{O}(N^2)$ to $\mathcal{O}(N)$. This formulation, however, requires that the signals have unit mass (i.e., $\|\mathbf{x}\|_1 = \|\mathbf{y}\|_1 = 1$), meaning that we cannot simply apply the pre-existing method. In the following, we outline how a reformulation of the Beckmann problem for unequal total masses [41] may be incorporated into the EMD-DF program. To allow input arguments with unequal total mass, we introduce slack variables \mathbf{w}, \mathbf{v} to artificially bound the flux from the original source \mathbf{x} and sink \mathbf{y} . The modified EMD program is then:

$$\begin{aligned}
d_{\text{emd}}(\mathbf{x}, \mathbf{y}) &= \min_{\mathbf{M}, \mathbf{w}, \mathbf{v}} \|\mathbf{M}\|_{2,1} \\
\text{subject to } & \text{div}(\mathbf{M}) + \mathbf{v} - \mathbf{w} = \mathbf{0}, \\
& \mathbf{0} \leq \mathbf{w} \leq \mathbf{x}, \mathbf{0} \leq \mathbf{v} \leq \mathbf{y}, \\
& \|\mathbf{w}\|_1 = \|\mathbf{v}\|_1 = \min(\|\mathbf{x}\|_1, \|\mathbf{y}\|_1), \tag{2.14}
\end{aligned}$$

where \mathbf{w}, \mathbf{v} are nonnegative vectors with similar dimensions as \mathbf{x}, \mathbf{y} . This optimization searches for the minimal vector field configuration that describes, via the first constraint, its flux to be travelling between a source \mathbf{w} and a sink \mathbf{v} . The second constraint describes the source and sink as nonnegative slack variables that are bounded above by their proxies \mathbf{x} and \mathbf{y} respectively; this constraint is analogous to the mass preservation constraints in (1.1). The last constraint states that the induced flux must be bounded by the total mass of the smaller operand signal, which is similar spirit in to the fourth constraint of (1.1). This formulation has $N(D + 2)$ variables, where D is the dimensions of the vector field (e.g., $D = 2$ for images). Applying this EMD formulation, (2.10) becomes

$$\begin{aligned}
\hat{\mathbf{x}}_n &= \underset{\mathbf{x}, \mathbf{M}, u, \mathbf{v}, \tilde{\mathbf{v}}}{\text{argmin}} \frac{1}{2} \|\mathbf{y}_n - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1 + \gamma \|\mathbf{M}\|_{2,1} - \mu u \\
\text{subject to } & \text{div}(\mathbf{M}) + \tilde{\mathbf{v}} - \mathbf{v} = \mathbf{0}, \\
& \mathbf{0} \leq \mathbf{v} \leq \mathbf{x}, \mathbf{0} \leq \tilde{\mathbf{v}} \leq \tilde{\mathbf{x}}, \\
& \|\mathbf{v}\|_1 = \|\tilde{\mathbf{v}}\|_1 = u, \quad u \leq \|\mathbf{x}\|_1, \quad u \leq \|\tilde{\mathbf{x}}\|_1, \tag{2.15}
\end{aligned}$$

where we have introduced another slack variable u to linearize the minimum operator in (2.14) to make the program convex. The complex variant of EMD-DF given by (2.13) can also be trivially converted to adopt this formulation, though it is not shown here for the sake of brevity. As we eluded in section 1.2, (2.15) enjoys the reduction in computational complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$ while preserving the benefits of partial optimal transport (OT) (in contrast to the traditional balanced OT Beckmann formulation) and avoiding the approximation error associated with methods such as Sinkhorn iterations.

2.4 Results

In this section, we demonstrate the utility and performance of EMD-DF through a series of simulations on synthetic and real data. First, we consider a stylized example where the goal is to track sparse targets moving throughout the state space. We then consider a more physically inspired example consisting of a 2-D network of Kuaramoto oscillators. Next, we study the problem of tracking time varying frequencies in a 1-D time series then use the same approach to track neural oscillations in electrophysiology data. Finally, we demonstrate the significant numerical speed up of EMD-DF due to Beckmann’s formulation.

Throughout these experiments, we use the TFOCS [64] software package to solve the optimization problems for BPDN, BPDN-DF, RWL1, and RWL1-DF and the variant of EMD-DF for nonnegative signals. For the complex-valued variant of EMD-DF we use the CVX software package [62, 63].

2.4.1 Target tracking

This set of simulations is motivated by applications such as magnetic resonance imaging angiography [65] and microscopy artifact removal [66] which leverage a sparsity model on a state vector with two spatial dimensions. Each 32×32 state vector contains a handful of positive target coefficients and the remaining elements are equal to zero. Rather than directly observing the state, we observe noisy linear measurements through a Gaussian

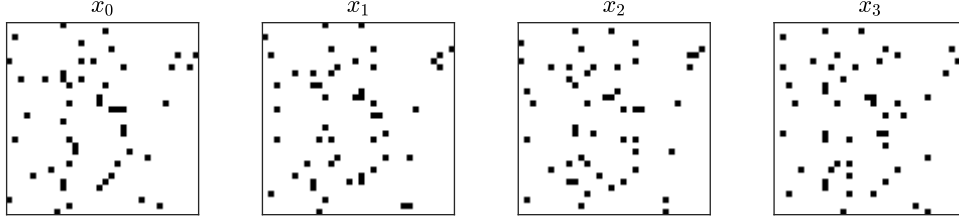


Figure 2.2: Example state sequence used in target tracking simulations. Each frame is a 32×32 scene containing $K = 0.05N$ targets which move randomly to adjacent locations with equal probability. The goal of these simulations is to recover the state from Gaussian compressive measurements.

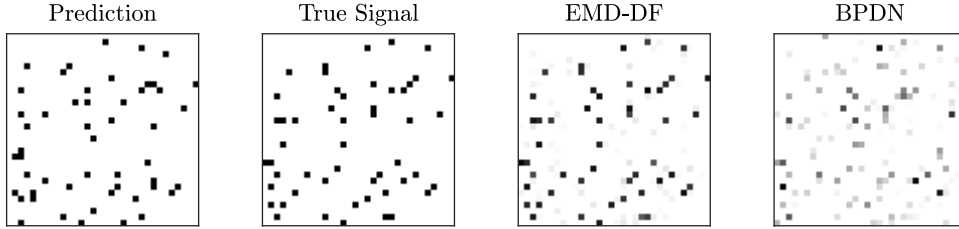


Figure 2.3: Example state recovery. The prediction is formed using an identity dynamics model: $\tilde{\mathbf{x}}_n = \hat{\mathbf{x}}_{n-1}$. EMD-DF is employed to recover the true signal \mathbf{x}_n using the prediction and a set of noisy compressive measurements $\mathbf{y}_n = \mathbf{A}\mathbf{x} + \sigma\epsilon_n$. Despite the slight inaccuracies in the locations of the predicted coefficients, the EMD regularizer enables nearly perfect recovery, whereas BPDN misses the majority of the targets.

observation operator (i.e., compressive sensing measurements). At each time step, targets move randomly to adjacent locations via discrete Brownian motion. An example trajectory is shown in Figure 2.2.

An example recovery for a single time point is shown in Figure 2.3. Because no information about the direction of object movement is available, the predictions for each algorithm are formed using an identity dynamics model (i.e., $\tilde{\mathbf{x}}_n = \hat{\mathbf{x}}_{n-1}$). The EMD dynamics regularizer shows a clear qualitative benefit, even though the precise locations of the prediction do not align with those in the ground truth.

Next, we evaluate algorithm performance by quantitatively computing the relative mean-square-error (rMSE) defined by

$$\text{rMSE} = \frac{\|\mathbf{x}_n - \hat{\mathbf{x}}_n\|_2^2}{\|\mathbf{x}_n\|_2^2}. \quad (2.16)$$

As before, an identity dynamics model is used for prediction in each of the tracking algorithms. Figure 2.4 shows the rMSE for the same simulated state sequences recovered using EMD-DF and a host of other sparse recovery algorithms. Note that EMD-DF maintains the lowest rMSE for the entire segment and the two competing tracking algorithms actually perform worse than BPDN (which does not account for the dynamics model at all). When $\gamma = 0$, BPDN-DF reduces to BPDN, however we use a modest positive value for γ to demonstrate how the dynamics mislead recovery when using the ℓ_p norm as a regularizer. The superior performance of EMD-DF reflects its ability to effectively utilize predictions even if they drift slightly from the true value. Next, the plots in Figure 2.5 demonstrate the mean performance of the various algorithms as functions of the sparsity level, K , and as a function of the number of measurements taken, M . Compared to the competing algorithms, EMD-DF is able to successfully track more targets for a given number of measurements, or successfully track a given number of targets using fewer measurements.

The error metric (rMSE) used in the simulations so far is based on the ℓ_2 -norm which also appears in the EMD-DF objective function. To demonstrate that EMD-DF improves performance under other metrics as well, we evaluate support recovery performance by considering a detection task in which an element of the recovered state vector is marked as a target if its value exceeds a fixed threshold. The evaluation metric used is the F_1 score, defined as

$$F_1 = \frac{2TP}{2TP + FN + FP},$$

where TP , FN , and FP denote the number of true positives, false negatives, and false positives respectively. The F_1 score takes values in $[0, 1]$, where a score of one corresponds to perfect detection (i.e., all targets are detected without producing false alarms). Figure 2.6 shows F_1 score as a function of M , the number of measurements taken. EMD-DF allows for more accurate recovery of the support using fewer measurements than the other algorithms shown.

Finally, Figure 2.7 shows rMSE as a function of target movement speed. For low-

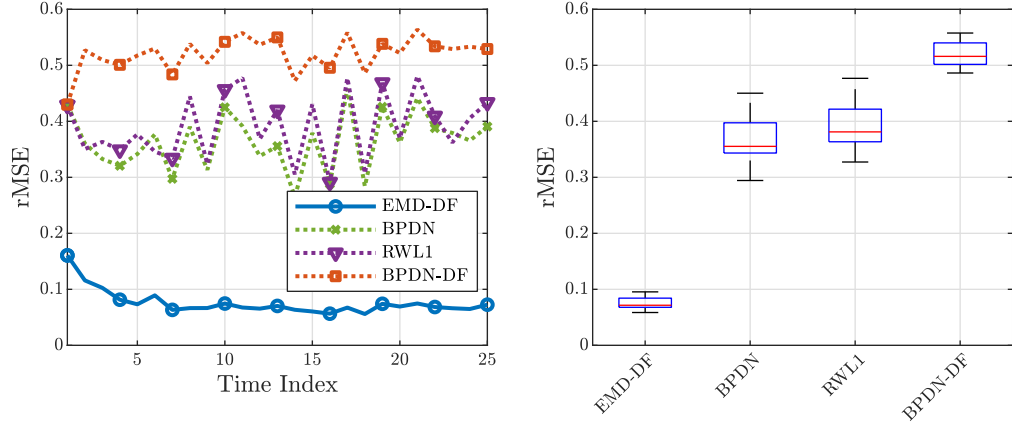


Figure 2.4: Recovery performance over time. Left: the rMSE for estimates produced by various recovery algorithms (averaged over 20 trials) is plotted as a function of time. The state size is 32×32 , measurement vectors have length $M = 0.2N$, and each frame contains $K = 0.05N$ targets. EMD-DF is the top performer throughout. An identity dynamics function is used in all of the tracking algorithms, however only EMD-DF is able to effectively use the information from the predictions. In fact, the inappropriate tracking regularizer in BPDN-DF and RWL1-DF (omitted here due to disproportionately large recovery error) actually degrades performance. Right median recovery rMSE. For each algorithm, the red line indicates the median rMSE with respect to time. Box boundaries indicate the 25% and 75% percentiles and whiskers indicate minimum and maximum values (excluding outliers indicated by red crosses). The rMSE of EMD-DF at each time point is tightly clustered around its median which is lower than competing algorithms.

speed targets, a static model with a small blurring kernel in the prediction may work well in conjunction with a traditional regularization approach using an ℓ_p norm regularizer. However, this approach requires another parameter to be continually adapted to the current target speed and wider blurring kernels result in more information loss in the prediction. To illustrate these effects, we simulate target tracking over different speeds while incorporating a 3×3 averaging filter into the dynamics prediction for RWL1-DF while keeping an identity dynamics model for the other algorithms. We see in this plot that target speed does affect the overall performance of all methods tested, but EMD-DF in general demonstrates more robustness to variations in target speed.

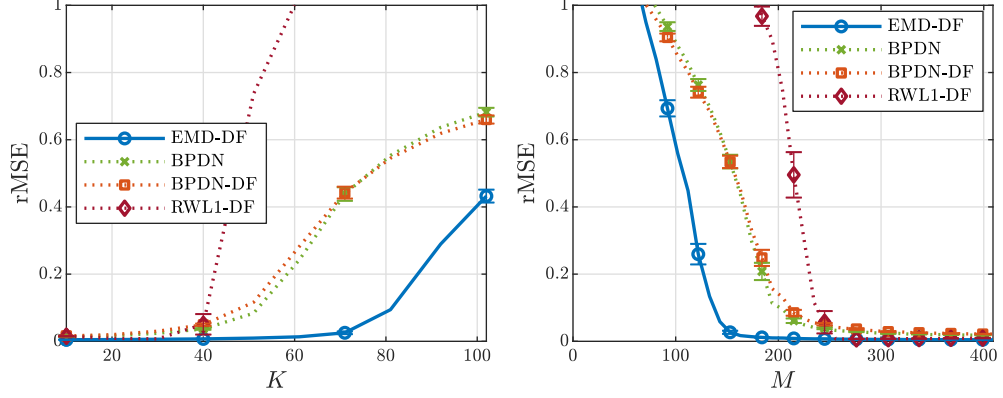


Figure 2.5: State recovery performance scaling behavior. In each of the simulations, states are 32×32 in size and the variance of the noise is $\sigma^2 = 0.001$. The left plot shows rMSE as a function of K , the number of targets in each state vector, using $M = 0.2N$ compressive measurements. On the right is a plot of rMSE as a function of M , the number of compressive measurements for $K = 0.05N$ targets. Note that EMD-DF produces the lowest rMSE throughout the range of M and significantly outperforms competing algorithms in the measurement constrained regime where the inference problem is particularly difficult. EMD-DF produces estimates with lower rMSE for a greater number of targets, or using a fewer number of measurements. Consistent with previous plots, BPDN-DF and RWL1-DF ineffectively utilize the predictions, resulting in worse performance than BPDN.

2.4.2 Tracking traveling waves

Traveling waves are another form of neural oscillation pattern of interest in the neuroscience community. For example, wave propagation has been shown to correlate to events and performance in tasks involving neurosurgical patients [67]. We generate synthetic traveling wave data using the phase-coupled Kuramoto oscillator model which has been used to study the properties of traveling waves in neuronal activity [68, 69]. The Kuramoto model describes the instantaneous phase of each node in an array of linked oscillators via a system of differential equations. Motivated by the work in [68], we simulate a 40×40 oscillator array where the instantaneous phase of oscillator (i, j) with intrinsic frequency ω_{ij} depends on its four nearest neighbors via the equation $\theta'_{ij} = \omega_{ij} + 300 \sum_{k=1}^4 \sin(\theta_k)$. We choose $\omega_{ij} = 2 + 0.103i + 0.359j$ to produce a linear frequency gradient which has been observed to yield traveling wave solutions. After solving for the θ_{ij} , we threshold the oscillator voltage $\sin(\theta_{ij}(t))$ to extract the wavefront $x_{ij}(t)$. The goal of these simulations is to recover

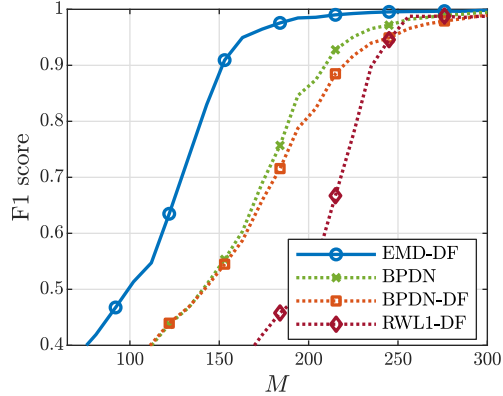


Figure 2.6: Performance in a detection task as a function of the number of compressive measurements, M . The F_1 score takes values in the interval $[0, 1]$ (higher is better) and is an aggregate score which balances the precision and recall. In this experiment, we compute the F_1 score for support recovery of the state vector across 20 trials, each consisting of 25 samples in time. State vectors are of size $N = 32 \times 32$ and contain $K = 0.05N$ targets. EMD-DF enables accurate support recovery using fewer measurements compared to competing algorithms.

the wavefront from noisy linear measurements $\mathbf{y}(t) = \Phi \mathbf{x}(t) + \epsilon$. We measure upper bound tracking performance by providing the ground truth previous frame as the prediction. Figure 2.8 shows how EMD-DF enables nearly perfect recovery compared to BPDN and BPDN-DF which are unable to effectively use the prediction.

2.4.3 Frequency tracking

In the next series of simulations, we study the performance of EMD-DF for complex-valued signals via a frequency tracking task. In particular, we observe noisy measurements of a 1-D time series which is composed of K sinusoids of different frequencies that change as a function of time:

$$y(t) = \sum_{k=1}^K a_k \cos(2\pi f_k(t) + \phi_k(t)) + \sigma \epsilon(t). \quad (2.17)$$

where the a_k are chosen by dividing the unit interval according to a uniform distribution, the phases $\phi_k(t)$ are chosen deterministically based on the $f_k(t)$ to ensure that $y(t)$ is continuous, and the last term represents additive Gaussian noise with variance σ^2 . The frequencies $f_k(t)$ change every c_t samples by an amount c_f , where c_t is an adjustable

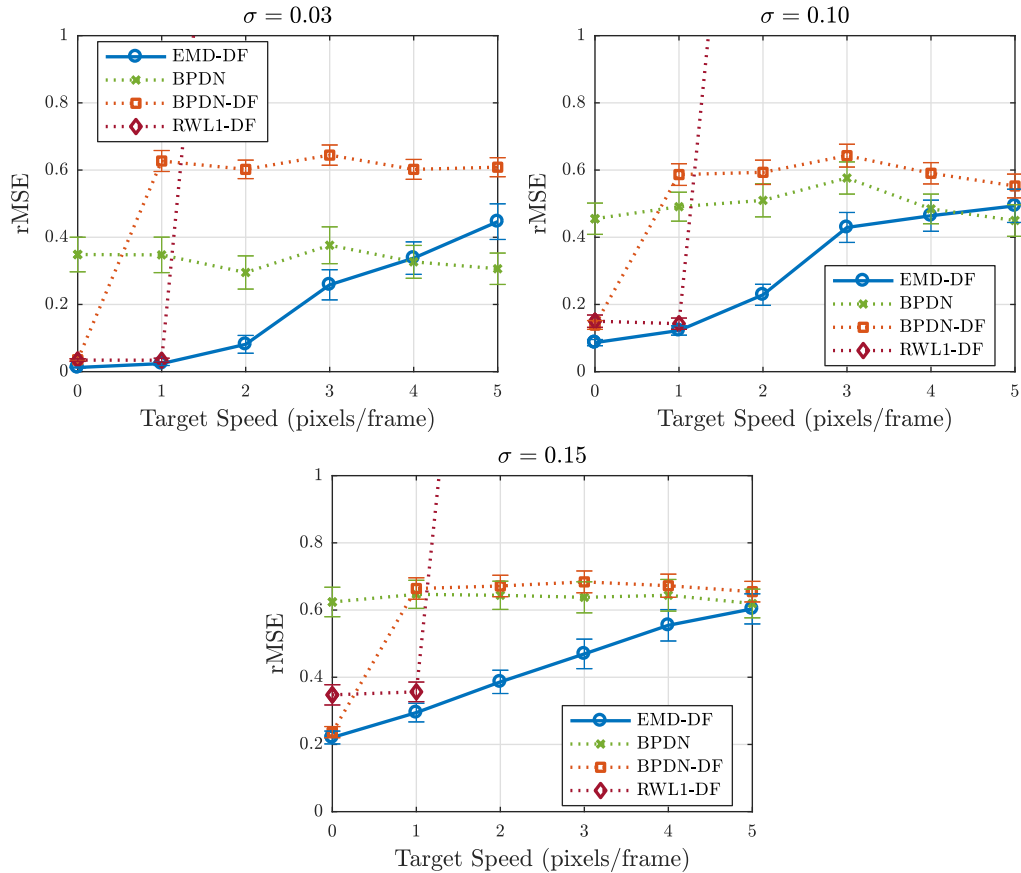


Figure 2.7: Algorithm performance as a function of target speed in target tracking simulations for different levels of noise variance, σ^2 . State vectors are of size 20×20 and contain $K = 5$ active elements. Performance of BPDN-DF plummets when there is any support mismatch in the prediction due to its ℓ_p -norm dynamics regularization term. By blurring the prediction, RWL1-DF is able to cope with small support mismatch between the prediction and the true signal. However, the addition of a blurring kernel introduces another parameter which may not be feasible to tune. Furthermore, blurring the prediction causes RWL1-DF to perform worse than BPDN-DF at high noise levels when there is no support mismatch. By contrast, EMD-DF handles more severe support mismatch with no additional parameters and its performance scales better as a function of noise level.

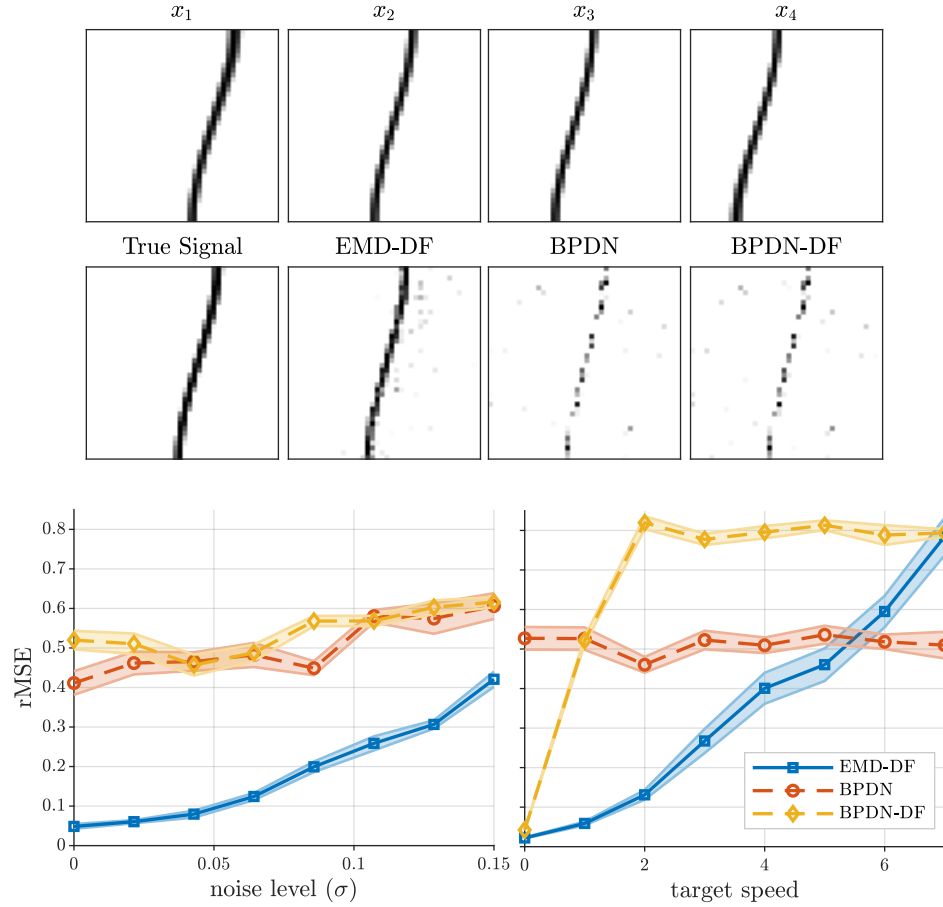


Figure 2.8: Single step recovery of wavefronts from a 40×40 Kuramoto oscillator array via linear Gaussian measurements. (Top) A linear gradient in the oscillators’ intrinsic frequencies results in traveling waves across the array. (Middle) Examples of wavefronts recovered using several methods. BPDN produces poor recovery performance due to undersampled ($M/N = 0.15$) and noisy ($\sigma = 0.08$) measurements. Dynamics regularization in BPDN-DF is of little help due to its inability to effectively utilize the prediction which has little support overlap with the ground truth signal. EMD regularization is robust to the mismatch in pixel location between the ground truth and prediction and thus enables successful recovery of the wavefront. (Bottom row) Recovery performance averaged over 10 trials. Error bars indicate $\alpha = 0.01$ confidence intervals. (Bottom-left) EMD-DF produces superior performance for various values of the noise standard deviation σ . (Bottom-right) The ℓ_2 -norm dynamics regularization in BPDN-DF actually degrades performance compared to BPDN for moving targets. In contrast, EMD-DF is substantially more robust to support location mismatch caused by target movement.

parameter and $c_f \sim \mathcal{N}(0, \sigma_f)$. New values of c_t and c_f are drawn after each frequency change. Frequencies are constrained to reside in a specified band; if c_f is generated such that $f_k(t) + c_f$ is outside of the specified band, c_f is regenerated until a permissible value is produced. In the simulations that follow, data are generated with a sampling frequency of $f_s = 256$ Hz using $K = 3$, $\mu_t = 40$, $\sigma_t = 0$, $\sigma_f = 4$ and each frequency is banded between 0 and 128 Hz.

Our goal is to recover denoised time-frequency plots with greater time and/or frequency resolution than is possible with standard short-time Fourier transform based methods. This can be accomplished within the sparse signal tracking framework by estimating sparse coefficients in an overcomplete DFT dictionary Φ , where

$$\Phi_{mn} = \exp(i2\pi mn/N), \quad (2.18)$$

for $m = 0, \dots, M - 1$ and $n = 0, \dots, N - 1$. Each element of the complex-valued state vector \mathbf{x} represents a frequency component in the dictionary Φ . In this context, the values of M and N have a different interpretation than our previous experiments. The parameter M controls the length of the analysis window. Larger values of M provide lower noise and higher frequency resolution estimates at the expense of lower temporal resolution. The parameter N controls the number of overcomplete DFT coefficients. Larger values of N result in better frequency resolution in the dictionary, but a more challenging inference problem. We call the ratio $S = N/M$ the oversampling factor. The dynamics model used in these experiments is a simple denoising function $g(\mathbf{x}) = \tau_q(\mathbf{x})$ which sets all but the largest q elements of \mathbf{x} equal to zero. The value of q is a parameter that controls the number of frequencies to track.

In the following experiments, we compare the performance of EMD-DF to other sparse recovery algorithms. We note that spectrogram reassignment is an alternative method for sharpening time-frequency (TF) representations beyond what is possible with the standard

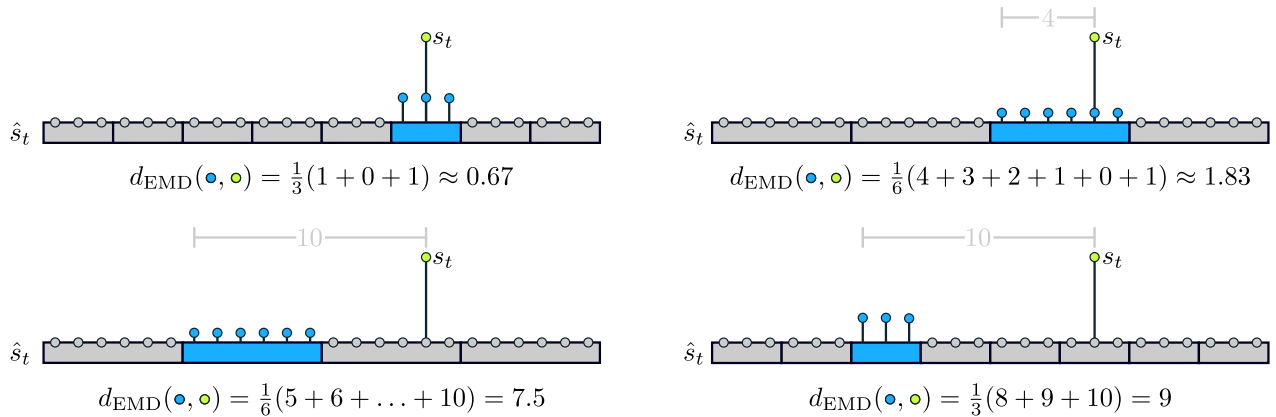


Figure 2.9: Examples of error computation used in frequency tracking simulations. Spectral estimates \hat{s}_t are shown for a single time slice, and the ground truth at time t is indicated by s_t . When the ground truth frequency (marked in green) is contained in the active bin in the spectral estimate (marked in blue), higher resolution estimates (upper-left) are favored over lower resolution estimates (upper-right). When the active bin does not contain the ground truth, estimates with center-of-mass closer to the ground truth are favored, regardless of the resolution of the estimate (bottom-left vs bottom-right).

STFT [70, 71, 72, 73, 74]. However, reassignment methods involve a batch procedure which reassigns energy in the spectrogram using a signal dependent transformation of the time-frequency plane. In contrast, EMD-DF is causal, a feature that is critical in online applications such as closed-loop control. Therefore, we do not provide a comparison to reassignment methods in these simulations.

The goal in these simulations is to produce a super-resolution TF representation which best describes the frequency content defined by the $f_k(t)$. However, recovery error, the error metric used in earlier simulations, is not an appropriate measure of progress toward this goal. For instance, imagine the problem of recovering the frequency content from noiseless measurements. Perfect reconstruction could be achieved by estimating frequencies using the STFT, however the resulting time and frequency resolution may be quite poor. Given this shortcoming, we will use an alternative error metric, illustrated in Figure 2.9, which promotes solutions with higher resolution and allows for comparison of representations of differing resolutions. Each time slice of the spectrum estimate is upsampled to a common frequency grid with resolution exceeding any of the spectra under consideration. For each

sample in time, the EMD is computed between the upsampled estimate and the ground truth. These distances are then summed for each time sample to form an aggregate error. We remark that although the EMD is used as a regularizer in EMD-DF as well as in the proposed error metric, it is used in a very different manner in each case. In the error metric, the EMD is computed between the estimate and the ground truth, while in EMD-DF the EMD is used to compare candidate estimates to the prediction. Because of this distinction, EMD-DF does not receive an unfair advantage under this error metric.

Figure 2.10 shows recovery error as a function of measurement noise. Algorithm parameters (e.g. λ , γ , etc.) are tuned for each noise value via direct search [75]. We find that using parameters found via this method yield performance that matches or exceeds those found by manual tuning, a common practice in the evaluation of sparse recovery algorithms. The dynamics function used in EMD-DF and BPDN-DF is $g(\mathbf{x}) = \tau_q(\mathbf{x})$ (i.e., set all but the largest q elements equal to zero). The dynamics function used in RWL1-DF additionally blurs the estimate to approximate a local frequency preference in the inference as this resulted in improved performance. At low noise levels, the measurements are reliable enough that high accuracy recovery is possible without dynamics information, so all of the algorithms perform well. At exceedingly high noise levels, the predictions given by the dynamics model yield no additional information. In the middle region however, dynamics significantly aid recovery.

2.4.4 Tracking neural oscillations

In this section, we apply EMD-DF to the problem of spectrum estimation from neurophysiology recordings. Oscillatory behavior is prominent in neural recordings in a variety of settings and is thought to be a fundamental phenomenon in brain function. There is great interest in the neuroscience community to understand the functional role of these oscillations [76, 77].

In many studies, the tools used for spectral analysis of neural recordings are based on

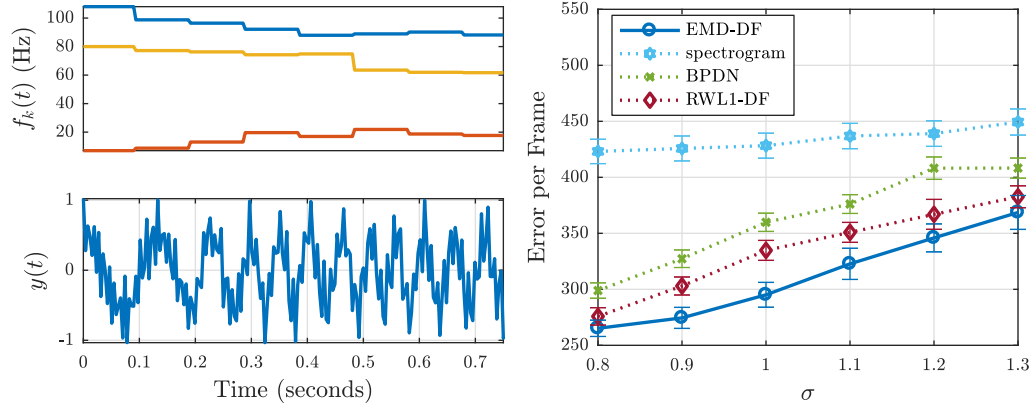


Figure 2.10: Mean spectral estimate error. Left: example of component frequencies and the resulting (noisy) time series data. Right: Mean spectral estimate error as a function of noise standard deviation σ for frequency tracking simulations. Observed signals consist of three frequencies which change randomly every 150 ms according to Brownian motion with standard deviation equal to 4Hz. The mean error across 1000 trials is shown. Error bars represent $\alpha = 0.01$ confidence intervals. For moderate noise levels, EMD-DF outperforms BPDN and RWL1-DF.

the classical STFT. The time and frequency resolution of such techniques is thus limited by the uncertainty principle which prevents simultaneously achieving high frequency and time resolution. Here, we study how higher TF resolution may be obtained by imposing a sparsity model on the data and using EMD-DF for recovery in an overcomplete DFT dictionary.

In particular, we study the phenomenon of oscillation phase coupling in the theta (4-7 Hz) and gamma (30-80 Hz) bands which is observed in tasks such as memory consolidation and learning of item-context associations [78, 79, 80, 81, 82]. We begin by generating synthetic data so that we have a ground truth against which to compare various sparse recovery algorithms. We generate data which consists of two components: a theta band frequency and a gamma band frequency which is modulated by that same theta frequency. More precisely, our simulated data are defined by

$$y(t) = a_\theta \cos(2\pi f_\theta(t)t + \phi_\theta(t)) [1 + a_\gamma \cos(2\pi f_\gamma(t)t + \phi_\gamma(t))] + \varepsilon(t), \quad (2.19)$$

where $f_\theta(t)$ and $f_\gamma(t)$ are theta and gamma band frequencies respectively that drift according

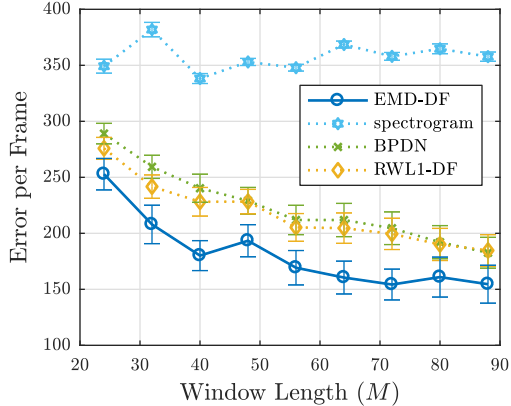


Figure 2.11: Mean frequency recovery error as a function of window length for simulated neural oscillation signals. Data is generated with a sampling frequency of 256 Hz and consists of a theta band (4-7 Hz) component and a gamma band (30 - 80 Hz) component which is modulated by the amplitude of the theta band activity. Frequencies of the theta and gamma band components change randomly according to Brownian motion every 150 ms with standard deviations of 0.5 Hz and 6 Hz respectively. Shown above is the mean error averaged over 300 trials. Error bars represent $\alpha = 0.01$ confidence intervals. All of the sparse recovery algorithms offer vastly improved performance compared to the STFT based spectrogram. EMD-DF is the top performer, especially for shorter window lengths where the inference problem is particularly difficult.

to Brownian motion, a_θ and a_γ are their respective amplitudes, and $\varepsilon(t)$ is Gaussian noise. The phases $\phi_\theta(t)$ and $\phi_\gamma(t)$ change at frequency change points to prevent discontinuities. In the simulations below, we choose $a_\theta = 1$ and $a_\gamma = 0.2$.

We use EMD-DF in the same way described in 2.4.3. In this setting, the power in the theta band is much greater than that in the gamma band, so recovery of the theta band component is trivial. Thus, we modify the error metric by masking out the theta band frequency to concentrate on the recovery of frequencies in the more challenging band.

First, we consider how performance scales as a function of window length, which directly determines time resolution. Recovery error is plotted as a function of window length in Figure 2.11. Compared to competing methods, EMD-DF produces estimates with lower error, especially when shorter window lengths are used. In this experiment, we keep the oversampling factor constant ($S = 5$), so using a longer window length results in higher frequency resolution. Furthermore, including more observations in our analysis

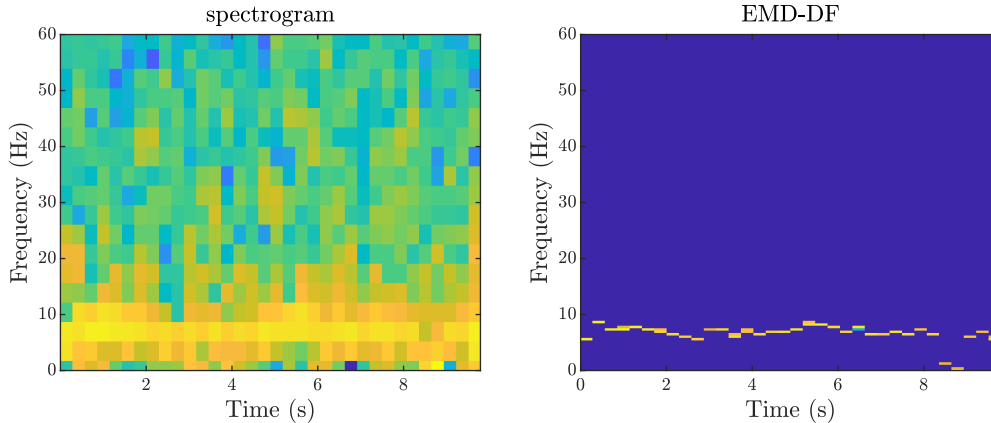


Figure 2.12: Time-frequency plots for a single channel of tetrode data recorded from the rat hippocampus. Data is sampled at 250 Hz and an analysis window length of 72 samples is used for both plots. The spectrogram (left), which is produced using the traditional STFT with a hamming window, yields lower frequency resolution and severe leakage in the lower frequencies. The TF plot on the right is produced by EMD-DF with an 5x overcomplete DFT matrix, resulting in high enough frequency resolution to smoothly track subtle changes in frequency.

window results in lower noise estimates. Both of these factors outweigh the loss in temporal resolution, and error thus decreases as a function of window length.

Finally, we employ EMD-DF to estimate the spectrum in a segment of real electrophysiology data recorded from a tetrode in rat hippocampus [83]. We set the dynamics function to track the top two frequencies ($g(\mathbf{x}) = \tau_2(\mathbf{x})$), and use an oversampling factor of $S = 5$. Figure 2.12 shows TF plots produced by the spectrogram and EMD-DF. Because EMD-DF utilizes the overcomplete DFT matrix for recovery, it produces a TF plot with vastly improved frequency resolution. Additionally, the spectrogram suffers from severe leakage in the theta band frequencies, an artifact which is not present in the sparse TF representation. Finally, we note that the improved resolution of the sparse TF plot reveals more subtle oscillatory dynamics that cannot be observed in the spectrogram.

2.4.5 Computational scalability

Given the increases in performance and robustness demonstrated by EMD-DF we are especially interested in improving computational complexity so that the algorithm can still

scale well in practical applications with large state spaces. Here we examine the impact of adapting an approach based on Beckmann’s EMD formulation into our tracking problem. We note that frequency tracking (from the previous sections) also benefits from this formulation because it can be treated as an image tracking problem.

We conducted a similar simulation detailed in section 2.4.1 and scaled the problem between state sizes of 12×12 ($N = 144$) and 48×48 ($N = 2304$), where N is the total number of state elements. For each state size, the sparsity level was fixed at 5%. Each experiment was repeated 10 times for statistics aggregation and error bars denote ± 1 standard deviation from the mean. Because our major concern is whether the proposed computational modification degrades EMD-DF performance over the general (but expensive) formulation, we compare differences *between* solutions using the root mean-square-error: $\sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2}$, where x_i and y_i are the values of state elements in the respective solutions. We use the CVX software package (employing interior point methods) for both formulations for a fair comparison and measure relative runtime on a personal computer (Intel Core i7 with 3.5 GHz processor speed).

Figure 2.13 shows that by using the Beckmann’s formulation of EMD-DF (2.15), we obtain a significant speed up over the general formulation of EMD-DF (2.10). Figure 2.13 also shows that the difference between the Beckmann’s formulation of EMD-DF and the general formulation have very small differences. Taken together, these results demonstrate that the proposed re-formulation is much more computationally tractable and scalable to larger problem sizes while producing solutions that are essentially the same as the general approach.

2.5 Summary and future work

The estimation of signals that traverse a gridded domain can be enhanced by regularizing for underlying sparsity and dynamical structure. While current tracking methods in the literature have investigated sparsity in a number of ways, the issue of dynamical support mismatch

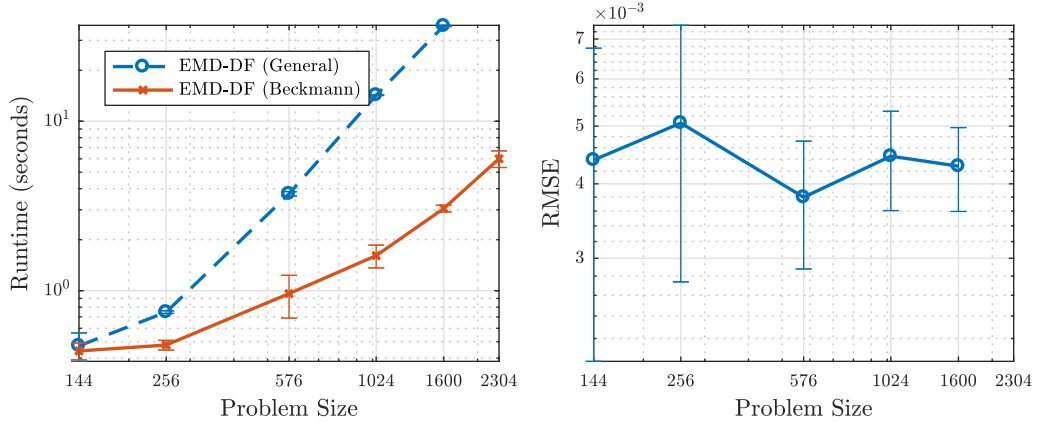


Figure 2.13: Demonstration of computational speed up with comparable solutions. We compare the runtime and the difference in solutions for two formulations of EMD-DF EMD-DF (General) (2.10) which adopts generic distance costs, and EMD-DF (Beckmann) (2.15) which assumes Euclidean distance costs. The left plot demonstrates that EMD-DF (Beckmann) significantly outperforms EMD-DF (General) in runtime, and in the right plot, the difference in solutions were shown to be negligible.

remains a challenging open problem. To address this, we apply the EMD as a tracking regularizer for time varying sparse inverse problems in our proposed EMD-DF algorithm. The EMD provides a natural geometric framework that specifically computes the amount of support mismatch between two signals measured over a fixed-grid. However, since the EMD is itself an optimization problem, this requires a reformulation such that it fits into a natural setup for sparse inverse problems. In this work, we introduce two convex algorithms for tracking nonnegative signals and complex-valued signals (as a convex relaxation) and show empirically that both variants outperform competing sparse recovery algorithms. In the context of frequency tracking, we show how EMD-DF can be used to produce time-frequency representations with resolution in both time and frequency that exceed what is possible with traditional methods like the STFT. In contrast to other approaches for increasing the readability of time-frequency plots such as spectrogram reassignment, EMD-DF is a causal algorithm making it applicable for online systems (e.g., for perturbation experiments or closed-loop control).

Computations using the traditional formulation of the EMD involve $\mathcal{O}(N^2)$ flow vari-

ables. In the context of real time tracking using the EMD as a regularizer, this computational burden can be prohibitive. When transport distances are Euclidean, we show that a reformulation of EMD yields an extremely efficient method that reduces the problem to require only $\mathcal{O}(N)$ optimization variables. This recasting of the problem of interest into a more efficient optimization program dramatically reduces computational complexity to allow EMD-DF to be run efficiently for non-trivial problem sizes.

The EMD calculation may remain prohibitive for extremely large problems or for more general cases that do not use Euclidean distances. Fortunately, recent work in the OT literature studies methods for more efficient computation of the EMD using a variety of relaxation techniques. Future work will focus on algorithmic advances [9, 10] to incorporate these techniques into the problem of sparse signal tracking.

Acknowledgement

We would like to thank C. Kemere for the tetrode recording data.

CHAPTER 3

APPLICATIONS IN OT-REGULARIZED RPCA

The primary objective of this chapter is to explore the utility of optimal transport regularization in infrared target tracking applications¹. Accurate detection of small dim targets in infrared imagery is a crucial component in infrared search and track which has broad utility in military and remote sensing applications. In this chapter, we extend the signal model used in the previous chapter to signals which can be decomposed into sparse and low-rank components. Low-rank models have enjoyed state-of-the-art performance in infrared tracking applications, but many approaches underutilize dynamics information which has the potential to improve performance in challenging tracking scenarios. We present two algorithms, robust principal components analysis with patched unbalanced optimal transport (RPCA+PUOT) and robust alignment by sparse and low-rank with patched unbalanced optimal transport (RASL+PUOT), which incorporate optimal transport dynamics regularization and demonstrate improved performance on realistic data. Additional related work including simulations involving natural video sequences and a proof of convergence for the UOT proximal algorithm is discussed in section 3.2.

¹Key aspects of this work were presented in [84, 85]. This work was completed in collaboration with Dr. John Lee, Dr. Keith Prussing, and Dr. Samuel Shapero. JL developed the unbalanced optimal transport formulation, the corresponding proximal algorithm, and the initial UOT regularized RPCA solver. These results were the focus of [84] for which JL was the primary author. NPB was the second author of [84], contributing a proof of convergence for the proposed proximal algorithm, a variant of the robust PCA with UOT-regularized dynamic filtering (RPCA+UOT-DF) algorithm which allows for general real-valued signals instead of strictly nonnegative ones, and simulation results for real natural video sequences. NPB independently proposed the use of the EMD as a regularizer in RPCA for infrared tracking applications. NPB was the primary author on [85] which centered around the infrared tracking problem and presented solvers for two variants of UOT-regularized RPCA (one tailored for the infrared tracking model, and one which additionally incorporates an affine transformation of the data). KP and SS provided simulated infrared video data as well as valuable feedback related to the specifics of the infrared tracking problem.

3.1 Infrared search and track with optimal transport regularization

3.1.1 Introduction

Infrared search and track (IRST) involves the identification of targets of interest from a sequence of infrared images and is crucial in military, surveillance, and remote sensing applications. A crucial component of IRST is the separation and denoising of pixels containing targets from those containing extraneous objects in the scene called clutter which may include clouds, reflections, foliage, breaking waves, and variations due to neighboring materials. Often the distance between the targets and imaging sensor is quite far, resulting in images where the target occupies only a few pixels which are dim due to optical scattering. These effects combine to yield in images with low signal-to-noise and clutter ratios. The resulting small and dim targets lack well-defined edges or easily identifiable textures which makes the application of traditional visual tracking methods which depend on these characteristics challenging.

A large body of work on the infrared target detection problem focuses on first removing background clutter and noise via spatial filtering, and then performing tracking on the resulting image. Two popular approaches for background removal are the max-mean and max-median filters [86], and morphological filters like the top-hat filter and its modifications [87, 88, 89]. A host of other methods have also been explored including wavelet-based methods [90], modifications to the least mean squares filter [91], partial differential equation based methods [92], principal components analysis [93], and the kernelized correlation filter [94]. Another class of work focuses on exploiting particular characteristics of the scene for background removal. For example, the authors of [95] use a decision tree based on characteristics of the scene such as the level of clutter and the presence of a horizon line. Similarly, [96] proposed a method which segments the image into sky, horizon, and water components and uses specialized processing methods for each area of the image. Tracking methods are typically based on modifications to standard techniques like Kalman or particle

filtering [97, 98].

Recent approaches leverage advances in recovery techniques for low-rank plus sparse models. Early work in this direction introduced the infrared patch image (IPI) model which decomposes each frame into overlapping patches and assumes a low-rank plus sparse model which is inspired by the observation that background clutter tends to have a self-similarity structure, while targets are small [99]. One potential drawback of the original IPI work is that the recovery algorithm processes each frame in isolation and does not consider temporal correlation between frames which may be crucial in especially challenging tracking problems. Other related approaches include the use of an overcomplete dictionary for the background and target [97], weighted nuclear norm minimization to aid low-rank recovery [100], total variation norm regularization [101] and the use of a weighted correlation filter for detection on a low-rank decomposition [102]. The authors of [103] enhanced the IPI model to include multiple time slices in an attempt to capture temporal dynamics in the model. A sparse Bayesian learning approach was employed to enforce the low-rank structure, and a Markov random field (MRF) was used to encourage local similarity in the noise and target statistics. However, in situations where the target is small, the MRF model which encourages neighboring pixels to have similar statistics may be inappropriate.

In this chapter, we propose the use of unbalanced optimal transport (UOT) as a regularizer in low-rank plus sparse recovery algorithms for infrared target tracking. We introduce the robust principal components analysis with patched unbalanced optimal transport (RPCA+PUOT) algorithm for separation of targets from the background in the IPI model, and robust alignment by sparse and low-rank with patched unbalanced optimal transport (RASL+PUOT) which further incorporates an affine transformation into the model to allow for movement of the scene (e.g., due to an unstabilized imaging sensor). Through simulations on realistic simulated data, we show that UOT can substantially improve tracking performance.

3.1.2 Background

Robust principal components analysis for IRST

robust principal components analysis (RPCA) aims to remedy one of the fundamental weaknesses of traditional principal components analysis — extreme sensitivity to outliers. Rather than attempting to represent data as the linear combination of a small number of principal components, RPCA further allows for sparse and potentially large errors. In particular, RPCA models an input data matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$ as

$$\mathbf{Y} = \mathbf{B} + \mathbf{T}, \quad (3.1)$$

where \mathbf{B} is low-rank and \mathbf{T} is sparse. It is shown in [104] that under certain conditions on the matrices \mathbf{B} and \mathbf{T} , they may be recovered exactly from \mathbf{Y} via Principal Component Pursuit which solves the following:

$$\begin{aligned} & \min_{\mathbf{B}, \mathbf{T}} \lambda \|\mathbf{T}\|_1 + \|\mathbf{B}\|_* \\ & \text{subject to } \mathbf{B} + \mathbf{T} = \mathbf{Y}. \end{aligned} \quad (3.2)$$

A stable version of RPCA was introduced in [105] which adds an additional noise term to allow for perturbations to the low-rank matrix:

$$\mathbf{Y} = \mathbf{B} + \mathbf{T} + \mathbf{N}. \quad (3.3)$$

Recovery is then performed by solving the dual formulation given by

$$\min_{\mathbf{B}, \mathbf{T}} \frac{1}{2} \|\mathbf{Y} - \mathbf{B} - \mathbf{T}\|_2^2 + \lambda \|\mathbf{T}\|_1 + \gamma \|\mathbf{B}\|_*. \quad (3.4)$$

This approach to recovery serves as the basis of the algorithms presented in this chapter.

Low-rank plus sparse models have enjoyed success in IRST applications. In [99], infrared images are shown to fit into the low-rank plus sparse framework via the IPI model which decomposes each video frame into a matrix whose columns are vectorized overlapping patches from the original image. The authors demonstrate that background scene images yield a low-rank matrix under this patching procedure and then use RPCA to separate targets from their background. Subsequent work which enhances the IPI model by incorporating patches from a sequence of images into the data matrix [103]. This algorithm was shown to achieve state-of-the-art performance and serves as a point of comparison when we evaluate our method.

3.1.3 Optimal transport regularization for IRST

In this section, we describe how optimal transport can be used in the context of target tracking in infrared images. We introduce the RPCA+PUOT algorithm which incorporates UOT regularization into the traditional RPCA approach for recovering the components of the IPI model. Next, we enhance the IPI model by further allowing for an affine transformation of the background patches and introduce the RASL+PUOT algorithm for recovery.

UOT formulation, proximal algorithm, and proof of convergence

As a preliminary, we first introduce the UOT formulation and proximal algorithm (shown in Algorithm 1) which will be used in the subsequent algorithms as well as proof of its convergence². We use the UOT formulation of [84], denoted by $\tilde{\mathcal{V}}_\mu$ for its scalability and ability to cope with operands of unequal mass. This UOT formulation is based on the computationally efficient Beckmann problem and is defined as

$$\begin{aligned} \tilde{\mathcal{V}}_\mu(\mathbf{x}, \mathbf{y}) = \min \quad & \|\mathbf{M}\|_{2,1} + \mu \|\mathbf{r}\|_p^p \\ \text{subject to} \quad & \text{div}(\mathbf{M}) - \mathbf{y} + \mathbf{x} = \mathbf{r}. \end{aligned} \tag{3.5}$$

²Proximal algorithm developed by JL and is presented here for completeness with respect to the convergence proof.

When considering functions that appear as part of a larger optimization program as we will do in the following section, it is often useful to evaluate the corresponding proximal operator. The UOT proximal formulation is the optimization program given by

$$\text{prox}_{\rho\tilde{\mathcal{V}}_\mu}(\mathbf{p}_0, \mathbf{p}_1) = \underset{\mathbf{x}_0, \mathbf{x}_1 \geq 0}{\text{argmin}} \tilde{\mathcal{V}}_\mu(\mathbf{x}_0, \mathbf{x}_1) + \frac{1}{2\rho} \|\mathbf{x}_0; \mathbf{x}_1\|_2^2 - \mathbf{p}_0; \mathbf{p}_1\|_2^2. \quad (3.6)$$

The solution to Equation 3.6 lies at the saddle point described by

$$\min_{M, \mathbf{x}, \mathbf{r}} \max_{\mathbf{a}} \mathcal{L}(M, \mathbf{r}, \mathbf{x}, \mathbf{a}), \quad (3.7)$$

which is solved iteratively in Algorithm 1.

Algorithm 1 Unbalanced Beckmann OT Proximal Algorithm.

Input: $M^{(0)}, \mathbf{x}^{(0)}, \mathbf{r}^{(0)}, \mathbf{a}^{(0)}, \mu, \rho, \tau_1, \tau_2$

Output: \mathbf{x}^k

- 1: $k = 1$
 - 2: **while** not converged **do**
 - 3: $\mathbf{m}_i^{k+1} = S_{\tau_1}^{\ell_2}(\mathbf{m}_i^k - \tau_1 \text{div}^*(\mathbf{a}^k)_i), \forall i$
 - 4: $\mathbf{x}^{k+1} = \max\left(0, \frac{\rho\tau_1}{1+\rho\tau_1}\mathbf{p} + \frac{1}{1+\rho\tau_1}(\mathbf{x}^k - \tau_1 \mathbf{A}^\top \mathbf{a}^k)\right)$
 - 5: $\mathbf{r}^{k+1} = S_{\mu\tau_1}^{\ell_1}(\mathbf{r}^k + \tau_1 \mathbf{a}^k)$
 - 6: $\mathbf{a}^{k+1} = \mathbf{a}^k + \tau_2 \mathbf{b}^{k+1}$
 - 7: **end while**
-

The following theorem provides an analytic guarantee that specifies step size conditions for the convergence of Algorithm 1 to the saddle point of (3.7).

Theorem 3.1.1 (Convergence guarantee). *Suppose $\tau_1\tau_2 < \frac{1}{\lambda_{\max}(\nabla^2)+3}$ where $\lambda_{\max}(\nabla^2)$ is the largest eigenvalue of the discrete Laplacian operator, ∇^2 . Then the steps in Algorithm 1 produce a series which converges to the saddle point of the Lagrangian, i.e.,*

$$(\mathbf{M}^k, \mathbf{r}^k, \mathbf{x}^k, \mathbf{a}^k) \rightarrow (\mathbf{M}^*, \mathbf{r}^*, \mathbf{x}^*, \mathbf{a}^*),$$

where $(\mathbf{M}^*, \mathbf{r}^*, \mathbf{x}^*, \mathbf{a}^*)$ is a solution of (3.7).

Proof. We proceed by showing that the conditions of Theorem 1 in [106] are satisfied. First, note that we may rewrite (3.6) as

$$\mathcal{L}(\mathbf{M}, \mathbf{r}, \mathbf{x}, \mathbf{a}) = G(\mathbf{M}, \mathbf{r}, \mathbf{x}) + \langle \mathbf{a}, \mathbf{K}\mathbf{b} \rangle - F(\mathbf{a}),$$

where $G(\mathbf{M}, \mathbf{r}, \mathbf{x}) = \|\mathbf{M}\|_{2,1} + \mu \|\mathbf{r}\|_p^p + \frac{1}{2\rho} \|\mathbf{x} - \mathbf{p}\|_2^2 + \iota_+(\mathbf{x})$, $\mathbf{K} = [\mathbf{D}, -\mathbf{I}, \mathbf{I}, -\mathbf{I}]$, \mathbf{D} is the matrix corresponding to the divergence operator, $\mathbf{b} = [\text{vec}(\mathbf{M}); \mathbf{x}; \mathbf{r}]$ and $F(\mathbf{a}) = 0$. The functions G and F are proper, convex, and lower semi-continuous, and \mathbf{K} is a linear operator. Note that if λ is an eigenvalue of a matrix $\mathbf{B}^*\mathbf{B}$, then $\lambda + 1$ is an eigenvalue of the matrix $\tilde{\mathbf{B}}^*\tilde{\mathbf{B}}$ where $\tilde{\mathbf{B}} = [\mathbf{B}, \pm\mathbf{I}]$ (this is easily verified by noting that if \mathbf{v} is the corresponding eigenvector of $\mathbf{B}^*\mathbf{B}$, then $[\lambda\mathbf{v}; \pm\mathbf{B}\mathbf{v}]$ is the corresponding eigenvector of $\tilde{\mathbf{B}}^*\tilde{\mathbf{B}}$). By repeated application of this identity, it follows that $\lambda_{\max}(\mathbf{K}) = \lambda_{\max}(\mathbf{D}^*\mathbf{D}) + 3$. Since $\mathbf{D}\mathbf{D}^*$ is the discrete Laplacian operator, $\lambda_{\max}(\nabla^2) + 3$ is the maximum eigenvalue of \mathbf{K} . Thus, the conditions of [106, Theorem 1] are satisfied when $\tau_1\tau_2 < \frac{1}{\lambda_{\max}(\nabla^2)+3}$. \square

RPCA+PUOT

We adopt the IPI model and augment (3.4) with an UOT dynamics regularizer by solving

$$\begin{aligned} \min_{\mathbf{B}, \mathbf{T} \geq 0} \quad & \frac{1}{2} \|\mathbf{Y} - \mathbf{B} - \mathbf{T}\|_2^2 + \lambda \|\mathbf{T}\|_1 + \gamma \|\mathbf{B}\|_* \\ & + \kappa \sum_{p=1}^P \sum_{t=1}^{T-1} \tilde{\mathcal{V}}_\mu(\mathbf{t}_{pt}, \mathbf{t}_{p,t+1}), \end{aligned} \quad (3.8)$$

where p indexes over the patches and t indexes over time. The UOT term encodes the assumption that pixels nearby a target are more likely to contain a target in the next frame than pixels which are far away. We recover \mathbf{T} and \mathbf{B} by solving (3.8) via the alternating direction method of multipliers (ADMM) [107] which splits the problem into a sequence of smaller problems that are each efficient to solve.

First, we rewrite (3.8) using auxiliary variables to form an equivalent objective function

which decomposes into separable components which are each solvable via proximal methods [108]:

$$\begin{aligned}
& \min_{\mathbf{B}, \mathbf{T}, \mathbf{L}, \mathbf{Z}, \mathbf{W}} \frac{1}{2} \|\mathbf{Y} - \mathbf{B} - \mathbf{T}\|_F^2 + \lambda \|\mathbf{T}\|_1 + \gamma \|\mathbf{L}\|_* \\
& \quad + \iota_+(\mathbf{T}) + \iota_+(\mathbf{B}) \\
& \quad + \kappa \sum_{p=1}^P \sum_{t=1}^{T-1} \tilde{\mathcal{V}}_\mu(\mathbf{z}_{pt}, \mathbf{w}_{p,t+1}), \\
& \text{subject to } \mathbf{L} = \mathbf{T}, \\
& \quad \mathbf{t}_{pt} = \mathbf{z}_{pt}, \quad \mathbf{t}_{p,t+1} = \mathbf{w}_{p,t+1}, \\
& \quad t = 1, \dots, T-1, \quad p = 1, \dots, P,
\end{aligned} \tag{3.9}$$

where $\iota_+(\cdot)$ is the indicator function which is equal to zero in the nonnegative orthant and infinity otherwise.

The augmented Lagrangian for (3.9) is then given by

$$\begin{aligned}
\mathcal{L}(\mathbf{B}, \mathbf{T}, \mathbf{L}, \mathbf{Z}, \mathbf{W}, \mathbf{A}, \mathbf{C}, \mathbf{D}) &= \frac{1}{2} \|\mathbf{Y} - \mathbf{B} - \mathbf{T}\|_F^2 \\
& \quad + \iota_+(\mathbf{T}) + \iota_+(\mathbf{B}) + \lambda \|\mathbf{T}\|_1 \\
& \quad + \gamma \|\mathbf{L}\|_* + \frac{\rho}{2} \|\mathbf{L} - \mathbf{B} + \mathbf{A}\|_F^2 \\
& \quad + \sum_{p=1}^P \sum_{t=1}^{T-1} \left(\kappa \tilde{\mathcal{V}}_\mu(\mathbf{z}_{pt}, \mathbf{w}_{p,t+1}) + \frac{\rho}{2} \|\mathbf{t}_{pt} - \mathbf{z}_{pt} + \mathbf{c}_{pt}\|_2^2 \right. \\
& \quad \quad \left. + \frac{\rho}{2} \|\mathbf{t}_{p,t+1} - \mathbf{w}_{p,t+1} + \mathbf{d}_{p,t+1}\|_2^2 \right)
\end{aligned} \tag{3.10}$$

We proceed by minimizing \mathcal{L} with respect to \mathbf{B} , \mathbf{T} , \mathbf{L} , \mathbf{Z} and \mathbf{W} in turn. The initialization and update steps are summarized in Algorithm 2. The minimization with respect to \mathbf{T} involves the ℓ_1 , ℓ_2 and indicator function terms and has a closed form solution via the shrinkage operator which is defined as $S_\rho^+(x) = \max(x - \rho, 0)$. The update for \mathbf{B} is the solution to a constrained ℓ_2 minimization problem which is solved by projection onto

the nonnegative orthant. The auxiliary variable L involves minimizing the sum of the nuclear norm and ℓ_2 -norm terms which has a closed form solution via shrinkage on the singular values defined by the function $S_\rho^*(\mathbf{X}) = US_\rho^+(\Sigma)V^*$ where $\mathbf{X} = U\Sigma V^*$ is a singular value decomposition (SVD) of the input matrix \mathbf{X} . Minimization with respect to the operands of $\tilde{\mathcal{V}}_\mu$ is carried out via the proximal algorithm of [84] which efficiently solves the proximal function

$$\begin{aligned} \text{prox}_{\rho\tilde{\mathcal{V}}_\mu}(\mathbf{x}_1, \mathbf{x}_2) = \min_{\mathbf{z}_1, \mathbf{z}_2} & \tilde{\mathcal{V}}_\mu(\mathbf{z}_1, \mathbf{z}_2) \\ & + \frac{1}{2\rho} \|\mathbf{z}_1, \mathbf{z}_2 - \mathbf{x}_1, \mathbf{x}_2\|_2^2. \end{aligned} \quad (3.11)$$

Finally, we update the dual variables $\mathbf{A}, \mathbf{C}, \mathbf{D}$ as prescribed by ADMM and repeat until convergence. The algorithm terminates when the primal and dual residuals defined in [107] both attain values below a provided threshold.

We note that calculation of the SVD represents the most computationally intensive step in RPCA+PUOT. When the background is well represented using elements from a known dictionary, solving for \mathbf{B} as a sparse combination of these elements may yield a more efficient method for incorporating a-priori knowledge of structure in the background. If the background is known to have fixed rank r , it may be tempting to express \mathbf{B} as a product of factors $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N \times r}$. However, the resulting optimization program over \mathbf{U} and \mathbf{V} is non-convex and thus numerical computation of its solution is prone to local minima. Although this approach may work in practice, we focus on the convex formulation which allows the use of numerical methods which are guaranteed to converge to a globally optimal solution.

RASL+PUOT

The sparse plus low-rank model is especially useful when the imaging sensor is stationary which tends to result in a background matrix with lower rank. Next, we propose a method

Algorithm 2 RPCA+PUOT

Input: $\mathbf{Y}, \lambda, \gamma, \kappa, \mu, \rho$ **Output:** \mathbf{B}, \mathbf{T}

- 1: $\alpha_{pt} \leftarrow 0$ for $t = T$ and ρ otherwise
 - 2: $\beta_{pt} \leftarrow 0$ for $t = 1$ and ρ otherwise
 - 3: $\zeta_{pt} \leftarrow 1 + \alpha_{pt} + \beta_{pt}$
 - 4: **while** not converged **do**
 - 5: $\mathbf{f}_{pt} \leftarrow (\mathbf{y}_{pt} - \mathbf{b}_{pt}) + \alpha_{pt}(\mathbf{z}_{pt} + \mathbf{c}_{pt}) + \beta_{pt}(\mathbf{w}_{pt} + \mathbf{d}_{pt})$
 - 6: $\mathbf{t}_{pt} \leftarrow S_{\lambda/\zeta_{pt}}^+(\mathbf{f}_{pt}/\zeta_{pt})$
 - 7: $\mathbf{B} \leftarrow \max(0, (\rho(\mathbf{L} + \mathbf{A}) + (\mathbf{Y} - \mathbf{T}))/ (1 + \rho))$
 - 8: $\mathbf{L} \leftarrow S_{\gamma/\rho}^*(\mathbf{B} - \mathbf{A})$
 - 9: $(\mathbf{z}_{pt}, \mathbf{w}_{p,t+1}) \leftarrow \text{prox}_{\frac{\rho}{\kappa}\tilde{\mathcal{V}}_\mu}(\mathbf{t}_{pt} - \mathbf{c}_{pt}, \mathbf{t}_{p,t+1} - \mathbf{d}_{p,t+1})$
 - 10: $\mathbf{A} \leftarrow \mathbf{A} + \mathbf{L} - \mathbf{B}$
 - 11: $\mathbf{c}_{pt} \leftarrow \mathbf{c}_{pt} + \mathbf{z}_{pt} - \mathbf{t}_{pt}, \quad t = 1, \dots, T - 1$
 - 12: $\mathbf{d}_{pt} \leftarrow \mathbf{d}_{pt} + \mathbf{w}_{pt} - \mathbf{t}_{pt}, \quad t = 2, \dots, T$
 - 13: **end while**
-

based on the robust alignment by sparse and low-rank [109] algorithm which aims to improve performance when the imaging platform is nonstationary, such as in maritime or airborne applications. We augment the sparse plus low-rank model by allowing for warping of the data matrix by an affine transformation denoted by g_τ . Here, $\tau = (\tau_{pt})$ is the set of parameters which defines the affine transformation for each image patch in \mathbf{Y} . Our model then becomes

$$g_\tau(\mathbf{Y}) = \mathbf{B} + \mathbf{T} + \mathbf{N}, \quad (3.12)$$

and we aim to solve for τ , \mathbf{B} and \mathbf{T} via the optimization program

$$\begin{aligned} \min_{\tau, \mathbf{B}, \mathbf{T}} \quad & \frac{1}{2} \|g_\tau(\mathbf{Y}) - \mathbf{B} - \mathbf{T}\| + \lambda \|\mathbf{T}\|_1 + \gamma \|\mathbf{B}\|_* \\ & + \sum_{p=1}^P \sum_{t=1}^{T-1} \tilde{\mathcal{V}}_\mu(\mathbf{t}_{pt}, \mathbf{t}_{p,t+1}). \end{aligned} \quad (3.13)$$

Unfortunately, (3.13) is non-convex with respect to τ , so we adopt a similar strategy as [109] which iteratively solves a linearized version of the problem. In particular, we denote

by \mathbf{J}_{pt} the Jacobian of each image in \mathbf{Y} with respect to $\boldsymbol{\tau}$ and solve the linearized problem:

$$\begin{aligned} \min_{\Delta\boldsymbol{\tau}, \mathbf{B}, \mathbf{T}} \quad & \frac{1}{2} \sum_{t=1}^T \sum_{p=1}^P \left\| g_{\boldsymbol{\tau}}(\mathbf{y}_{pt}) + \mathbf{J}_{pt} \Delta\boldsymbol{\tau}_{pt} - \mathbf{b}_{pt} - \mathbf{t}_{pt} \right\|_2^2 \\ & + \lambda \|\mathbf{T}\|_1 + \gamma \|\mathbf{B}\|_* + \kappa \sum_{p=1}^P \sum_{t=1}^{T-1} \tilde{\mathcal{V}}_{\mu}(\mathbf{t}_{pt}, \mathbf{t}_{p,t+1}). \end{aligned} \quad (3.14)$$

We then update the solution of transform parameters via $\boldsymbol{\tau}_{pt} \leftarrow \boldsymbol{\tau}_{pt} + \Delta\boldsymbol{\tau}_{pt}$ and repeat. To solve (3.14), we use a similar variable splitting strategy as we did for (3.8) which results in the augmented Lagrangian given by

$$\begin{aligned} \mathcal{L}(\Delta\boldsymbol{\tau}, \mathbf{B}, \mathbf{T}, \mathbf{L}, \mathbf{Z}, \mathbf{W}, \mathbf{A}, \mathbf{C}, \mathbf{D}) = & \\ & \frac{1}{2} \sum_{t=1}^T \sum_{p=1}^P \left\| g_{\boldsymbol{\tau}}(\mathbf{y}_{pt}) + \mathbf{J}_{pt}^{(i)} \Delta\boldsymbol{\tau}_{pt} - \mathbf{b}_{pt} - \mathbf{t}_{pt} \right\|_2^2 \\ & + \iota_+(\mathbf{T}) + \iota_+(\mathbf{B}) + \lambda \|\mathbf{T}\|_1 + \gamma \|\mathbf{L}\|_* \\ & + \frac{\rho}{2} \|\mathbf{L} - \mathbf{T} + \mathbf{A}\|_F^2 + \sum_{p=1}^P \sum_{t=1}^{T-1} \left(\kappa \tilde{\mathcal{V}}_{\mu}(\mathbf{z}_{pt}, \mathbf{w}_{p,t+1}) \right. \\ & \quad \left. + \frac{\rho}{2} \|\mathbf{t}_{pt} - \mathbf{z}_{pt} + \mathbf{c}_{pt}\|_2^2 \right. \\ & \quad \left. + \frac{\rho}{2} \|\mathbf{t}_{p,t+1} - \mathbf{w}_{p,t+1} + \mathbf{d}_{p,t+1}\|_2^2 \right). \end{aligned} \quad (3.15)$$

Successive minimization of each of the variables in (3.15) results in the update steps described in Algorithm 4. Many of the updates are of a similar form as those in Algorithm 2 with the addition of an ℓ_2 minimization to update $\Delta\boldsymbol{\tau}$. The full description of the RASL+PUOT procedure is summarized in Algorithms 3 and 4.

3.1.4 Results

In this section, we first evaluate the efficacy of RPCA+PUOT for tracking scenarios where the imaging sensor is stationary, and evaluate RASL+PUOT on images with sensor movement. We compare our results against the algorithm described in [103] (denoted RPCA+MRF)

Algorithm 3 RASL Outer Loop [109]

Input: $Y, \lambda, \gamma, \kappa, \mu, \rho$ **Output:** τ, B, T

- 1: **while** not converged **do**
 - 2: $\mathbf{J}_{pt} \leftarrow \frac{\partial}{\partial \tau} g_{\tau}(\mathbf{y}_{pt})$
 - 3: solve for $\Delta \tau$ in (3.14) via Algorithm 4
 - 4: $\tau \leftarrow \tau + \Delta \tau$
 - 5: **end while**
-

Algorithm 4 UOT Regularized Inner Loop

Input: $Y, J, \lambda, \gamma, \kappa, \mu, \rho$ **Output:** $\Delta \tau, B, T$

- 1: $\alpha_{pt} \leftarrow 0$ for $t = T$ and ρ otherwise
 - 2: $\beta_{pt} \leftarrow 0$ for $t = 1$ and ρ otherwise
 - 3: $\zeta_{pt} \leftarrow 1 + \alpha_{pt} + \beta_{pt}$
 - 4: **while** not converged **do**
 - 5: $\tilde{\mathbf{y}}_{pt} \leftarrow \mathbf{y}_{pt} + \mathbf{J}_{pt} \Delta \tau$
 - 6: $\mathbf{f}_{pt} \leftarrow (\tilde{\mathbf{y}}_{pt} - \mathbf{b}_{pt}) + \alpha_{pt}(\mathbf{z}_{pt} + \mathbf{c}_{pt}) + \beta_{pt}(\mathbf{w}_{pt} + \mathbf{d}_{pt})$
 - 7: $\mathbf{t}_{pt} \leftarrow S_{\lambda/\zeta_{pt}}^+(\mathbf{f}_{pt}/\zeta_{pt})$
 - 8: $\mathbf{B} \leftarrow \max(0, (\rho(\mathbf{L} + \mathbf{A}) + (\tilde{\mathbf{Y}} - \mathbf{T}))/(\rho + 1))$
 - 9: $\mathbf{L} \leftarrow S_{\gamma/\rho}^*(\mathbf{B} - \mathbf{A})$
 - 10: $\tau_{pt} \leftarrow (\mathbf{J}_{pt}^{(i)})^{-1}(\mathbf{t}_{pt} + \mathbf{b}_{pt} - \mathbf{y}_{pt})$
 - 11: $(\mathbf{z}_{pt}, \mathbf{w}_{p,t+1}) \leftarrow \text{prox}_{\frac{\rho}{\kappa} \tilde{\mathcal{V}}_{\mu}}(\mathbf{t}_{pt} - \mathbf{c}_{pt}, \mathbf{t}_{p,t+1} - \mathbf{d}_{p,t+1})$
 - 12: $\mathbf{A} \leftarrow \mathbf{A} + \mathbf{L} - \mathbf{B}$
 - 13: $\mathbf{c}_{pt} \leftarrow \mathbf{c}_{pt} + \mathbf{z}_{pt} - \mathbf{t}_{pt}, \quad t = 1, \dots, T - 1$
 - 14: $\mathbf{d}_{pt} \leftarrow \mathbf{d}_{pt} + \mathbf{w}_{pt} - \mathbf{t}_{pt}, \quad t = 2, \dots, T$
 - 15: **end while**
-

which uses a similar signal model and was recently shown to yield state-of-the-art performance.

The data used for the simulation was generated using the Georgia Tech simulations integrated modeling system (GTSIMS) simulator [110, 111]. GTSIMS uses first principals based physics models driven by the material properties of the scene and environmental inputs to predict the response of a sensor in an open environment across the visible and infrared spectrum. GTSIMS also incorporates effects such as optical distortion and sensor noise to model how light is converted to a signal through a sensor. The simulated images are of a littoral scene off the coast of Georgia and South Carolina with a scanning IRST in the long-wave infrared $8\ \mu\text{m}$ to $12\ \mu\text{m}$ band. The full field of regard is 110° by 30° with an original image size of 1848×504 that was down sampled to 232×64 .

Algorithm parameters should be chosen according to several considerations. The joint scale of the parameters λ , γ , and κ is reflective of the amount of measurement noise which is penalized via the ℓ_2 term in (3.8). Target size and power influence the choice of λ as support size should be penalized heavily for small targets and weakly for large ones. Changes in target size or power are accounted for by the parameter μ which penalizes change in mass from frame to frame. Finally, κ should be tuned according to the target speed since the UOT penalty grows as the distance between a target in subsequent frames increases.

In practice, we tune these parameters automatically via a simple training procedure. In particular, we tune the parameters λ , γ , κ and μ for each problem configuration using direct search [75] over a logarithmic grid. We found this method to yield results which were at least comparable to manual tuning which is commonly practiced with sparse tracking algorithms.

In our first set of simulations, we generate data via

$$\mathbf{Y} = \mathbf{B} + \beta\mathbf{T} + \sigma\mathbf{N}, \quad (3.16)$$

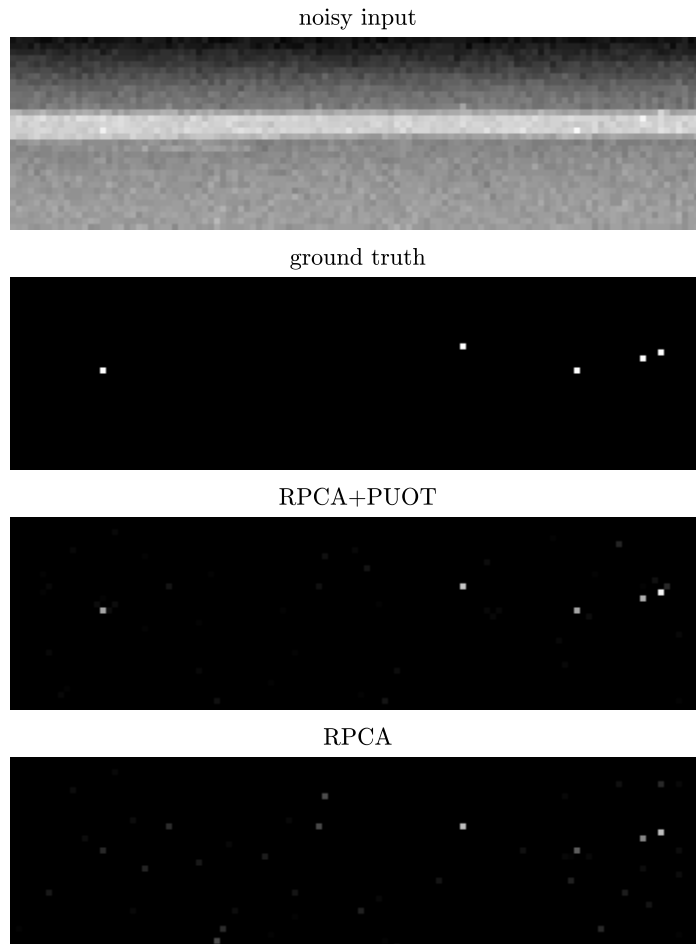


Figure 3.1: Example frame of simulated infrared video and representative recovery of sparse targets. Data are generated via the GTSIMS simulator which utilizes physical modelling and real terrain data to generate realistic images [111]. The ground truth consists of single pixel targets which move through the scene. Optimal transport regularization between successive frames allows RPCA+PUOT to effectively recover each of the targets with minimal noise. In contrast, standard RPCA contains noise with enough power to be confused for targets, and misses some targets entirely.

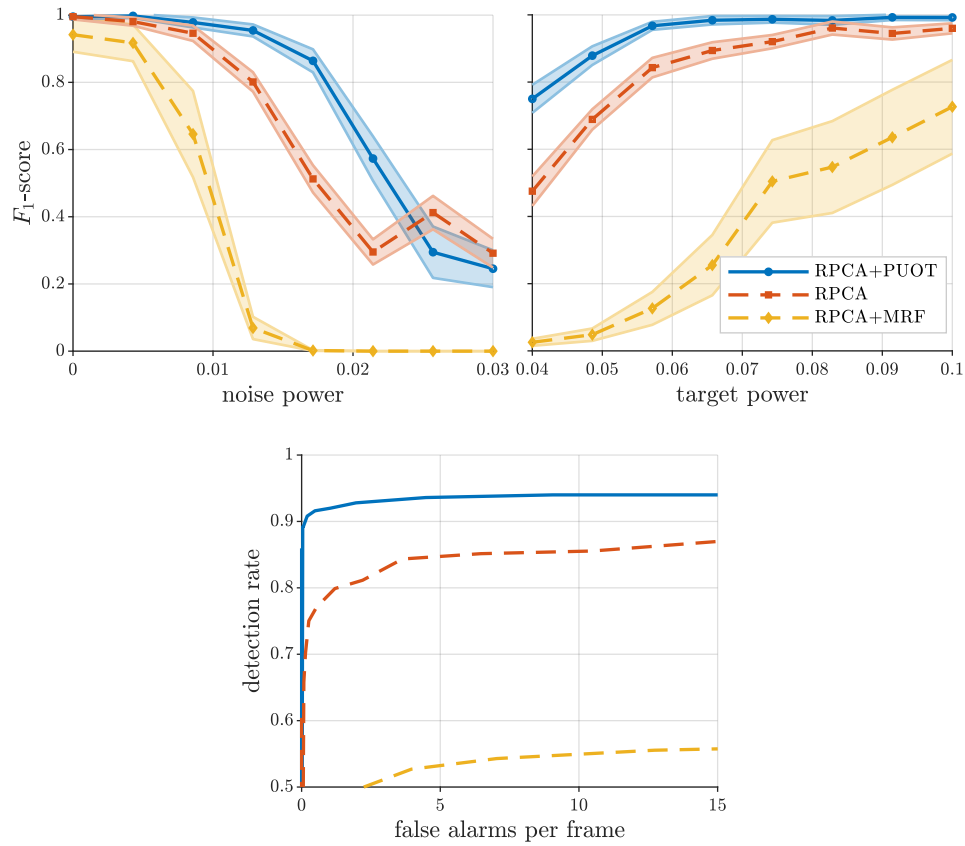


Figure 3.2: Aggregate tracking performance. The top two plots show how performance varies as a function of the noise and target power (denoted by β and σ respectively in (3.16)). The F_1 -score takes values in $[0, 1]$ where a score of one represents successful detection of all targets with no false alarms. The top left plot demonstrates that RPCA+PUOT achieves superior tracking performance in the presence of heavy sensor noise. The top right plot shows improved tracking performance in challenging scenarios where the target power is much less than that of the background. Interestingly, RPCA+MRF performs worse than the other two algorithms due to its inappropriate modelling of the small targets. Finally, the bottom plot shows ROC curves which confirm that the superior results shown in the top two plots are not due to a poor choice of threshold.

where the columns of \mathbf{B} contain a static background scene, \mathbf{T} contains sparse targets which move at random, and \mathbf{N} is a matrix with unit variance Gaussian entries. We use a single patch (i.e., $P = 1$) in order to decouple our evaluation of tracking performance from the image patching and reconstruction procedures which have already been proven to be effective in [99]. An example image and recovery is shown in Figure 3.1. Standard RPCA produces many false alarms due to the low signal-to-noise ratio (SNR) in the input image sequence. RPCA+PUOT in contrast readily recovers each of the targets while suppressing most of the noise. Figure 3.2 shows the trial averaged receiver operating curve which demonstrates that RPCA+PUOT allows higher detection accuracy for a given number of false alarms. Finally, we show how detection performance varies as a function of target power and noise levels. Default noise and target power were set to 0.01 and 0.07 respectively. Targets maintain constant power and move at a rate corresponding to a maximum of two pixels between frames. As an aggregate detection performance metric, we use the F_1 -score which is computed as the harmonic mean of the precision and recall. In particular, the F_1 -score is given by

$$F_1 = \frac{2TP}{2TP + FP + FN},$$

where TP, FP, and FN denote the number of true positives, false positives, and false negatives respectively. The F_1 -score takes on values in the range $[0, 1]$ where a score of one represents perfect detection performance. Figure 3.2 demonstrates that RPCA+PUOT allows for improved detection performance in scenarios with higher sensor noise or lower target power (i.e., lower signal-to-clutter ratio). We note that the RPCA+MRF method actually yields degraded performance compared to standard RPCA due to the modeling assumption that pixels neighboring those which contain target or noise tend to display statistics of that same class. For the single pixel targets we consider here, this assumption does not hold.

Finally, we evaluate the performance of RASL+PUOT for scenarios where the imaging

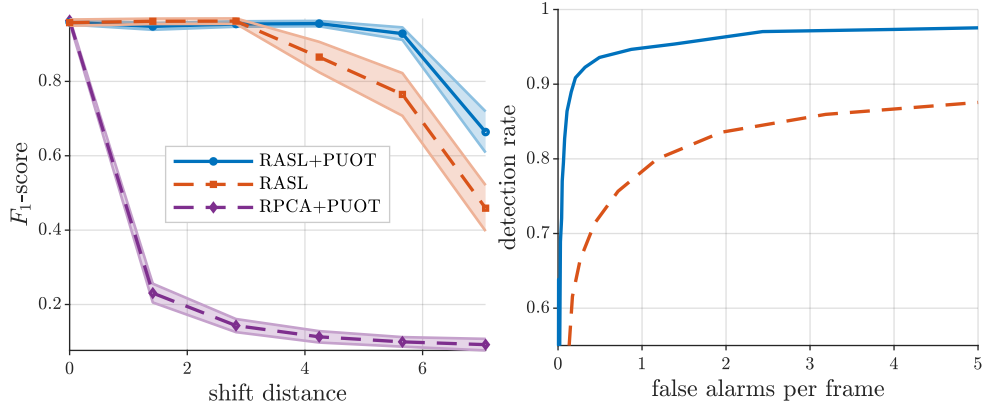


Figure 3.3: Performance of algorithms for simultaneous alignment and low-rank plus sparse recovery. UOT regularization enables successful target recovery in the presence of more severe sensor movement. Choosing a shift distance $\delta_{\max} = 5.8$, the ROC curve confirms that RASL+PUOT outperforms standard RPCA regardless of the choice of threshold.

sensor records unstabilized video. We generate data as

$$\mathbf{Y} = g_{\tau}(\mathbf{B} + \beta\mathbf{T} + \sigma\mathbf{N}),$$

where τ represents random translation of the input frames. Each frame shifts in the xy -plane uniformly at random up to a maximum of δ_{\max} away from the starting position. Figure 3.3 shows how optimal transport dynamics regularization in RASL+PUOT enables improved detection performance in the presence of more severe sensor movement.

3.1.5 Summary

While low-rank plus sparse models have proved to be effective in infrared target detection, many approaches do not effectively leverage dynamics information between successive frames. We propose the use of optimal transport regularization as a way to enforce temporal consistency and introduce two new algorithms: RPCA+PUOT which is tailored for the state-of-the-art IPI model, and an extension, RASL+PUOT, which further models sensor movement via affine transformations. We demonstrate using realistic simulated data that these new approaches can improve tracking performance.

3.2 Compressive RPCA for natural images

Here we present a collection of additional results on UOT regularized RPCA. Whereas the focus of section 3.1 is specifically on the infrared tracking problem, here we present a variation of the UOT regularized RPCA problem in which instead of performing strict denoising, we recover the sparse and low rank components from compressive measurements. We provide the relevant background and experimental results on real natural images.

3.2.1 Background: compressive RPCA

A rich literature has developed around the idea of sparse and low-rank decompositions [112]. One branch of work focuses on enforcing additional structure on the sparse component to encourage solutions that vary continuously over time, for example by employing optical flow based methods [113] or Markov random fields [114]. However, these methods assume full access to the original video frames and would require non-trivial adaptations to allow recovery in the context of an inverse problem.

We extend the stable formulation of RPCA [105] with linear measurements [115]

$$\min_{\mathbf{S}, \mathbf{L} \geq 0} \frac{1}{2} \sum_{t=1}^T (\|\mathbf{y}_t - \Phi_t(\mathbf{s}_t + \mathbf{l}_t)\|_2^2) + \lambda \|\mathbf{S}\|_1 + \gamma \|\mathbf{L}\|_*, \quad (3.17)$$

and show how our unbalanced optimal transport (OT) model can easily be incorporated to use optimal transport as a continuity regularizer on the sparse component.

3.2.2 Results: RPCA+UOT-DF with natural images

We evaluate performance on a video sequence of a person walking through an indoor scene. The snippet consists of 2 seconds of footage recorded at 30 frames per second and is downsampled to a resolution of 95×160 pixels ($T = 60$ and $N = 15,200$). We highlight the fact that OT-regularized problems of this size were simply intractable before our proposed proximal method. In such practical applications, the foreground component

may be darker or lighter than the background, so we must modify the robust PCA with UOT-regularized dynamic filtering (RPCA+UOT-DF) formulation to remove the $\mathbf{S} \geq 0$ constraint. Since the OT formulation takes nonnegative signals as its inputs, we decompose the sparse component into positive and negative components (in similar fashion as [19, 60, 61, 59]) $\mathbf{S} = \mathbf{S}^+ - \mathbf{S}^-$ with $\mathbf{S}^+, \mathbf{S}^- \geq 0$ and add an OT regularization term for each component. The RPCA+UOT-DF objective function then becomes

$$\begin{aligned} \min_{\mathbf{S}^+, \mathbf{S}^-, \mathbf{L} \geq 0} \quad & \frac{1}{2} \sum_{t=1}^T (\|\mathbf{y}_t - \Phi_t(\mathbf{s}_t^+ - \mathbf{s}_t^- + \mathbf{l}_t)\|_2^2) \\ & + \lambda(\|\mathbf{S}^+\|_1 + \|\mathbf{S}^-\|_1) + \gamma\|\mathbf{L}\|_* \\ & + \kappa \sum_{t=1}^{T-1} \left(\tilde{\mathcal{V}}_\mu(\mathbf{s}_t^+, \mathbf{s}_{t+1}^+) + \tilde{\mathcal{V}}_\mu(\mathbf{s}_t^-, \mathbf{s}_{t+1}^-) \right). \end{aligned} \quad (3.18)$$

We can solve (3.18) via Algorithm 5 which has a similar derivation to the one for the RPCA+UOT-DF algorithm of [84].

Algorithm 5 RPCA+UOT-DF for real-valued \mathbf{S}

Input: $\mathbf{Y}, \lambda, \gamma, \kappa, \mu, \rho$

Output: \mathbf{L}, \mathbf{S}

- 1: $\alpha_t \leftarrow 0$ for $t = T$ and ρ otherwise
 - 2: $\beta_t \leftarrow 0$ for $t = 1$ and ρ otherwise
 - 3: $\zeta_t \leftarrow 1 + \alpha_t + \beta_t$
 - 4: **while** not converged **do**
 - 5: $\mathbf{f}_t^\pm \leftarrow (\pm \mathbf{x}_t \mp \mathbf{l}_t \pm \mathbf{s}_t^\mp \pm \mathbf{a}_t) + \alpha_t(\mathbf{z}_t^\pm + \mathbf{b}_t^\pm) + \beta_t(\mathbf{w}_t^\pm + \mathbf{c}_t^\pm)$
 - 6: $\mathbf{s}_t^\pm \leftarrow S_{\lambda/\zeta_t}^+(\mathbf{f}_t/\zeta_t)$
 - 7: $\mathbf{L} \leftarrow \max(0, \frac{1}{2}(\mathbf{X} - \mathbf{S}^+ + \mathbf{S}^- + \mathbf{A} + \mathbf{T} - \mathbf{D}))$
 - 8: $\mathbf{x}_t \leftarrow (\Phi_t^T \Phi_t + \rho \mathbf{I})^{-1}(\Phi_t^T \mathbf{y}_t + \mathbf{s}_t^+ - \mathbf{s}_t^- + \mathbf{l}_t - \mathbf{a}_t)$
 - 9: $\mathbf{T} \leftarrow S_{\gamma/\rho}^*(\mathbf{L} + \mathbf{D})$
 - 10: $(\mathbf{z}_t^\pm, \mathbf{w}_{t+1}^\pm) \leftarrow \text{prox}_{\frac{\rho}{\kappa} \tilde{\mathcal{V}}_\mu}(\mathbf{s}_t^\pm - \mathbf{b}_t^\pm, \mathbf{s}_{t+1}^\pm - \mathbf{c}_{t+1}^\pm)$
 - 11: $\mathbf{A} \leftarrow \mathbf{A} + \mathbf{X} - \mathbf{S}^+ + \mathbf{S}^- - \mathbf{L}$
 - 12: $\mathbf{b}_t^\pm \leftarrow \mathbf{b}_t^\pm + \mathbf{z}_t^\pm - \mathbf{s}_t^\pm, \quad t = 1, \dots, T-1$
 - 13: $\mathbf{c}_t^\pm \leftarrow \mathbf{c}_t^\pm + \mathbf{w}_t^\pm - \mathbf{s}_t^\pm, \quad t = 2, \dots, T$
 - 14: $\mathbf{D} \leftarrow \mathbf{D} + \mathbf{L} - \mathbf{T}$
 - 15: **end while**
 - 16: $\mathbf{S} \leftarrow \mathbf{S}^+ - \mathbf{S}^-$
-

In the following simulations, we observe linear random projections measurements (in

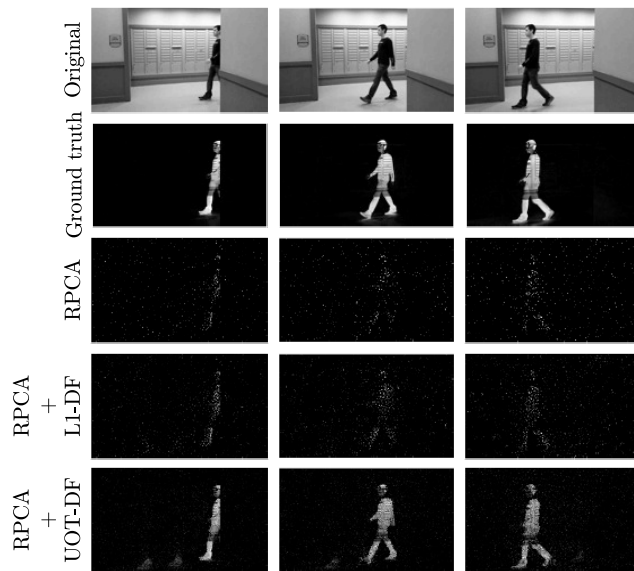


Figure 3.4: Separation of a moving subject from its environment using compressive measurements of a real video clip. Even under heavy compression ($M/N = 0.15$), the UOT regularizer enables nearly perfect recovery and separation of the person walking through the scene from the background. The foreground recovered by RPCA is noisy and the subject is all but lost. Although the dynamics regularizer in RPCA+L1-DF reduces noise somewhat, the ℓ_1 regularizer is not able to effectively leverage the continuity between frames and the subject remains barely distinguishable.

this case, severely compressed with $M/N = 0.15$) and use RPCA (3.17), robust PCA with ℓ_1 dynamic filtering (RPCA+L1-DF) and RPCA+UOT-DF (3.18) to extract the moving person from the background scene. As before, pattern search was employed in the selection of algorithm parameters. However, to avoid the prohibitive computation time required to optimize directly with the full resolution data, parameters were chosen by first using pattern search on heavily downsampled data to obtain an approximation to the optimal parameter set, and then fine-tuned manually using the original data. Unlike previous simulations, we found that the relationship $\gamma = \lambda\sqrt{N}$ did not yield optimal results, so λ and γ were selected independently. Figure 3.4 shows several example frames which demonstrate how the UOT regularizer enables successful recovery even after compression. RPCA misses the foreground almost entirely, while RPCA+L1-DF yields only a crude estimate due to the inability of the ℓ_1 dynamics regularizer to effectively capture continuity in the sparse component.

CHAPTER 4

EFFICIENT RANDOMIZED DIMENSIONALITY REDUCTION

The aim of this chapter is to evaluate the practical use of efficient filtering based on random projections in realistic data processing scenarios¹. Many areas in science and engineering now have access to technologies enabling rapid collection of overwhelming data volumes, but lack tools for efficient and effective transmission, storage, and processing. Here we present randomized filtering (RF) which leverages the fact that in many applications, streaming high-dimensional data evolves on a low-dimensional attractor manifold. In contrast to previous chapters, we assume here a non-linear manifold model which is well-suited to applications where the data vary smoothly as a function of a small collection of parameters. Compared to other dimensionality reduction approaches, RF embeds non-linear manifold structure while remaining data-independent and efficient to compute. Although rigorous mathematical analysis exists for the building blocks of RF, these theoretical results have until now remained divorced from practical utility. In this chapter, we introduce novel methods, analysis, and experimental verification to illuminate the practicality of RF in diverse scientific applications.

¹The material in this chapter was presented in [116] and was in collaboration with Dr. Adam Charles and Dr. Han Lun Yap. HLY developed the core theorem on which this work is based in [117]. NPB extended this abstract result on distance preserving matrices by posing a concrete filtering algorithm for dimensionality reduction, proposing example domain specific tasks, and conducting a thorough evaluation on data from a diverse range of application areas. ASC contributed crucial feedback during the brainstorming stages as well as on the presentation of the work, especially aspects related to neuroscience.

4.1 Introduction

Recent developments in sensor technology have presented scientists with a data deluge wherein the bottleneck in transforming observations into knowledge has shifted from the acquisition of data to its telemetry, storage, and processing. For example, in the pursuit of understanding the human brain, neuroscientists are investing massive resources in new neural recording technology with ever higher spatial and temporal resolution. Should such technologies reach their end goal of simultaneously recording the activity of every neuron in the human brain, the resulting data rate to capture a mere bit per neuron per second would be on the order of 100 Gbps. Indeed, such an endeavor may not be far beyond the horizon using existing materials and fabrication technologies [118]. Furthermore, recording hardware must be heavily power-constrained to achieve non-destructive tissue imaging. The ability to reduce the dimensionality of such data in a flexible and efficient manner is paramount to progress in such important data-rich applications.

High throughput streaming data compression requires techniques that are efficient (fast to compute in power constrained environments), data-independent (do not require training data), and universal (robust to changing data statistics and diverse tasks). Most compression methods, from classic principal components analysis [119] to modern variational autoencoders [120, 121], are data-dependent and thus suffer several shortcomings. First, training data are required to compute the dimensionality reduction map before new data can be compressed. This limits applicability to streaming and online applications which lack such training data. Second, such approaches are unsuitable for nonstationary data wherein the underlying statistics drift from the training dataset. Finally, data acquisition is often intertwined with the discovery process in which the practitioner does not know *a-priori* which features of the data will be important, making it difficult to determine the appropriate features to preserve. Although online principal components analysis [122] remedies some of these shortcomings, it incurs a time delay during the online training

process, has time complexity which is cubic in its accuracy parameter, and is restricted to a linear subspace model. Here, we present randomized filtering (RF), an alternative approach based on random projections [123], that possesses the necessary efficiency, universality, and data-independence for effective dimensionality reduction of streaming high-dimensional data. In previous work we laid the foundation for RF with a collection of abstract results for geometry preserving linear transformations [117]. Here we present the theoretical and technical extensions that make RF a practical tool across diverse scientific applications.

4.2 Methods

In this section, we describe the algorithmic steps involved in RF and present theoretical results on distance and angle preservation. We then describe methods for evaluating how well RF preserves the geometry of the input space as well as using the reduced measurements in practical signal processing tasks.

4.2.1 Stable manifold embeddings

Suppose we observe streaming measurements \mathbf{x}_t from a dynamical system which is evolving on a low-dimensional attractor manifold $\mathcal{M} \subset \mathbb{R}^N$. The application of RF to \mathbf{x}_t consists of three steps² (Fig. 4.1): randomly flipping the sign of each measurement, computing the discrete Fourier transform (DFT), and randomly subsampling the Fourier coefficients. As the algorithmic core is the DFT, the fast Fourier transform (FFT) enables a fast, efficient implementation in only $\mathcal{O}(N \log N)$ computations. Furthermore, the only filter parameters are the binary sign flips and a list of Fourier coefficients, meaning that the filter can be generated independently of the data and stored in $\mathcal{O}(N)$ memory.

The benefits of RF are a consequence of the fact that many observations are collected

²The reader may recognize these computational steps as the same ones in the fast Johnson-Lindenstrauss (JL) transform [124]. The important distinction to note is that the theoretical guarantees for RF assume a much more powerful manifold data model which is appropriate for streaming measurements as opposed to the finite point cloud model used in the Fast JL transform.

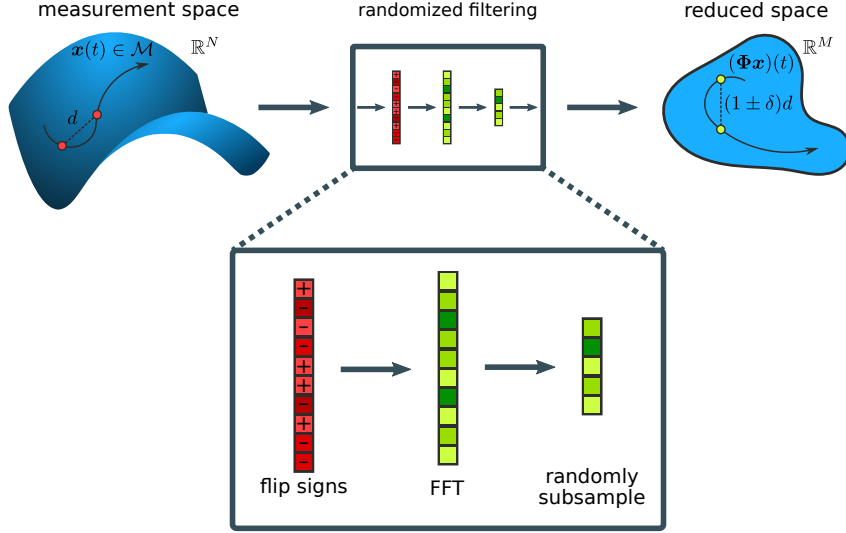


Figure 4.1: Illustration of RF which consists of three steps: 1. randomize the signs of the input vector; 2. compute the FFT of the result; 3. randomly subsample the Fourier coefficients. RF maps points from a D dimensional manifold residing in \mathbb{R}^N to the reduced space \mathbb{R}^M where $M < N$. Theorem 1 guarantees that for sufficiently large M , the mapping is a stable embedding, i.e., pairwise distances are approximately preserved. Since the main computational step involves the FFT, the algorithm is of complexity $\mathcal{O}(N \log N)$.

from a D -dimensional manifold in an N -dimensional ambient space, where $N \gg D$. Manifolds may be intuitively understood as a generalization of two-dimensional surfaces in three-dimensional space to hypersurfaces embedded in a higher dimension. Manifold-based modeling has proven effective in describing systems whose state depends nonlinearly on relatively few parameters, e.g., images of objects at different angles [1] or neural population activity [125].

The theoretical underpinnings of RF are based on the notion of *stable embeddings*; i.e., functions that preserve distances between pairs of points in the input space. Formally, we say that a map Φ is a stable embedding of \mathcal{M} if for every pair of points $\mathbf{x}, \mathbf{y} \in \mathcal{M}$ we have

$$1 - \delta \leq \frac{\|\Phi \mathbf{x} - \Phi \mathbf{y}\|_2^2}{\|\mathbf{x} - \mathbf{y}\|_2^2} \leq 1 + \delta. \quad (4.1)$$

Such stable embeddings provide critical robustness to measurement noise that may severely corrupt the measurements during compression. The following theorem [117] guarantees that

the map produced by RF is a stable embedding of the input manifold with high probability as long as the number of measurements M is sufficiently large³.

Theorem 4.2.1. *Let \mathcal{M} be a compact D -dimensional Riemannian submanifold of \mathbb{R}^N with geodesic regularity R , volume V , and condition number τ^{-1} . Let $F \in \mathbb{C}^{M \times N}$ be a subsampled Fourier matrix whose rows are chosen uniformly from the $N \times N$ DFT matrix, and let D_ξ be a diagonal Rademacher matrix, i.e., the entries along the diagonal of D_ξ are ± 1 with probability $1/2$. If*

$$M \geq \frac{C_1}{\delta_{\mathcal{M}}^2} \left(D \log \left(\frac{RN}{\tau \delta_{\mathcal{M}}} \right) + \log \left(\frac{V}{\rho} \right) \right) \log^4(N) \log(\rho^{-1}), \quad (4.2)$$

then with probability greater than $1 - C_2\rho$, $\Phi = FD_\xi$ stably embeds \mathcal{M} with isometry constant $\delta_{\mathcal{M}}$. Universal constants that do not vary with any other quantities in the theorem are denoted by C_1 and C_2 .

Note that the number of required measurements in Equation 4.2 grows linearly in D (which may be understood intuitively as the amount of “information” encoded by the manifold), and only logarithmically in the ambient dimension N .

The use of random projections for dimensionality reduction stems from the seminal work of Johnson and Lindenstrauss [128], who showed the existence of a random projection operator that stably embeds arbitrary point clouds. Ensuing work expanded this theory to random embeddings that can be applied efficiently [129] as well as to proving embedding results for more general manifolds [130]. Theorem 1 unifies these results, providing an efficient embedding based on random projections that also generalizes to manifold data models. For streaming scientific measurements where the high-dimensional data is governed by an underlying low-dimensional dynamical system, combining the manifold model generalizations with efficient computation is critical for model-free online compression.

³We note that the number of measurements required may be further reduced beyond the bound given in Theorem 1 by applying appropriate post-processing steps at the filter output [126, 127]. However, for simplicity of exposition, we adopt the original procedure described in [117].

4.2.2 Preservation of inner products

In this section, we introduce a result on the preservation of inner products which serves as theoretical justification for the approximation of the uncompressed estimator in the neuroimaging simulations. Specifically, under the mild assumption that Φ is a stable embedding of $\mathcal{M} \cup -\mathcal{M}$ (i.e. that Φ also stably embeds the manifold's reflection about the origin), we develop a bound for the deviation of inner products computed in the reduced space versus the original space. We note that the bound of [131, Theorem 4] cannot be applied directly due to its implicit assumption that Φ is also a stable embedding of the unit vectors in the direction of each point on \mathcal{M} . Instead, we have the following theorem.

Theorem 4.2.2. *Suppose that $\ell, s \in \mathcal{M}$ and that Φ is a δ -stable embedding of $(\mathcal{M} \cup -\mathcal{M})$, then*

$$|\langle \Phi \ell, \Phi s \rangle - \langle \ell, s \rangle| \leq \frac{\delta}{2} (\|\ell\|_2^2 + \|s\|_2^2).$$

Proof. Since Φ is a stable embedding of $(\mathcal{M} \cup -\mathcal{M})$, we have

$$1 - \delta \leq \frac{\|\Phi \ell \pm \Phi s\|^2}{\|\ell \pm s\|^2} \leq 1 + \delta,$$

or equivalently (by expanding the norm in the denominator)

$$1 - \delta \leq \frac{\|\Phi \ell \pm \Phi s\|^2}{\|\ell\|^2 + \|s\|^2 \pm 2 \langle \ell, s \rangle} \leq 1 + \delta.$$

By the parallelogram law,

$$\begin{aligned} \langle \Phi \ell, \Phi s \rangle &= \frac{\|\Phi \ell + \Phi s\|^2 - \|\Phi \ell - \Phi s\|^2}{4} \\ &\leq \frac{(1 + \delta)(\|\ell\|^2 + \|s\|^2 + 2 \langle \ell, s \rangle) - (1 - \delta)(\|\ell\|^2 + \|s\|^2 - 2 \langle \ell, s \rangle)}{4} \\ &= \langle \ell, s \rangle + \frac{\delta}{2} (\|\ell\|^2 + \|s\|^2), \end{aligned}$$

Similarly, we can show that

$$\langle \Phi \ell, \Phi s \rangle \geq \langle \ell, s \rangle - \frac{\delta}{2}(\|\ell\|^2 + \|s\|^2).$$

Taken together, these two inequalities yield the desired result

$$-\frac{\delta}{2}(\|\ell\|^2 + \|s\|^2) \leq \langle \Phi \ell, \Phi s \rangle - \langle \ell, s \rangle \leq \frac{\delta}{2}(\|\ell\|^2 + \|s\|^2).$$

□

Remark. *If we further assume that $x/\|x\|_2 \in \mathcal{M} \quad \forall x \in \mathcal{M}$, then we recover the result in [131, Theorem 4] which deals mostly with sparse signal models rather than the manifold models considered here.*

4.2.3 Isometry constant estimation

Here we outline a procedure for estimating the isometry constant δ given dataset $\mathbf{D} \in \mathbb{R}^{N \times K}$ with K samples of dimension N . We first compute all pairwise distances in the original space:

$$d_o(i, j) = \|\mathbf{x}_i - \mathbf{x}_j\|_2, \quad i < j \leq K,$$

and then in the reduced space

$$d_r(i, j) = \|\mathbf{y}_i - \mathbf{y}_j\|_2, \quad i < j \leq K,$$

where $\mathbf{Y} = \Phi \mathbf{D}$. Next let $Q(i, j)$ denote the ratio of pairwise distances in the reduced space to those in the original space, i.e.,

$$Q(i, j) = \frac{d_o(i, j)}{d_r(i, j)},$$

and define

$$Q_{\min} = \min_{i,j} Q(i,j), \quad Q_{\max} = \max_{i,j} Q(i,j), \quad Q_{\text{mean}} = \frac{2}{K(K-1)} \sum_{i<j} Q(i,j).$$

We estimate the lower and upper isometry constants as

$$\delta_{\text{lower}} = 1 - \frac{Q_{\min}}{Q_{\text{mean}}}, \quad \delta_{\text{upper}} = \frac{Q_{\max}}{Q_{\text{mean}}} - 1,$$

where we normalize by the mean ratio to account for any constant scaling caused by Φ .

Finally, we take δ to be the worse of the two values above:

$$\delta = \max(\delta_{\text{lower}}, \delta_{\text{upper}}).$$

4.2.4 Estimation with complex-valued measurements: real-valued noise

Let $\mathbf{X} \sim \mathcal{N}(\mathbf{s}, \sigma^2 \mathbf{I}_N)$ and $\mathbf{Z} = \Phi \mathbf{X}$ where $\Phi \in \mathbb{C}^{M \times N}$. Suppose we wish to describe the density function associated with \mathbf{Z} in order to compute a maximum likelihood estimate (MLE) of one of the model parameters. Since the mean and covariance matrix of \mathbf{Z} are given by

$$\mathbb{E}[\mathbf{Z}] = \mathbb{E}[\Phi \mathbf{X}] = \Phi \mathbb{E}[\mathbf{X}] = \Phi \mathbf{s},$$

and

$$\begin{aligned} \text{Cov}[\mathbf{Z}, \mathbf{Z}] &= \mathbb{E}[(\mathbf{Z} - \mathbb{E}[\mathbf{Z}])(\mathbf{Z} - \mathbb{E}[\mathbf{Z}])^*] \\ &= \mathbb{E}[(\Phi \mathbf{X} - \mathbb{E}[\Phi \mathbf{X}])(\Phi \mathbf{X} - \mathbb{E}[\Phi \mathbf{X}])^*] \\ &= \Phi \mathbb{E}[(\mathbf{X} - \mathbb{E}[\mathbf{X}])(\mathbf{X} - \mathbb{E}[\mathbf{X}])^*] \Phi^* \\ &= \Phi \text{Cov}[\mathbf{X}, \mathbf{X}] \Phi^* \\ &= \Phi \Phi^*, \end{aligned} \tag{4.3}$$

respectively, it may be tempting to express the distribution using the well-known formula for the multivariate Gaussian, i.e.,

$$f_{\mathbf{Z}}(\mathbf{z} | \mathbf{s}) = (2\pi)^{-n/2} \det(\mathbf{\Phi}\mathbf{\Phi}^*)^{-1/2} \exp \left\{ -\frac{1}{2\sigma^2} (\mathbf{z} - \mathbf{\Phi}\mathbf{s})^* (\mathbf{\Phi}\mathbf{\Phi}^*)^{-1} (\mathbf{z} - \mathbf{\Phi}\mathbf{s}) \right\}, \quad (4.4)$$

where $\mathbf{\Phi}^*$ denotes the conjugate transpose of the matrix $\mathbf{\Phi}$. Unfortunately, the mean and covariance matrix alone are insufficient to fully specify the distribution of \mathbf{Z} . To see this, it is useful to instead consider the real-valued random vector produced by the canonical isomorphism from $\mathbb{C}^m \rightarrow \mathbb{R}^{2m}$ given by

$$\mathbf{a} + i\mathbf{b} \mapsto \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}.$$

In our problem, we can generate this vector via the measurement matrix

$$\mathbf{A} := \begin{bmatrix} \mathbf{\Phi}_R \\ \mathbf{\Phi}_I \end{bmatrix},$$

where $\mathbf{\Phi}_R = \Re\{\mathbf{\Phi}\}$ and $\mathbf{\Phi}_I = \Im\{\mathbf{\Phi}\}$ denote the real and imaginary components of $\mathbf{\Phi}$. Denoting the resulting measurements by $\mathbf{Y} = \mathbf{A}\mathbf{X}$, we know from the analysis of real Gaussian vectors that the distribution of \mathbf{Y} is determined by the mean $\mathbf{A}\mathbf{s}$ and covariance matrix $\mathbf{A}\mathbf{A}^T$. The likelihood function is then given by⁴.

$$f_{\mathbf{Y}}(\mathbf{y} | \mathbf{s}) = (2\pi)^{-n/2} \det(\mathbf{A}\mathbf{A}^T)^{-1/2} \exp \left\{ -\frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{A}\mathbf{s})^T (\mathbf{A}\mathbf{A}^T)^{-1} (\mathbf{y} - \mathbf{A}\mathbf{s}) \right\}.$$

To understand why we cannot use the likelihood function (4.4), we will argue that the mean and covariance matrix are insufficient to describe the distribution of \mathbf{Y} . To begin, we

⁴We assume here that $\mathbf{A}\mathbf{A}^T$ is non-singular. In the singular case, one may consider a subset of the entries in \mathbf{Y} such that the corresponding covariance matrix is non-singular. Alternatively, we may replace the determinant and inverse with the pseudo-determinant and pseudo-inverse respectively.

note that the covariance matrix $\mathbf{A}\mathbf{A}^T$ may be written in block form as

$$\mathbf{A}\mathbf{A}^T = \begin{bmatrix} \Phi_R \Phi_R^T & \Phi_R \Phi_I^T \\ \Phi_I \Phi_R^T & \Phi_I \Phi_I^T \end{bmatrix}$$

In contrast, the covariance matrix associated with \mathbf{y} is given by

$$\Phi\Phi^* = (\Phi_R \Phi_R^T + \Phi_I \Phi_I^T) + i(\Phi_I \Phi_R^T - \Phi_R \Phi_I^T).$$

Since $\Phi\Phi^*$ contains only sums of the blocks of $\mathbf{A}\mathbf{A}^T$, it is impossible to infer the covariance matrix of \mathbf{Y} —and hence the distribution of \mathbf{Z} —from the covariance matrix of \mathbf{Z} alone.

To specify the distribution of \mathbf{Z} , we require one additional quantity called the *pseudo-covariance* matrix defined by [132]

$$\mathbb{E}[\mathbf{Z}\mathbf{Z}^T] = \Phi\Phi^T = (\Phi_R \Phi_R^T - \Phi_I \Phi_I^T) + i(\Phi_I \Phi_R^T + \Phi_R \Phi_I^T). \quad (4.5)$$

The covariance and pseudo-covariance matrices of \mathbf{Z} be used to compute the full covariance matrix of \mathbf{Y} . Thus the mean, covariance, and pseudo-covariance of \mathbf{Z} fully specify its distribution.

If the pseudo-covariance matrix of the noise is equal to zero, then it follows a special type of complex Gaussian distribution called the *circularly symmetric Gaussian* [133]. In this case, (4.5) yields

$$\Phi_R \Phi_R^T = \Phi_I \Phi_I^T, \quad \Phi_I \Phi_R^T = -\Phi_R \Phi_I^T.$$

The covariance matrix of \mathbf{Y} then becomes

$$\mathbf{A}\mathbf{A}^T = \begin{bmatrix} \Phi_R \Phi_R^T & \Phi_R \Phi_I^T \\ -\Phi_R \Phi_I^T & \Phi_R \Phi_R^T \end{bmatrix},$$

and each block may be computed using the covariance matrix of \mathbf{Z} which is given by

$$\Phi\Phi^* = 2\Phi_R\Phi_R^T - 2i\Phi_R\Phi_I^T.$$

In other words, the mean and covariance matrix are sufficient to determine the distribution of circularly symmetric Gaussian vectors. Furthermore, it may be shown that the corresponding density function is of the form in (4.4). In the setting of this work, the noise is not assumed to be circularly symmetric, so we carry out analysis using \mathbf{Y} . However, assuming a circularly symmetric complex noise model as an approximation to the real noise model simplifies analysis and yields connections to estimators in the uncompressed space, so we also consider this noise model in subsequent sections.

4.2.5 Estimation with complex-valued measurements: complex-valued noise

We now consider a variation on the problem in the previous section where we instead assume complex-valued noise. We denote by $\mathcal{N}_{\mathbb{C}}(\mathbf{0}, \mathbf{K})$ the distribution of a circularly symmetric Gaussian random vector with covariance matrix \mathbf{K} . Our signal model may then be expressed as $\mathbf{X} \sim \mathbf{s} + \epsilon$ where $\epsilon \sim \mathcal{N}_{\mathbb{C}}(0, \sigma^2 \mathbf{I}_N)$. As before, we observe data through a complex measurement operator $\Phi \in \mathbb{C}^{M \times N}$ via $\mathbf{Z} = \Phi\mathbf{X}$ and wish to estimate a parameter of the model. As in the previous section, the covariance matrix of \mathbf{Z} is given by $\Phi\Phi^*$, but in this case, since the noise is circularly symmetric, \mathbf{s} and the covariance matrix are sufficient to fully specify the density function:

$$f_{\mathbf{Z}}(\mathbf{z} | \mathbf{s}) = \frac{1}{\pi^n \sigma^2 \det(\Phi\Phi^*)} \exp \left\{ -\frac{1}{\sigma^2} (\mathbf{z} - \Phi\mathbf{s})^* (\Phi\Phi^*)^{-1} (\mathbf{z} - \Phi\mathbf{s}) \right\}. \quad (4.6)$$

We will see in subsequent sections that estimators simplify especially well under this noise model and the randomized filtering matrix Φ .

4.2.6 Application: time series estimation from reduced measurements

Suppose \mathbf{s} is a spatial profile represented by a vectorized binary mask (i.e., $\mathbf{s}[i] = 1$ if pixel i contains the cell of interest and 0 otherwise). Let $\mathbf{x}_t = \mu_t \mathbf{s} + \sigma \boldsymbol{\epsilon}$ represent the noisy image at time t whose mean is the spatial template modulated by a time varying parameter μ_t which we wish to estimate. In the original space, the distribution function of \mathbf{y}_t is then given by

$$f_{\mathbf{X}}(x_t | \mu_t, \mathbf{s}) = (2\pi)^{-n/2} \prod_i \exp \left\{ -\frac{1}{2\sigma^2} (\mathbf{x}_t - \mu_t \mathbf{s})^T (\mathbf{x}_t - \mu_t \mathbf{s}) \right\},$$

Setting the derivative with respect to μ_t of the log likelihood equal to zero yields

$$\hat{\mu}_t = \frac{\mathbf{x}_t^T \mathbf{s}}{\|\mathbf{s}\|^2}.$$

Now suppose we are given measurements in the reduced space given by $\mathbf{z}_t = \Phi \mathbf{x}_t$. Defining the equivalent measurement matrix $\mathbf{A} = [\mathbf{A}_R; \mathbf{A}_I]$ and letting $\mathbf{y}_t = \mathbf{A} \mathbf{x}_t$, we have the distribution function given by

$$f_{\mathbf{Y}}(\mathbf{y}_t | \mu_t, \mathbf{s}) = C \exp \left\{ -\frac{1}{2\sigma^2} (\mathbf{y}_t - \mu_t \mathbf{A} \mathbf{s})^T (\mathbf{A} \mathbf{A}^T)^{-1} (\mathbf{y}_t - \mu_t \mathbf{A} \mathbf{s}) \right\},$$

where $C = (2\pi)^{-n/2} \det(\mathbf{A} \mathbf{A}^T)^{-1/2}$. Setting the derivative log likelihood equal to zero yields the following estimator:

$$\hat{\mu}_t = \frac{\mathbf{y}_t^T (\mathbf{A} \mathbf{A}^T)^{-1} \mathbf{A} \mathbf{s}}{(\mathbf{A} \mathbf{s})^T (\mathbf{A} \mathbf{A}^T)^{-1} \mathbf{A} \mathbf{s}}. \quad (4.7)$$

Thus, the estimator is an inner product between the measurements and the spatial profile in the reduced space after whitening by the matrix $(\mathbf{A} \mathbf{A}^T)^{-1}$.

While (4.7) is the technically correct MLE for μ_t , it is also instructive to consider the estimator under a circularly symmetric complex noise model. In this case, the compressed

measurements \mathbf{z} are then distributed as

$$\mathbf{z}_t \sim \mathcal{N}_{\mathbb{C}}(\mu_t \Phi \mathbf{s}, \sigma^2 \Phi \Phi^*),$$

and by (4.6) we have the distribution function

$$f_{\mathbf{z}}(\mathbf{z}_t | \mu_t, \mathbf{s}) = \frac{1}{\pi^n \det(\Phi \Phi^*)} \exp \left\{ -\frac{1}{\sigma^2} (\mathbf{z}_t - \mu_t \Phi \mathbf{s})^* (\Phi \Phi^*)^{-1} (\mathbf{z}_t - \mu_t \Phi \mathbf{s}) \right\}.$$

Maximizing the likelihood is equivalent to minimizing the quadratic form in the exponential which we do by computing its derivative:

$$\begin{aligned} \frac{\partial Q}{\partial \mu_t} &= \frac{\partial}{\partial \mu_t} ((\mathbf{z}_t - \mu_t \Phi \mathbf{s})^* (\Phi \Phi^*)^{-1} (\mathbf{z}_t - \mu_t \Phi \mathbf{s})) \\ &= \frac{\partial}{\partial \mu_t} (\mathbf{z}_t^* (\Phi \Phi^*)^{-1} \mathbf{z}_t - \mu_t \mathbf{z}_t^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s} - \mu_t (\Phi \mathbf{s})^* (\Phi \Phi^*)^{-1} \mathbf{z}_t + \mu_t^2 (\Phi \mathbf{s})^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s}) \\ &= \frac{\partial}{\partial \mu_t} (\mathbf{z}_t^* (\Phi \Phi^*)^{-1} \mathbf{z}_t - \mu_t \mathbf{z}_t^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s} - \mu_t (\mathbf{z}_t^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s})^* + \mu_t^2 (\Phi \mathbf{s})^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s}) \\ &= \frac{\partial}{\partial \mu_t} (\mathbf{z}_t^* (\Phi \Phi^*)^{-1} \mathbf{z}_t - 2\mu_t \Re \{ \mathbf{z}_t^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s} \} + \mu_t^2 (\Phi \mathbf{s})^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s}) \\ &= -2\Re \{ \mathbf{z}_t^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s} \} + 2\mu_t (\Phi \mathbf{s})^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s} \end{aligned} \quad (4.8)$$

As before, we set the derivative with respect to μ_t of the log likelihood equal to zero which yields the MLE for a general sensing matrix Φ :

$$\hat{\mu}_t = \frac{\Re \{ \mathbf{z}_t^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s} \}}{(\Phi \mathbf{s})^* (\Phi \Phi^*)^{-1} \Phi \mathbf{s}}.$$

In the RF setting, the estimator simplifies significantly. Note that the construction of Φ for randomized filtering may be expressed as $\Phi = \mathbf{S} \mathbf{F} \mathbf{D}$ where \mathbf{S} is the matrix which selects a subset of the rows in the matrix it multiplies, \mathbf{F} is the DFT matrix, and \mathbf{D} contains

a Rademacher sequence on its diagonal. Then we have

$$\Phi\Phi^* = SFDDF^*S^T = I_M,$$

where I_M is the $M \times M$ identity matrix. Thus, the estimator simplifies to

$$\hat{\mu}_t = \frac{\Re\{z_t^* \Phi \mathbf{s}\}}{\|\Phi \mathbf{s}\|^2},$$

which is the estimator (4.11). Alternatively, in light of Theorem 4.2.2, we may view the estimator in the compressed space as an approximation of the one in the original space.

4.2.7 Application: classification from reduced measurements

Suppose we measure noisy observations of an image with mean equal to an unknown element \mathbf{s}^* from a set of known candidates S :

$$\mathbf{x} \sim \mathcal{N}(\mathbf{s}^*, \sigma^2 \mathbf{I}_N), \quad \mathbf{s}^* \in S.$$

The density function is given by

$$f_{\mathbf{X}}(\mathbf{x} | \mathbf{s}_i) = (2\pi)^{-n/2} \exp\left\{-\frac{1}{2\sigma^2}(\mathbf{x} - \mathbf{s}_i)^T(\mathbf{x} - \mathbf{s}_i)\right\},$$

which we maximize with respect to i yielding the classifier:

$$\hat{\mathbf{s}} = \operatorname{argmin}_{\mathbf{s}_i \in S} \|\mathbf{x} - \mathbf{s}_i\|^2.$$

Now suppose we observe reduced measurements via the RF operator Φ . As before, we consider the equivalent measurements through the corresponding operator \mathbf{A} which yields

the likelihood function

$$f_{\mathbf{Y}}(\mathbf{y} | \mathbf{s}_i) = (2\pi)^{-n/2} \det(\mathbf{A}\mathbf{A}^T) \exp \left\{ -\frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{A}\mathbf{s}_i)^T (\mathbf{A}\mathbf{A}^T)^{-1} (\mathbf{y} - \mathbf{A}\mathbf{s}_i) \right\},$$

so the corresponding classifier is

$$\hat{\mathbf{s}} = \underset{\mathbf{s}_i \in S}{\operatorname{argmin}} (\mathbf{y} - \mathbf{A}\mathbf{s}_i)^T (\mathbf{A}\mathbf{A}^T)^{-1} (\mathbf{y} - \mathbf{A}\mathbf{s}_i).$$

Finally, if we approximate the noise as circularly symmetric and note that $\Phi\Phi^* = \mathbf{I}_M$, the likelihood function may be expressed as

$$f_{\mathbf{Z}}(\mathbf{z} | \mathbf{s}_i) = (2\pi)^{-n/2} \exp \left\{ -\frac{1}{2\sigma^2} (\mathbf{z} - \Phi\mathbf{s}_i)^* (\mathbf{z} - \Phi\mathbf{s}_i) \right\},$$

which yields the classifier

$$\hat{\mathbf{s}} = \underset{\mathbf{s}_i \in S}{\operatorname{argmin}} \|\mathbf{z} - \Phi\mathbf{s}_i\|^2.$$

Again, we may view this as an approximation to the classifier in the original space which has error bounded by the isometry constant δ by virtue of the fact that Φ is a stable embedding.

4.3 Results

In this section, we explore the practical utility of RF in various application areas. We first provide a brief description of the datasets considered in subsequent sections.

4.3.1 Datasets

Sine manifold

The sine manifold, also called the complex exponential curve [134], is a one dimensional manifold in \mathbb{C}^N defined by

$$\mathbf{y}(t) = \begin{bmatrix} e^{-i2\pi f_C t} \\ e^{-i2\pi(f_C-1)t} \\ \vdots \\ e^{i2\pi(f_C-1)t} \\ e^{i2\pi f_C t} \end{bmatrix},$$

where f_C is an integer that defines the dimension of the ambient space via $N = 2f_C + 1$. Because the sine manifold can be easily embedded in arbitrary ambient dimension, and because bounds exist for its manifold properties (e.g., its reach), the sine manifold is an informative toy example in the analysis of manifold embeddings [134]. The results in Figure 4.2 are for a sine manifold with $f_C = 5,000$ resulting in an ambient dimension of $N = 10,001$.

Cardiac simulation

Mathematical modeling of the heart plays a key role in the development of treatments for patients with cardiovascular disease. Figure 4.2 shows the isometry constant for RF applied to data from a modified O’Hara-Virág-Varró-Rudy model [135]. The model we use incorporates real clinical data and faithfully reproduces key features of action potentials (e.g. duration, amplitude and resting membrane potential) for patients with heart failure. The data used here represent the voltage on a three dimensional grid consisting of $N = 4,129$ points over $K = 12,500$ time samples.

Calcium imaging

Calcium imaging is a staple neural recording technology in modern neuroscience. We generate data which simulates some of the key features of wide field calcium imaging recordings while simultaneously allowing full control over parameters of interest. Our data generation procedure consists of three components: creation of cell spatial profiles, creation of temporal events, and synthesis of the final dataset. Cell spatial profiles are generated sequentially by first centering a two dimensional Gaussian with radius r at a location chosen uniformly at random. The radius r follows a Gaussian distribution with a mean of one pixel and a standard deviation of 0.1. The next cell is generated in the same way, but with probability p_o the location is chosen such that the cell overlaps with the previous one. With probability $1 - p_o$, the cell location is again chosen uniformly at random. This process repeats until N_c cells have been placed and the resulting profiles are vectorized and stored as the columns of the $N \times N_c$ matrix \mathbf{S} . The results in Figure 4.3 use $N_c = 512$ cells spread over a 256×256 pixel field of view.

To generate the temporal event profiles, we perform the following procedure for each cell: first, generate a spike train in which the time between spikes follows the exponential distribution (i.e., a Poisson process), but with an additional constraint on the minimum time between spikes. This modification to the standard Poisson process simulates the refractory period during which a cell which has just fired is unable to fire again for a specified duration of time. The rate parameter for each cell is equal to one and the refractory period is 0.3. Next, we convolve the resulting spike train with a Gaussian bump to simulate the diffuse temporal profile commonly seen in calcium imaging recordings. The resulting temporal profile for each cell is stored in the $N_t \times N_c$ matrix \mathbf{T} .

Finally, data matrix is constructed by modulating each spatial profile by its corresponding temporal profile and adding noise via $\mathbf{D} = \mathbf{S}\mathbf{T}^T + \sigma\epsilon$.

Vorticity equations

The vorticity quantifies a fluid's localized tendency to rotate. The equations describing vorticity can be derived from the laws of conservation of mass and conservation of momentum, or from the Navier-Stokes equations. Here we consider shallow fluid flow which is well approximated by setting up the problem in two dimensions instead of three. For the results in Figure 4.4, we specify periodic boundary conditions and an initial condition with two vortex patches next to one another defined by

$$\omega_0(x, y) = e^{-17.5(x-0.45)^2-0.7y^2} + e^{-17.5(x+0.45)^2-0.7y^2}.$$

We solve the vorticity equations (4.12) via spectral methods which are particularly well-suited for problems defined over rectangular domains with periodic boundary conditions, yielding fast solutions with accuracy beyond all algebraic orders.

First, we rewrite the top equation in (4.12) as

$$\partial_t \omega = \gamma \Delta \omega + f - (\nabla^\perp \psi) \cdot \nabla \omega.$$

Computing the Fourier transform of the both sides with respect to x and then y , we have

$$\begin{aligned} \mathcal{F} \{ \partial_t \omega \} &= \mathcal{F} \{ \gamma \Delta \omega + f - (\nabla^\perp \psi) \cdot \nabla \omega \} \\ &= -\gamma \mathcal{F} \{ \omega \} (k_x^2 + k_y^2) + \mathcal{F} \{ f(\nabla^\perp \psi) \cdot \nabla \omega \}, \end{aligned} \quad (4.9)$$

where k_x and k_y are the wave numbers over x and y respectively. Similarly, for the bottom equation of (4.12) we have

$$\mathcal{F} \{ \omega \} = \frac{\mathcal{F} \{ \psi \}}{k_x^2 + k_y^2}.$$

We then use MATLAB's `ode45` ordinary differential equation solver to compute $\mathcal{F} \{ \omega \}$ and recover ω via the inverse transform.

Functional magnetic resonance imaging

Functional magnetic resonance imaging measures the brain by using blood flow as proxy for neural activity. Figure 4.2 shows the isometry constant for the publicly available Enhanced Nathan Kline Institute-Rockland Sample dataset [136]. We use slices 30–33 which contain a total of 24,964 voxels over 900 samples in time.

Voltage sensitive dye

Voltage sensitive dye (VSD) is an imaging technique which enables recordings of activations across multiple cortical columns for use in scientific applications. Figure 4.2 shows the isometry constant for the dataset reported in [137] which uses VSD to record from the primary sensory cortex in anesthetized rats for the purpose of studying adaptation to stimuli in the vibrissa pathway. The dataset contains recordings from multiple experimental trials for a total of 3,000 image samples containing 22,632 pixels per image.

4.3.2 Distance preservation in practice

Although Theorem 1 guarantees that RF is a stable embedding when condition in Equation 4.2 is satisfied, we seldom have access to the manifold parameters involved in this expression. It may thus be unclear how many measurements are required to practically apply Theorem 1. Given representative data, we can empirically explore the isometry constant δ as a function of the compression ratio N/M^5 . Remarkably, RF produces low estimates of δ across a wide range of synthetic and real datasets despite using no prior knowledge of the particular details for each data type (Fig. 4.2). As a point of comparison, low-pass filtering (LPF)—a dimensionality reduction method with our desired properties of efficiency, universality and data-independence—has higher isometry constants and therefore cannot

⁵For a given dataset, pairwise distances are exhaustively computed in the input and output space and the most severely distorted distance is used to compute the isometry constant δ . The value of δ is averaged over instantiations of the randomized filtering parameters. More details on this procedure may be found in section 4.2.3.

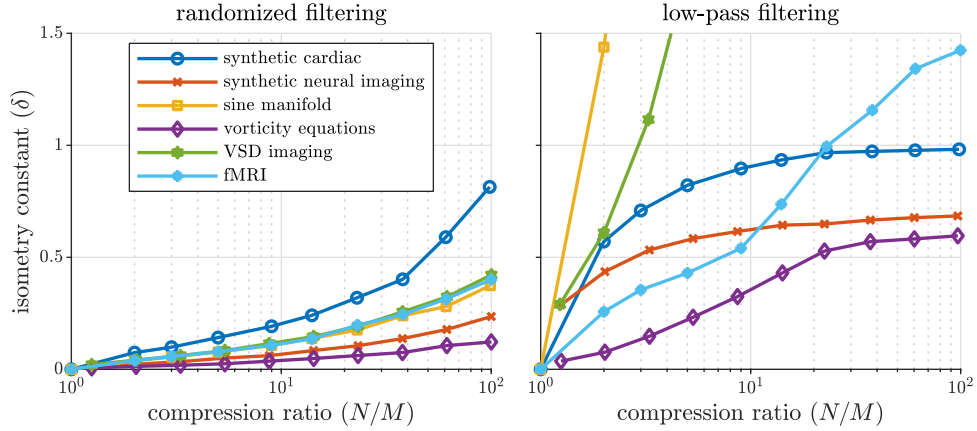


Figure 4.2: The isometry constant δ quantifies how well an embedding preserves the geometry of the input space. A value of $\delta = 0$ corresponds to a perfect embedding where distances between all pairs of input points are equal to corresponding distances in the reduced space. Shown are estimates of the isometry constant for several synthetic datasets (cardiac model [135], the neural imaging data used in Fig. 4.3, the sine manifold [134], and solutions to the vorticity equations as in Fig. 4.4) as well as real datasets (voltage sensitive dye imaging from rodent experiments [137] and functional magnetic resonance imaging [136]). Lower isometry constants may be achieved with far fewer measurements with RF (left) compared to LPF (right) which lacks similar stability guarantees.

preserve the data geometry critical for ensuing analysis. Note that when using RF in a given application for the first time, choosing the single compression ratio parameter will require far less data compared to data-dependent methods which require full training to characterize the underlying structure. Furthermore, this initial tuning need only be carried out once, after which the resulting value may be used in a variety of experimental conditions if the same underlying manifold model applies. As RF becomes more widely used, the choice of compression ratio may become largely experience driven. To further assess the applicability of RF in practice, we explore how its geometry preservation translates to real-world data processing tasks: neural activity inference from calcium imaging and flow classification from vorticity measurements.

4.3.3 Application: neural imaging

One rapidly growing area where fast, online compression will be critical to stemming the data deluge in the near future is in high-density neural recordings such as optical functional microscopy [138, 139] and high-density electrode designs [140]. While necessary to capture the many nuances of neural activity, the extremely high resolution of these datasets limits the ability to efficiently process them or to potentially transmit the data wirelessly from freely moving animal subjects with chronic implantations. Here we consider one such recording technology—calcium imaging—which is both a staple neuroimaging technique and actively being developed to capture larger neural volumes with higher fidelity [141, 142, 143].

To test the applicability of RF to such datasets, we generate synthetic calcium imaging data with the goal of recovering neural activity directly from reduced measurements⁶. Simulated data affords full control over the data characteristics which permits the assessment of RF in specific parameter regimes. Denote by $\mathbf{S} \in \mathbb{R}^{N \times N_c}$ the matrix whose columns correspond to vectorized cell profiles, $\mathbf{T} \in \mathbb{R}^{N_t \times N_c}$ the matrix whose columns contain the temporal profiles of each cell, and $\mathbf{D} = \mathbf{S}\mathbf{T}^T + \sigma\epsilon$ the data matrix with additive white Gaussian noise with standard deviation σ . In the absence of noise and cell overlap, the time series for cell i (denoted by t_i) may be recovered via the inner product of the data matrix with its spatial cell profile⁷:

$$\hat{t}_i = \mathbf{D}^T \mathbf{s}_i / \|\mathbf{s}_i\|_2^2, \quad (4.10)$$

where \mathbf{s}_i is the spatial profile of cell i . Although Theorem 1 does not guarantee complete recovery of the original data, we can still approximate quantities of interest in the reduced

⁶Data are generated by first placing N_c circular neuron cell profiles in the $W \times H$ pixel scene. Each neuron may overlap with the previously generated one with a fixed probability. Temporal dynamics are generated by convolving a Poisson spiking process over N_t time samples which we subsequently convolve with a Gaussian kernel intended to approximate the temporal profile seen in calcium imaging.

⁷In the presence of Gaussian noise, this can be shown to be the maximum likelihood estimator. Cell overlap may be considered as an additional source of noise.

space [131]. In particular, we can approximate the inner product as^{8,9}

$$\hat{t}_i = \Re \{ (\Phi D)^T \Phi s_i \} / \|\Phi s_i\|_2^2, \quad (4.11)$$

where Φ is the RF operator. This estimator in the reduced space is an inner product similar to the one in the original space, but with the dimensionality reduction operator applied to both the data and cell profiles. Using RF with Equation (4.11), event times are recoverable using a fraction of the original measurements (Fig. 4.3).

4.3.4 Application: fluid flow classification

A second example of high-dimensional, complex dynamical data arises in the study of nonlinear fluid flow behavior such as turbulence—a fundamental phenomenon in fluid mechanics which can be notoriously difficult to model—via optical measurements of the vorticity fields. Vorticity quantifies the local tendency for rotational movement in a fluid and, for the case of two dimensional incompressible flow, may be described as a scalar ω field that obeys the following system of partial differential equations¹⁰ [144]

$$\begin{aligned} \partial_t \omega + (\nabla^\perp \psi) \cdot \nabla \omega &= \gamma \Delta \omega + f \\ \Delta \psi &= \omega. \end{aligned} \quad (4.12)$$

Inspired by the simulations presented in [145], we consider two vortex patches interacting

⁸Under mild assumptions on Φ , stable embeddings can also preserve inner products. See Theorem 4.2.2 for a formal statement of this fact.

⁹Since the measurements are complex-valued, we need to take the real component of the inner product shown. The full derivation for this estimator is provided in the supplemental materials. Alternatively, one may consider an efficient real-valued restricted isometry property (RIP) matrix and apply the approach in [117] to produce a similar guarantee as Theorem 1. For example, operators based on the subsampled discrete cosine transform are real-valued, efficient, and RIP optimal, although their construction is slightly more complicated than the one presented here [127].

¹⁰The function $\omega(x, y, t)$ is the scalar vorticity field, $\psi(x, y, t)$ is the stream function, γ is the kinematic viscosity, $f(x, y)$ is a forcing function, $\nabla^\perp = (-\partial_y, \partial_x)^T$ is the perpendicular gradient operator, and $\Delta = \partial_{xx} + \partial_{yy}$ is the Laplace operator.

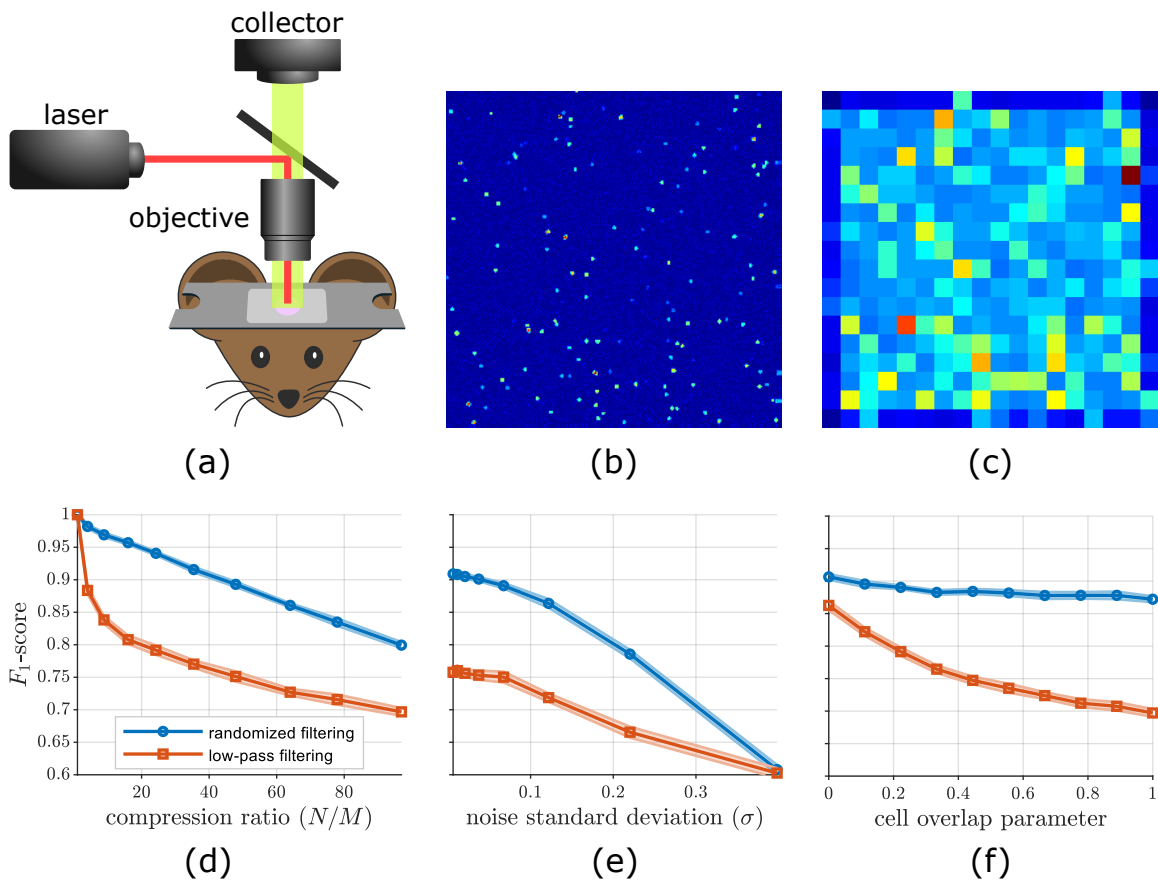


Figure 4.3: **(a)** Calcium imaging is a neural recording modality commonly used in animal experiments which uses florescent calcium indicators to measure activity in the brain. **(b)** Example frame of simulated wide-field calcium imaging data. **(c)** The same frame is blurred and downsampled by a factor of 15 in each spatial dimension as a crude form of dimensionality reduction. The fine details of the image are lost, especially in regions which contain overlapping cells. **(d-f)** Comparison of performance in an event detection task in synthetic wide-field calcium imaging data. Events are estimated by thresholding the output of (4.11) and the F_1 -score is used as the performance metric throughout (higher is better). **(d)** RF produces favorable results even after heavy compression. **(e)** RF outperforms low-pass filtering up to moderate levels of noise. **(f)** In the sequential generation of the cell body templates, a cell is forced to overlap with the previous cell with a prescribed probability. As the number of overlapping cells increases, low-pass filtering is unable to distinguish activity between nearby or overlapping cells. In contrast, RF is able to separate activity between individual cells with significant levels of overlap.

under the influence of a high-frequency forcing function f defined as

$$f(x, y) = 0.075 \frac{\sin(64x + \phi) + \sin(32y + \phi)}{1 + 0.25(\cos(128x + \phi) + \cos(64y + \phi))}. \quad (4.13)$$

We generate solutions to (4.12) using a forcing function described by (4.13) with an unknown phase ϕ chosen uniformly at random from a set of phase candidates $\{\phi_1, \dots, \phi_P\}$, and then perform classification to determine which phase generated the observed vorticity field. This classification simply selects the phase that minimizes the distance between the vorticity field at a given time and the forcing functions generated using each possible phase,

$$\hat{\phi} = \underset{\phi_i}{\operatorname{argmin}} \|\Phi\omega - \Phi f_{\phi_i}\|_2^2,$$

where $f_{\phi_i}(x, y)$ is the forcing function produced by phase ϕ_i . The ability to capture the high frequency content of the forcing term allows RF to produce higher-accuracy phase estimates than equivalent dimensionality reduction using LPF (Fig. 4.4).

4.4 Discussion

These two applications, taken together, demonstrate of the power of universal dimensionality reduction with RF. Despite the vastly different datasets, the dimensionality reduction procedure used was identical. It required no knowledge of either the data features or the post-compression task, making it extremely useful in discovery oriented experiments where the characteristics of interest are yet unknown.

We have explored the potential of RF by evaluating its performance on practical filtering tasks, but its applicability is in no way limited to this use case. As RF preserves the manifold geometry underlying the observations, it is compatible with other techniques that depend on this structure, such as hypothesis testing, clustering, regression, classification and manifold learning approaches like Locally Linear Embedding [146] or Isomap [147]. In imaging

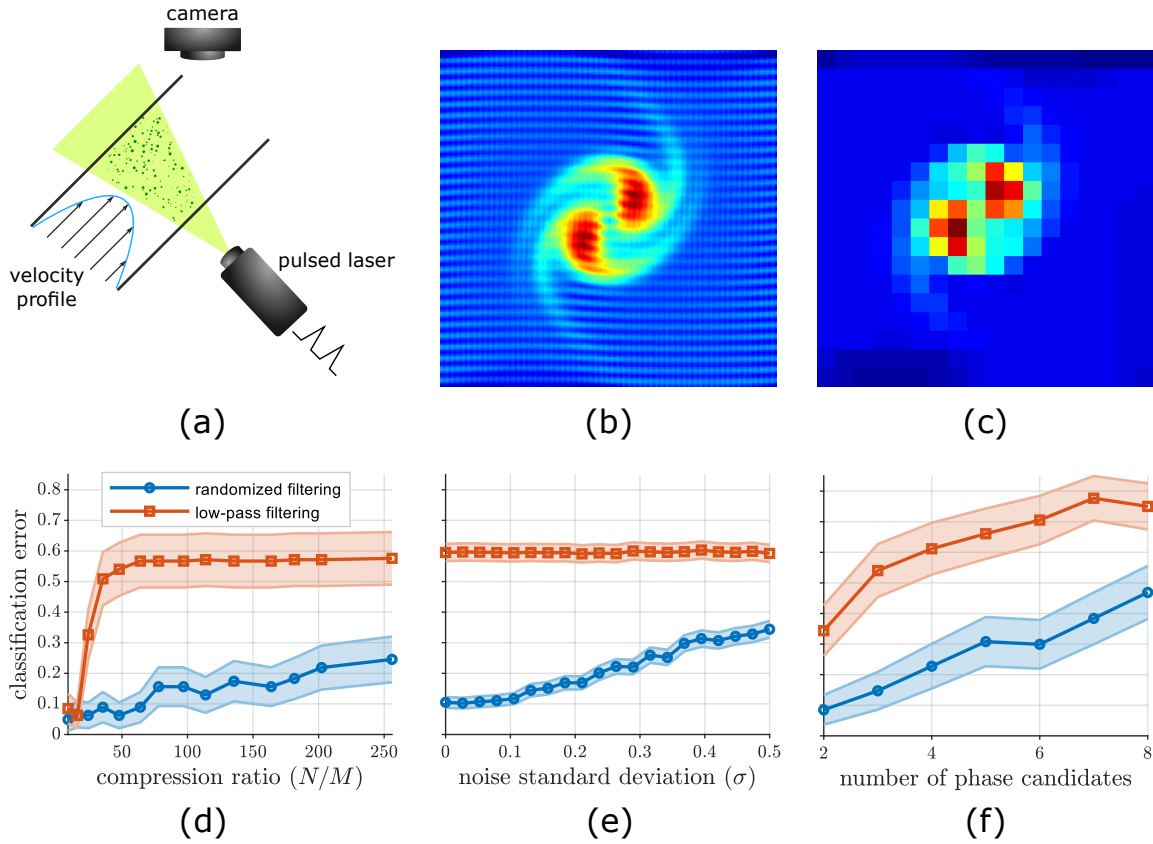


Figure 4.4: **(a)** Vorticity is a quantity which describes the local tendency of a fluid to rotate. By first measuring the velocity field, for example using particle image velocimetry as shown, vorticity may be measured experimentally. **(b)** Example of one frame in time of the solution to the vorticity equations described in Equation (4.12). **(c)** Crude dimensionality reduction is performed by resizing the image by a factor of 15 in each dimension using LPF and downsampling. As expected, the high-frequency content has vanished. **(d-f)** Performance in a phase classification task on solutions to the vorticity equations. Solutions are generated with forcing functions that have a certain unknown phase from a known set of candidates. The task is to classify which phase was used to generate the solution based on compressed measurements of the solution. **(d)** RF achieves a low classification error rate even after 2 – 3 orders of magnitude of compression. **(e)** RF produces superior phase classification performance in noisy conditions, whereas information necessary for classification is completely lost under low-pass filtering. **(f)** RF allows for correct classification more often than low-pass filtering even when the problem is made more difficult by drawing phases from a larger set of candidates.

applications, RF could even be implemented optically by using a lens to calculate the DFT and remove the computational cost associated with dimensionality reduction completely.

The development of new dimensionality reduction tools is essential in ensuring that data abundance is a blessing and not a curse. Here we have demonstrated a simple yet powerful dimensionality reduction tool with wide applicability. The shift from deterministic to randomized schemes represents a fundamental change in the way we think about dimensionality reduction, but it may be the only way to stay afloat as data volumes continue to surge.

CHAPTER 5

CONCLUSION

5.1 Summary

Understanding the underlying structure of our observations in scientific and engineering tasks is crucial especially in scenarios where measurements contain uncertainty. One source of such uncertainty is noise in the measurement process (e.g. an imaging sensor operating in low light conditions). Alternatively, uncertainty may arise due to design constraints of the measuring system which necessitate subsampling to ease the burden of the telemetry and processing pipelines. In each case, leveraging a-priori knowledge of the structure in the underlying system can enable vastly improved performance.

In this thesis, we primarily explore two types of structure: geometric structure among signal elements and smooth manifold structure.

- In Chapter 2, we consider problems where the signals of interest have a natural geometry or ordering over their elements. Optimal transport proves to be a powerful framework for exploiting this geometric structure which can lead to significant performance improvements in time series target tracking applications. We introduce several variants of the earth mover's distance dynamic filtering (EMD-DF) algorithm which are shown to improve performance in tracking targets in imaging applications, tracking wavefronts in oscillator network models, and tracking frequencies in electrophysiology simulations.
- Next, Chapter 3 explores how optimal transport can be combined with other structured models by incorporating optimal transport (OT) regularization in the well-known robust principal components analysis (RPCA) problem. Simulations on natural and

infrared video sequences demonstrate how OT can encode target continuity between neighboring images and enable successful tracking.

- Finally, Chapter 4 considers efficient dimensionality reduction for data from a dynamical system which evolves on a low-dimensional attractor manifold. We build on the theoretical foundation laid out in [117] by proposing a practical filtering scheme. We demonstrate that randomized filtering may be suitable in a host of application areas despite the fact that it uses no training data in its construction. Even in the absence of signal recovery guarantees, we show that practical tasks such as parameter estimation and classification can be performed in neuroscience and fluid mechanics applications.

5.2 Limitations

As we demonstrated in chapters 2 and 3, optimal transport regularization can significantly improve tracking performance in many problems. However, we found that the benefits of OT regularization truly shine in particularly challenging tracking scenarios (i.e., ones with high compression ratio or low signal-to-noise ratio (SNR)) where conventional methods fail. In problems with limited uncertainty, simpler methods such as the Kalman filter or basis pursuit denoising with dynamic filtering (BPDN-DF) may offer acceptable performance without incurring the computational penalties associated even with efficiency-oriented OT approaches. Thus, we recommend its use primarily after less complex methods have already been applied without success.

On the other end of the spectrum, the dimensionality reduction tools presented in Chapter 4 are extremely efficient and make very mild assumptions about the manifold structure of the underlying system. However, specialized methods which leverage domain specific knowledge typically yield superior performance. Thus, the randomized filtering approach presented here is best used when domain specific methods are unavailable or are too computationally intensive to be used at the required scale.

5.3 Conclusions and future work

Taken together, the results of this thesis highlight the trade-off that model sophistication presents between performance and practicality. We demonstrate in several application areas that tracking regularizers which more faithfully represent similarity between operands can yield superior performance compared to the simpler methods which are traditionally used. However, these benefits come at the cost in terms of computational complexity as well as ease-of-use (e.g., more involved implementation and parameter tuning). In many cases, traditional regularizers may provide sufficient descriptive power of the discrepancies between signals and the use of more complicated regularizers is not justified (e.g., when target signal elements between temporal samples contain significant overlap).

This thesis represents a starting point for future development in OT regularized tracking algorithms. In the context of dynamic filtering, we focussed on tracking targets that are sparse in the canonical basis. One avenue of future research might involve expanding these ideas to sparsity models in other bases which will require careful consideration of meaningful ways to compute distances between the sparse coefficients. For example, how one might consider incorporating wavelet tree structure into the optimal transport ground distance to enable tracking of natural images (which are known to be sparse in wavelet bases) without the need for the low-rank plus sparse structure studied in Chapter 3. Another direction for subsequent work may consider alternative mass transport models. We generalized the simple case of nonnegative signals by approximating the modulus of elements in complex-valued signals for the purpose of frequency tracking, but alternative models may be more appropriate in other application areas. Finally, future work might consider incorporating models for more complicated signal types such as color images.

The results presented here demonstrate that simple randomized approaches can provide a powerful dimensionality reduction tool in scenarios involving extreme data volumes. Although randomized algorithms have grown in popularity in many engineering disciplines,

we believe there are still hurdles to overcome before we see widespread use of these techniques for scientific discovery. In particular, the idea of using randomized measurements with no recovery guarantees of the original signal may feel uncomfortable to practitioners in fields where deterministic sampling methods (e.g. based on the Shannon-Nyquist theorem) have dominated as the de-facto measurement scheme. However, as data volumes continue to soar, we believe randomized measurement systems will become commonplace by necessity.

REFERENCES

- [1] D. L. Donoho and C. Grimes, “Image manifolds which are isometric to Euclidean space,” *Journal of Mathematical Imaging and Vision*, vol. 23, no. 1, pp. 5–24, Jul. 1, 2005.
- [2] G. Monge, *Mémoire sur la théorie des déblais et des remblais*. Paris: De l’Imprimerie Royale, 1781, OCLC: 51928110.
- [3] S. Peleg, M. Werman, and H. Rom, “A unified approach to the change of resolution: Space and gray-level,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 739–742, Jul. 1989.
- [4] Y. Rubner, C. Tomasi, and L. J. Guibas, “The earth mover’s distance as a metric for image retrieval,” *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, Nov. 1, 2000.
- [5] H. Ling and K. Okada, “An efficient earth mover’s distance algorithm for robust histogram comparison,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 840–853, May 2007.
- [6] R. Gupta, P. Indyk, and E. Price, “Sparse recovery for earth mover distance,” in *2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sep. 2010, pp. 1742–1744.
- [7] L. Schmidt, C. Hegde, and P. Indyk, “The constrained earth mover distance model, with applications to compressive sensing,” in *10th International Conference on Sampling Theory and Applications (SAMPTA)*, 2013.
- [8] D. Mo and M. F. Duarte, “Compressive parameter estimation with earth mover’s distance via K-median clustering,” presented at the Proc. SPIE 8858, Wavelets and Sparsity XV, vol. 8858, 2013.
- [9] M. Cuturi, “Sinkhorn distances: Lightspeed computation of optimal transport,” in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2013, pp. 2292–2300.
- [10] J. Benamou, G. Carlier, M. Cuturi, L. Nenna, and G. Peyré, “Iterative Bregman projections for regularized transportation problems,” *SIAM Journal on Scientific Computing*, vol. 37, no. 2, A1111–A1138, Jan. 1, 2015.

- [11] G. Peyré, “Entropic approximation of Wasserstein gradient flows,” *SIAM Journal on Imaging Sciences*, vol. 8, no. 4, pp. 2323–2351, Jan. 1, 2015.
- [12] M. Beckmann, “A continuous model of transportation,” *Econometrica*, vol. 20, no. 4, pp. 643–660, 1952.
- [13] W. Li, E. K. Ryu, S. Osher, W. Yin, and W. Gangbo, “A parallel method for earth mover’s distance,” *Journal of Scientific Computing*, vol. 75, no. 1, pp. 182–197, Apr. 1, 2018.
- [14] J. Karlsson and A. Ringh, “Generalized Sinkhorn iterations for regularizing inverse problems using optimal mass transport,” *SIAM Journal on Imaging Sciences*, vol. 10, no. 4, pp. 1935–1962, Jan. 1, 2017.
- [15] R. Sinkhorn, “A relationship between arbitrary positive matrices and doubly stochastic matrices,” *The Annals of Mathematical Statistics*, vol. 35, no. 2, pp. 876–879, 1964.
- [16] M. Cuturi and G. Peyré, “A smoothed dual approach for variational Wasserstein problems,” *SIAM Journal on Imaging Sciences*, vol. 9, no. 1, pp. 320–343, Jan. 1, 2016.
- [17] A. S. Charles, N. P. Bertrand, J. Lee, and C. J. Rozell, “Earth-mover’s distance as a tracking regularizer,” in *2017 IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Dec. 2017, pp. 1–5.
- [18] N. P. Bertrand, J. Lee, A. S. Charles, P. Dunn, and C. J. Rozell, “Sparse dynamic filtering via earth mover’s distance regularization,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2018, pp. 4334–4338.
- [19] N. P. Bertrand, A. S. Charles, J. Lee, P. B. Dunn, and C. J. Rozell, “Efficient tracking of sparse signals via an earth mover’s distance dynamics regularizer,” Jun. 12, 2018, Submitted. arXiv: 1806.04674 [eess].
- [20] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, Mar. 1, 1960.
- [21] M. Elad, M. A. T. Figueiredo, and Y. Ma, “On the role of sparse and redundant representations in image processing,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982, Jun. 2010.
- [22] R. G. Baraniuk, “Compressive sensing [lecture notes],” *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, Jul. 2007.

- [23] D. L. Donoho and J. Tanner, “Sparse nonnegative solution of underdetermined linear equations by linear programming,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 27, pp. 9446–9451, May 7, 2005. pmid: 15976026.
- [24] J. A. Tropp and A. C. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.
- [25] D. Needell and J. A. Tropp, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, May 1, 2009.
- [26] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, ser. Applied and Numerical Harmonic Analysis. New York: Birkhäuser, 2013, 625 pp., ISBN: 978-0-8176-4947-0.
- [27] M. S. Asif and J. Romberg, “Dynamic updating for ℓ_1 minimization,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 421–434, Apr. 2010.
- [28] M. S. Asif, A. Charles, J. Romberg, and C. Rozell, “Estimation and dynamic updating of time-varying signals with sparse variations,” in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011, pp. 3908–3911.
- [29] A. Carmi, P. Gurfil, and D. Kanevsky, “Methods for sparse signal recovery using Kalman filtering with embedded pseudo-measurement norms and quasi-norms,” *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 2405–2409, Apr. 2010.
- [30] N. Vaswani, “Kalman filtered compressed sensing,” in *2008 15th IEEE International Conference on Image Processing*, Oct. 2008, pp. 893–896.
- [31] D. Zachariah, S. Chatterjee, and M. Jansson, “Dynamic iterative pursuit,” *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4967–4972, Sep. 2012.
- [32] J. Ziniel, L. C. Potter, and P. Schniter, “Tracking and smoothing of time-varying sparse signals via approximate belief propagation,” in *2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers*, Nov. 2010, pp. 808–812.
- [33] N. Vaswani, “LS-CS-residual (LS-CS): Compressive sensing on least squares residual,” *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 4108–4120, Aug. 2010.

- [34] N. Vaswani and W. Lu, “Modified-CS: Modifying compressive sensing for problems with partially known support,” *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4595–4607, Sep. 2010.
- [35] D. Sejdinović, C. Andrieu, and R. Piechocki, “Bayesian sequential compressed sensing in sparse dynamical systems,” in *2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sep. 2010, pp. 1730–1736.
- [36] A. Charles, M. S. Asif, J. Romberg, and C. Rozell, “Sparsity penalties in dynamical system estimation,” in *2011 45th Annual Conference on Information Sciences and Systems*, Mar. 2011, pp. 1–6.
- [37] E. C. Hall and R. M. Willett, “Dynamical models and tracking regret in online convex programming,” in *Proceedings of the 30th International Conference on Machine Learning*, S. Dasgupta and D. McAllester, Eds., PMLR, Feb. 13, 2013, pp. 579–587.
- [38] A. S. Charles, A. Balavoine, and C. J. Rozell, “Dynamic filtering of time-varying sparse signals via ℓ_1 minimization,” *IEEE Transactions on Signal Processing*, vol. 64, no. 21, pp. 5644–5656, Nov. 2016.
- [39] A. Balavoine, C. J. Rozell, and J. Romberg, “Discrete and continuous-time soft-thresholding for dynamic signal recovery,” *IEEE Transactions on Signal Processing*, vol. 63, no. 12, pp. 3165–3176, Jun. 2015.
- [40] A. Figalli, “The optimal partial transport problem,” *Archive for Rational Mechanics and Analysis*, vol. 195, no. 2, pp. 533–560, Feb. 1, 2010.
- [41] E. K. Ryu, W. Li, P. Yin, and S. Osher, “Unbalanced and partial L_1 monge-kantorovich problem: A scalable parallel first-order method,” *Journal of Scientific Computing*, pp. 1–18, Nov. 15, 2017.
- [42] S. S. Haykin, *Kalman Filtering and Neural Networks*. New York: Wiley, 2001, OCLC: 52366672.
- [43] E. A. Wan and R. V. D. Merwe, “The unscented Kalman filter for nonlinear estimation,” in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, 2000, pp. 153–158.
- [44] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, “Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition,” in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, Nov. 1993, 40–44 vol.1.

- [45] E. Candès and J. Romberg, *ℓ_1 -magic: recovery of sparse signals via convex programming*. 2005.
- [46] S. J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, “An interior-point method for large-scale ℓ_1 -regularized least squares,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 606–617, Dec. 2007.
- [47] D. M. Malioutov, M. Cetin, and A. S. Willsky, “Homotopy continuation for sparse signal representation,” in *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 5, Mar. 2005, v/733–v/736 Vol. 5.
- [48] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, Dec. 2007.
- [49] E. J. Candès, M. B. Wakin, and S. P. Boyd, “Enhancing sparsity by reweighted ℓ_1 minimization,” *Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 877–905, Dec. 1, 2008.
- [50] P. Garrigues and B. A. Olshausen, “Group sparse coding with a laplacian scale mixture prior,” in *Advances in Neural Information Processing Systems 23*, J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, Eds., Curran Associates, Inc., 2010, pp. 676–684.
- [51] D. Angelosante, S. I. Roumeliotis, and G. B. Giannakis, “Lasso-Kalman smoother for tracking sparse signals,” in *2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, Nov. 2009, pp. 181–185.
- [52] S. Farahmand, G. B. Giannakis, and D. Angelosante, “Doubly robust smoothing of dynamical processes via outlier sparsity constraints,” *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4529–4543, Oct. 2011.
- [53] Z. Zhang and B. D. Rao, “Sparse signal recovery with temporally correlated source vectors using sparse Bayesian learning,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 912–926, Sep. 2011.
- [54] A. C. Sankaranarayanan, P. K. Turaga, R. G. Baraniuk, and R. Chellappa, “Compressive acquisition of dynamic scenes,” in *Computer Vision – ECCV 2010*, ser. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, Sep. 5, 2010, pp. 129–142.
- [55] M. S. Asif, L. Hamilton, M. Brummer, and J. Romberg, “Motion-adaptive spatio-temporal regularization for accelerated dynamic MRI,” *Magnetic Resonance in Medicine*, vol. 70, no. 3, pp. 800–812, Sep. 1, 2013.

- [56] A. Y. Aravkin, J. V. Burke, and G. Pillonetto, “Optimization viewpoint on Kalman smoothing with applications to robust and sparse estimation,” in *Compressed Sensing & Sparse Filtering*, ser. Signals and Communication Technology, Springer, Berlin, Heidelberg, 2014, pp. 237–280.
- [57] ———, “Sparse/robust estimation and Kalman smoothing with nonsmooth log-concave densities: Modeling, computation, and theory,” *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 2689–2728, Jan. 2013.
- [58] A. S. Charles and C. J. Rozell, “Spectral superresolution of hyperspectral imagery using reweighted ℓ_1 spatial filtering,” *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 3, pp. 602–606, Mar. 2014.
- [59] E. Mainini, “A description of transport cost for signed measures,” *Journal of Mathematical Sciences*, vol. 181, no. 6, pp. 837–855, Mar. 1, 2012.
- [60] H. Janati, M. Cuturi, and A. Gramfort, “Wasserstein regularization for sparse multi-task regression,” May 20, 2018. arXiv: 1805.07833 [cs, stat].
- [61] A. Profeta and K.-T. Sturm, “Heat flow with Dirichlet boundary conditions via optimal transport and gluing of metric measure spaces,” Sep. 4, 2018. arXiv: 1809.00936 [math].
- [62] M. Grant and S. Boyd, “CVX: MATLAB software for disciplined convex programming, version 2.1,” Mar. 2014.
- [63] M. C. Grant and S. P. Boyd, “Graph implementations for nonsmooth convex programs,” in *Recent Advances in Learning and Control*, ser. Lecture Notes in Control and Information Sciences, Springer, London, 2008, pp. 95–110.
- [64] S. R. Becker, E. J. Candès, and M. C. Grant, “Templates for convex cone problems with applications to sparse signal recovery,” *Mathematical Programming Computation*, vol. 3, no. 3, p. 165, Sep. 1, 2011.
- [65] R. Otazo, E. Candès, and D. K. Sodickson, “Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components,” *Magnetic Resonance in Medicine*, vol. 73, no. 3, pp. 1125–1136, 2015.
- [66] Z. Yin, T. Kanade, and M. Chen, “Understanding the phase contrast optics to restore artifact-free microscopy images for segmentation,” *Medical Image Analysis*, vol. 16, no. 5, pp. 1047–1062, Jul. 1, 2012.

- [67] H. Zhang, A. J. Watrous, A. Patel, and J. Jacobs, “Theta and alpha oscillations are traveling waves in the human neocortex,” *Neuron*, vol. 98, no. 6, 1269–1281.e4, Jun. 27, 2018.
- [68] G. B. Ermentrout and D. Kleinfeld, “Traveling electrical waves in cortex: Insights from phase dynamics and speculation on a computational role,” *Neuron*, vol. 29, no. 1, pp. 33–44, Jan. 1, 2001.
- [69] Y. Kuramoto, “Rhythms and turbulence in populations of chemical oscillators,” *Physica A: Statistical Mechanics and its Applications*, vol. 106, no. 1, pp. 128–143, Mar. 1, 1981.
- [70] T. J. Gardner and M. O. Magnasco, “Sparse time-frequency representations,” *Proceedings of the National Academy of Sciences*, vol. 103, no. 16, pp. 6094–6099, Apr. 18, 2006. pmid: 16601097.
- [71] D. J. Thomson, “Multitaper analysis of nonstationary and nonlinear time series data,” in *Nonlinear and Nonstationary Signal Processing*, W. J. Fitzgerald, Ed., OCLC: ocm45339400, Cambridge ; New York: Cambridge University Press, 2000, pp. 317–394, ISBN: 978-0-521-80044-0.
- [72] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, “The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis,” *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 454, no. 1971, pp. 903–995, Mar. 8, 1998.
- [73] F. Auger and P. Flandrin, “Improving the readability of time-frequency and time-scale representations by the reassignment method,” *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1068–1089, May 1995.
- [74] S. A. Fulop and K. Fitz, “Algorithms for computing the time-corrected instantaneous frequency (reassigned) spectrogram, with applications,” *The Journal of the Acoustical Society of America*, vol. 119, no. 1, pp. 360–371, Jan. 1, 2006.
- [75] T. Kolda, R. Lewis, and V. Torczon, “Optimization by direct search: New perspectives on some classical and modern methods,” *SIAM Review*, vol. 45, no. 3, pp. 385–482, Jan. 1, 2003.
- [76] L. Cornelissen, S.-E. Kim, P. L. Purdon, E. N. Brown, and C. B. Berde, “Age-dependent electroencephalogram (EEG) patterns during sevoflurane general anesthesia in infants,” *eLife*, vol. 4, J. C. Culham, Ed., e06513, Jun. 23, 2015.
- [77] J. Aru, J. Aru, V. Priesemann, M. Wibral, L. Lana, G. Pipa, W. Singer, and R. Vicente, “Untangling cross-frequency coupling in neuroscience,” *Current Opinion*

in Neurobiology, SI: Brain Rhythms and Dynamic Coordination, vol. 31, pp. 51–61, Apr. 1, 2015.

- [78] B. Schack, N. Vath, H. Petsche, H. .-.G. Geissler, and E. Möller, “Phase-coupling of theta–gamma EEG rhythms during short-term memory processing,” *International Journal of Psychophysiology*, vol. 44, no. 2, pp. 143–163, May 1, 2002.
- [79] R. T. Canolty, E. Edwards, S. S. Dalal, M. Soltani, S. S. Nagarajan, H. E. Kirsch, M. S. Berger, N. M. Barbaro, and R. T. Knight, “High gamma power is phase-locked to theta oscillations in human neocortex,” *Science*, vol. 313, no. 5793, pp. 1626–1628, Sep. 15, 2006. pmid: 16973878.
- [80] O. Jensen and L. L. Colgin, “Cross-frequency coupling between neuronal oscillations,” *Trends in Cognitive Sciences*, vol. 11, no. 7, pp. 267–269, Jul. 1, 2007.
- [81] A. B. L. Tort, R. W. Komorowski, J. R. Manns, N. J. Kopell, and H. Eichenbaum, “Theta–gamma coupling increases during the learning of item–context associations,” *Proceedings of the National Academy of Sciences*, vol. 106, no. 49, pp. 20942–20947, Aug. 12, 2009. pmid: 19934062.
- [82] M. A. Belluscio, K. Mizuseki, R. Schmidt, R. Kempter, and G. Buzsáki, “Cross-frequency phase–phase coupling between theta and gamma oscillations in the hippocampus,” *Journal of Neuroscience*, vol. 32, no. 2, pp. 423–435, Jan. 11, 2012. pmid: 22238079.
- [83] C. Kemere, M. F. Carr, M. P. Karlsson, and L. M. Frank, “Rapid and continuous modulation of hippocampal network state during exploration of new places,” *PLOS ONE*, vol. 8, no. 9, e73114, Sep. 2, 2013.
- [84] J. Lee, N. P. Bertrand, and C. J. Rozell, “Parallel unbalanced optimal transport regularization for large scale imaging problems,” Aug. 31, 2019. arXiv: 1909.00149 [eess].
- [85] N. P. Bertrand, J. Lee, S. Shapero, and C. J. Rozell, “Infrared search and track with unbalanced optimal transport dynamics regularization,” Sep. 2019, In preparation.
- [86] S. D. Deshpande, M. H. Er, R. Venkateswarlu, and P. Chan, “Max-mean and max-median filters for detection of small targets,” in *Signal and Data Processing of Small Targets 1999*, vol. 3809, International Society for Optics and Photonics, Oct. 4, 1999, pp. 74–84.
- [87] M. Zeng, J. Li, and Z. Peng, “The design of top-hat morphological filter and application to infrared target detection,” *Infrared Physics & Technology*, vol. 48, no. 1, pp. 67–76, Apr. 1, 2006.

- [88] X. Bai, F. Zhou, T. Jin, and Y. Xie, "Infrared small target detection and tracking under the conditions of dim target intensity and clutter background," in *MIPPR 2007: Automatic Target Recognition and Image Analysis; and Multispectral Image Acquisition*, vol. 6786, International Society for Optics and Photonics, Nov. 15, 2007, p. 67862M.
- [89] J.-H. Kim, J.-J. Park, S.-H. Ahn, D. G. Lee, D. Moon, and S.-K. Kim, "Small target detection using morphology and modified Gaussian distance function," *Security and Communication Networks*, vol. 9, no. 6, pp. 555–560, Apr. 1, 2016.
- [90] Y.-Q. Sun, J.-W. Tian, and J. Liu, "Background suppression based-on wavelet transformation to detect infrared target," in *2005 International Conference on Machine Learning and Cybernetics*, vol. 8, Aug. 2005, 4611–4615 Vol. 8.
- [91] T.-W. Bae, F. Zhang, and I.-S. Kweon, "Edge directional 2D LMS filter for infrared small target detection," *Infrared Physics & Technology*, vol. 55, no. 1, pp. 137–145, Jan. 1, 2012.
- [92] B. Zhang, T. Zhang, Z. Cao, and K. Zhang, "Fast new small-target detection algorithm based on a modified partial differential equation in infrared clutter," *Optical Engineering*, vol. 46, no. 10, p. 106 401, Oct. 2007.
- [93] A. Baldacci, M. Diani, and G. Corsini, "Novel background removal algorithm for navy infrared search and track systems," *Optical Engineering*, vol. 40, no. 8, pp. 1729–1735, Aug. 2001.
- [94] K. Qian, H. Zhou, S. Rong, B. Wang, and K. Cheng, "Infrared dim-small target tracking via singular value decomposition and improved kernelized correlation filter," *Infrared Physics & Technology*, vol. 82, pp. 18–27, May 1, 2017.
- [95] X. Wang and T. Zhang, "Clutter-adaptive infrared small target detection in infrared maritime scenarios," *Optical Engineering*, vol. 50, no. 6, p. 067 001, Jun. 2011.
- [96] S. Kim, J. Lee, S. Kim, and J. Lee, "Small infrared target detection by region-adaptive clutter rejection for sea-based Infrared search and track," *Sensors*, vol. 14, no. 7, pp. 13 210–13 242, Jul. 22, 2014.
- [97] X. Zhang, K. Ren, M. Wan, G. Gu, and Q. Chen, "Infrared small target tracking based on sample constrained particle filtering and sparse representation," *Infrared Physics & Technology*, vol. 87, pp. 72–82, Dec. 1, 2017.
- [98] M. Liu, Z. Huang, Z. Fan, S. Zhang, and Y. He, "Infrared dim target detection and tracking based on particle filter," in *2017 36th Chinese Control Conference (CCC)*, Jul. 2017, pp. 5372–5378.

- [99] C. Gao, D. Meng, Y. Yang, Y. Wang, X. Zhou, and A. G. Hauptmann, “Infrared patch-image model for small target detection in a single image,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4996–5009, Dec. 2013.
- [100] H. Wang, J. Lou, C. Zhang, and Y. Liu, “Infrared small dim target detection based on weighted nuclear norm minimization,” in *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, Jul. 2017, pp. 388–393.
- [101] X. Wang, Z. Peng, D. Kong, P. Zhang, and Y. He, “Infrared dim target detection based on total variation regularization and principal component pursuit,” *Image and Vision Computing*, vol. 63, pp. 1–9, Jul. 1, 2017.
- [102] Y.-J. He, M. Li, J. Zhang, and J.-P. Yao, “Infrared target tracking via weighted correlation filter,” *Infrared Physics & Technology*, vol. 73, pp. 103–114, Nov. 1, 2015.
- [103] C. Gao, L. Wang, Y. Xiao, Q. Zhao, and D. Meng, “Infrared small-dim target detection based on Markov random field guided noise modeling,” *Pattern Recognition*, vol. 76, pp. 463–475, Apr. 1, 2018.
- [104] E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust principal component analysis?” *J. ACM*, vol. 58, no. 3, 11:1–11:37, Jun. 2011.
- [105] Z. Zhou, X. Li, J. Wright, E. Candès, and Y. Ma, “Stable principal component pursuit,” in *2010 IEEE International Symposium on Information Theory*, Jun. 2010, pp. 1518–1522.
- [106] A. Chambolle and T. Pock, “A first-order primal-dual algorithm for convex problems with applications to imaging,” *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, May 1, 2011.
- [107] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, Jul. 26, 2011.
- [108] N. Parikh and S. Boyd, “Proximal algorithms,” *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, Jan. 13, 2014.
- [109] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, “RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2233–2246, Nov. 2012.

- [110] J. M. Cathcart and A. D. S. Jr, “Generation and application of high-resolution infrared computer imagery,” *Optical Engineering*, vol. 30, no. 11, pp. 1745–1756, Nov. 1991.
- [111] K. F. Prussing, O. Pierson, C. Cordell, J. Stewart, and K. Nielson, “Enhanced backgrounds in scene rendering with GTSIMS,” in *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXIX*, vol. 10625, International Society for Optics and Photonics, May 14, 2018, 106250O.
- [112] T. Bouwmans, N. S. Aybat, E.-h. Zahzah, N. S. Aybat, and E.-h. Zahzah, *Handbook of Robust Low-Rank and Sparse Matrix Decomposition: Applications in Image and Video Processing*. Chapman and Hall/CRC, Jul. 6, 2016, ISBN: 978-1-4987-2463-0.
- [113] X. Ye, J. Yang, X. Sun, K. Li, C. Hou, and Y. Wang, “Foreground–background separation from video clips via motion-assisted matrix restoration,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 11, pp. 1721–1734, Nov. 2015.
- [114] X. Zhou, C. Yang, and W. Yu, “Moving object detection by detecting contiguous outliers in the low-rank representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 597–610, Mar. 2013.
- [115] A. E. Waters, A. C. Sankaranarayanan, and R. Baraniuk, “SpaRCS: Recovering low-rank and sparse matrices from compressive measurements,” in *Advances in Neural Information Processing Systems 24*, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2011, pp. 1089–1097.
- [116] N. P. Bertrand, A. S. Charles, H. L. Yap, and C. J. Rozell, “Stable filtering for efficient dimensionality reduction of streaming manifold data,” Sep. 2019, In preparation.
- [117] H. L. Yap, M. B. Wakin, and C. J. Rozell, “Stable manifold embeddings with structured random matrices,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 4, pp. 720–730, Aug. 2013.
- [118] D. Kleinfeld, L. Luan, P. P. Mitra, J. T. Robinson, R. Sarpeshkar, K. Shepard, C. Xie, and T. D. Harris, “Can one concurrently record electrical spikes from every neuron in a mammalian brain?” *Neuron*, vol. 103, no. 6, pp. 1005–1015, Sep. 2019.
- [119] H. Hotelling, “Analysis of a complex of statistical variables into principal components,” *Journal of Educational Psychology*, vol. 24, no. 6, pp. 417–441, Sep. 1933.
- [120] D. J. Rezende, S. Mohamed, and D. Wierstra, “Stochastic backpropagation and approximate inference in deep generative models,” in *Proceedings of the 31st International Conference on Machine Learning*, (Beijing, China), E. P. Xing and

- T. Jebara, Eds., ser. *Proceedings of Machine Learning Research*, vol. 32, PMLR, Jun. 22–24, 2014, pp. 1278–1286.
- [121] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in *International Conference on Learning Representations (ICLR) 2014*, Dec. 20, 2013. arXiv: 1312.6114.
- [122] C. Boutsidis, D. Garber, Z. Karnin, and E. Liberty, “Online principal components analysis,” in *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, (San Diego, California), ser. SODA ’15, Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2015, pp. 887–901.
- [123] E. J. Candes and T. Tao, “Near-optimal signal recovery from random projections: Universal encoding strategies?” *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.
- [124] N. Ailon and B. Chazelle, “Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform,” in *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, ser. STOC ’06, New York, NY, USA: ACM, 2006, pp. 557–563, ISBN: 978-1-59593-134-4.
- [125] J. A. Gallego, M. G. Perich, S. N. Naufel, C. Ethier, S. A. Solla, and L. E. Miller, “Cortical population activity within a preserved neural manifold underlies multiple motor behaviors,” *Nature Communications*, vol. 9, no. 1, p. 4233, Oct. 12, 2018.
- [126] J. Nelson, E. Price, and M. Wootters, “New constructions of RIP matrices with fast multiplication and fewer rows,” in *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA ’14, Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2014, pp. 1515–1528, ISBN: 978-1-61197-338-9.
- [127] N. Ailon and H. Rauhut, “Fast and RIP-optimal transforms,” *Discrete & Computational Geometry*, vol. 52, no. 4, pp. 780–798, Dec. 1, 2014.
- [128] W. B. Johnson and J. Lindenstrauss, “Extensions of Lipschitz mappings into a Hilbert space,” *Contemporary mathematics*, vol. 26, no. 189-206, p. 1, 1984.
- [129] N. Ailon and E. Liberty, “An almost optimal unrestricted fast Johnson-Lindenstrauss transform,” *ACM Trans. Algorithms*, vol. 9, no. 3, 21:1–21:12, Jun. 2013.
- [130] R. G. Baraniuk and M. B. Wakin, “Random projections of smooth manifolds,” *Foundations of Computational Mathematics*, vol. 9, no. 1, pp. 51–77, Feb. 1, 2009.

- [131] M. A. Davenport, P. T. Boufounos, M. B. Wakin, and R. G. Baraniuk, “Signal processing with compressive measurements,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 445–460, Apr. 2010.
- [132] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, May 26, 2005, 600 pp., ISBN: 978-1-139-44466-8.
- [133] A. Lapidoth, *A Foundation in Digital Communication*. Cambridge University Press, 2009, ISBN: 978-1-107-17732-1.
- [134] A. Eftekhari and M. B. Wakin, “New analysis of manifold embeddings and signal recovery from compressive measurements,” *Applied and Computational Harmonic Analysis*, vol. 39, no. 1, pp. 67–109, Jul. 1, 2015.
- [135] M. M. Elshrif and E. M. Cherry, “A quantitative comparison of the behavior of human ventricular cardiac electrophysiology models in tissue,” *PLOS ONE*, vol. 9, no. 1, e84401, Jan. 8, 2014.
- [136] K. B. Nooner, S. Colcombe, R. Tobe, M. Mennes, M. Benedict, A. Moreno, L. Panek, S. Brown, S. Zavitz, Q. Li, S. Sikka, D. Gutman, S. Bangaru, R. T. Schlachter, S. Kamiel, A. Anwar, C. Hinz, M. Kaplan, A. Rachlin, S. Adelsberg, B. Cheung, R. Khanuja, C. Yan, C. Craddock, V. Calhoun, W. Courtney, M. King, D. Wood, C. Cox, C. Kelly, A. DiMartino, E. Petkova, P. Reiss, N. Duan, D. Thompsen, B. Biswal, B. Coffey, M. Hoptman, D. C. Javitt, N. Pomara, J. Sidtis, H. Koplewicz, F. X. Castellanos, B. Leventhal, and M. Milham, “The NKI-Rockland sample: A model for accelerating the pace of discovery science in psychiatry,” *Frontiers in Neuroscience*, vol. 6, 2012.
- [137] H. J. V. Zheng, Q. Wang, and G. B. Stanley, “Adaptive shaping of cortical response selectivity in the vibrissa pathway,” *Journal of Neurophysiology*, vol. 113, no. 10, pp. 3850–3865, Mar. 18, 2015.
- [138] W. Denk, J. H. Strickler, and W. W. Webb, “Two-photon laser scanning fluorescence microscopy,” *Science*, vol. 248, no. 4951, pp. 73–76, Apr. 6, 1990. pmid: 2321027.
- [139] D. Smetters, A. Majewska, and R. Yuste, “Detecting action potentials in neuronal populations with calcium imaging,” *Methods*, vol. 18, no. 2, pp. 215–221, Jun. 1, 1999.
- [140] J. J. Jun, N. A. Steinmetz, J. H. Siegle, D. J. Denman, M. Bauza, B. Barbarits, A. K. Lee, C. A. Anastassiou, A. Andrei, Ç. Aydın, M. Barbic, T. J. Blanche, V. Bonin, J. Couto, B. Dutta, S. L. Gratiy, D. A. Gutnisky, M. Häusser, B. Karsh, P. Ledochowitsch, C. M. Lopez, C. Mitelut, S. Musa, M. Okun, M. Pachitariu, J. Putzeys, P. D. Rich, C. Rossant, W.-l. Sun, K. Svoboda, M. Carandini, K. D. Harris, C. Koch, J. O’Keefe, and T. D. Harris, “Fully integrated silicon probes for high-

density recording of neural activity,” *Nature*, vol. 551, no. 7679, pp. 232–236, Nov. 2017.

- [141] H. Yu, P. T. Galwaduge, V. Voleti, K. Patel, W. Li, M. A. Shaik, and E. M. Hillman, “Two-photon swept confocally aligned planar excitation microscopy (2P-SCAPE),” in *Optics in the Life Sciences Congress (2017), Paper NW4C.3*, Optical Society of America, Apr. 2, 2017, NW4C.3.
- [142] A. Song, A. S. Charles, S. A. Koay, J. L. Gauthier, S. Y. Thiberge, J. W. Pillow, and D. W. Tank, “Volumetric two-photon imaging of neurons using stereoscopy (vTwINS),” *Nature Methods*, vol. 14, no. 4, pp. 420–426, Apr. 2017.
- [143] L. Kong, J. Tang, J. P. Little, Y. Yu, T. Lämmermann, C. P. Lin, R. N. Germain, and M. Cui, “Continuous volumetric imaging via an optical phase-locked ultrasound lens,” *Nature Methods*, vol. 12, no. 8, pp. 759–762, Aug. 2015.
- [144] A. J. Majda and A. L. Bertozzi, *Vorticity and Incompressible Flow*. Cambridge University Press, 2002, vol. 27.
- [145] H. Schaeffer, R. Caflisch, C. D. Hauck, and S. Osher, “Sparse dynamics for partial differential equations,” *Proceedings of the National Academy of Sciences*, vol. 110, no. 17, pp. 6634–6639, Apr. 23, 2013. pmid: 23533273.
- [146] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, no. 5500, p. 2323, Dec. 22, 2000.
- [147] J. B. Tenenbaum, V. de Silva, and J. C. Langford, “A global geometric framework for nonlinear dimensionality reduction,” *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 22, 2000. pmid: 11125149.