ADVANCES IN NEAR-FIELD TO BLIND FAR-FIELD PTYCHOGRAPHY, AND COMPRESSED CLASSIFICATION FROM PHASELESS MEASUREMENTS

By

Mark Philip Roach

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Mathematics – Doctor of Philosophy

2023

ABSTRACT

Chapter 1 initially concerns the introduction of Fourier Phase Retrieval. In many imaging systems, one can only measure the magnitude-square of the Fourier transform of the underlying signal, known as the **power spectral density**. At a large enough distance from the imaging plane, the measurements are given by the Fourier transform of the image. Thus for capturing images of large distances, optical devices essentially measure the Fourier transform magnitude of the object being imaged. The problem of reconstructing a signal from its Fourier magnitude is known as **Fourier phase retrieval**. This problem arises in many areas of engineering and applied physics, including optics, X-ray crystallography (determining the atomic and molecular structure of a crystal), astronomical imaging, speech processing, computational biology, etc. Also, in Chapter 1, we introduce the concept of **Dimensionality Reduction** ([98], [25], [58], [61]), which is a tool used in several disciplines, including statistics, data mining, pattern recognition, machine learning, artificial intelligence, and optimization. Dimensionality reduction refers to the act of transforming data from a high-dimensional space to a low-dimensional space.

Chapter 2 discusses **Fourier Ptychography**, which is an imaging technique that involves sample being illuminated at different angles of incidence (effectively shifting the sample's Fourier transform) after which a lens acts as a low-pass filter, thereby effectively providing localized Fourier information about the sample around frequencies dictated by each angle of illumination. **Near-Field (Fourier) Ptychography (NFP)** (see, e.g., [106, 107, 124]) occurs when the sample is placed at a short defocus distance having a large Fresnel number. We prove that certain NFP measurements are robustly invertible (up to an unavoidable global phase ambiguity) for specific Point Spread Functions (PSFs) and physical masks which lead to well-conditioned lifted linear systems. We then apply a block phase retrieval algorithm using weighted angular synchronization and prove that the proposed approach accurately recovers the measured sample for these specific PSF and mask pairs. Finally, we also propose using a Wirtinger Flow for NFP problems and numerically evaluate that alternate approach both against our main proposed approach, as well as with NFP measurements for which our main approach does not apply.

We now move onto Chapter 3 concerning **Blind Ptychography**. **Far-field Ptychography** occurs when there is a large enough defocus distance (when the Fresnel number is $\ll 1$) to obtain magnitude-square Fourier transform measurements. In an attempt to remove ambiguities, masks are utilized to ensure unique outputs to any recovery algorithm are unique up to a global phase. In Chapter 3, we assume that both the sample and the mask are unknown, and we apply blind deconvolutional techniques to solve for both. Numerical experiments demonstrate that the technique works well in practice, and is robust under noise.

Finally, we have Chapter 4. Let $\mathcal{M}$ be a compact d-dimensional submanifold of $\mathbb{R}^N$ with reach τ and volume $V_{\mathcal{M}}$. Fix $\epsilon \in (0, 1)$. In this paper we prove that a nonlinear function $f : \mathbb{R}^N \to \mathbb{R}^m$ exists with $m \leq C\left(d/\epsilon^2\right) \log\left(\frac{\sqrt[d]{V_{\mathcal{M}}}}{\tau}\right)$ such that

$$(1-\epsilon)\|\mathbf{x}-\mathbf{y}\|_2 \leq \|f(\mathbf{x})-f(\mathbf{y})\|_2 \leq (1+\epsilon)\|\mathbf{x}-\mathbf{y}\|_2$$

holds for all $\mathbf{x} \in \mathcal{M}$ and $\mathbf{y} \in \mathbb{R}^N$. In effect, f not only serves as a bi-Lipschitz function from $\mathcal{M}$ into $\mathbb{R}^m$ with bi-Lipschitz constants close to one, but also approximately preserves all distances from points not in $\mathcal{M}$ to all points in $\mathcal{M}$ in its image. Furthermore, the proof is constructive and yields an algorithm which works well in practice. In particular, it is empirically demonstrated herein that such nonlinear functions allow for more accurate compressive nearest neighbor classification than standard linear Johnson-Lindenstrauss embeddings do in practice.

Dedicated to Holly (2004-2018),
Gypsy (2007-2021),
and Sparkle (2006-2022).

ACKNOWLEDGEMENTS

I would like to thank Mark Iwen for his guidance and support.I would like to thank Guoan Zheng for answering questions about (and providing code for) his work in [124]. I would like to thank Michael Perlmutter for his work and collaboration on the Near-field Ptychography chapter.

My work was supported in part by NSF DMS 1912706.

TABLE OF CONTENTS

KEY TO ABBREVIATIONS

Ptychography Notation

- Let $\mathbf{x}, \mathbf{m} \in \mathbb{C}^d$ denote the specimen and mask
- **Far-field Ptychographic Measurements**: $\left| \sum_{n=0}^{d-1} x_n m_{n-\ell} e^{-2\pi i n k/d} \right|^2$
- **Point Spread Function (PSF)**: $\mathbf{p} \in \mathbb{C}^d$
- **Near-field Ptychographic Measurements**: $|(\mathbf{p} * (S_k \mathbf{m} \circ \mathbf{x}))_\ell|^2$,
- **Indexing**: $[d] = \{1, 2, \ldots, d\}, [d]_0 = \{0, 1, \ldots, d-1\}$
- **Support**: $supp(\mathbf{x}) := \{n \in [d]_0 \mid x_n \neq 0\}$
- **Fourier Transform**: $\mathbf{F}_d$ - $d \times d$ DFT matrix
- **Reversal**: $\tilde{x}_n := x_{-n \bmod d}, \quad \forall n \in [d]_0$
- **Shift Operator**: $(S_\ell \mathbf{x})_n = x_{(\ell+n) \bmod d}, \quad \forall n \in [d]_0$
- **Circular Convolution** : $(\mathbf{x} *_d \mathbf{y})_\ell := \sum_{n=0}^{d-1} x_n y_{(\ell-n) \bmod d}$
- **Hadamard Product**: $(\mathbf{x} \circ \mathbf{y})_\ell := x_\ell y_\ell$
- **Decoupling Lemma**: $\left((\mathbf{x} \circ S_{-\ell}\mathbf{y}) *_d (\bar{\tilde{\mathbf{x}}} \circ S_\ell \bar{\tilde{\mathbf{y}}})\right)_k = \left((\mathbf{x} \circ S_{-k}\bar{\mathbf{x}}) *_d (\tilde{y} \circ S_k \bar{\tilde{\mathbf{y}}})\right)_\ell$

Blind Deconvolution Notation

- $\bar{\mathbf{A}}$ - complex Gaussian matrix, $\bar{A}_{ij} \sim \mathcal{N}(0, 1/2) + i\mathcal{N}(0, 1/2)$
- $\mathbf{a}_\ell$ - ℓ-th column of $\mathbf{A}^*$
- $\mathbf{B}$ - first K columns of $\mathbf{F}_d$, $\mathbf{b}_\ell$ - ℓ-th column of $\mathbf{B}^*$
- $\mathbf{e}$ - complex Gaussian vector, $\mathbf{e} \sim \mathcal{N}(0, \frac{\sigma^2 L_0^2}{2d} I_d) + i\mathcal{N}(0, \frac{\sigma^2 L_0^2}{2d} I_d)$
- $\mathcal{A}$ - linear operator: $\mathcal{A}(Z) := \{\mathbf{b}_\ell^* Z \mathbf{a}_\ell\}_{\ell=1}^d \in \mathbb{C}^{d \times 1}$
- $\mathcal{A}^*$ - adjoint linear operator: $\mathcal{A}^*(z) := \sum_{\ell=1}^d z_\ell \mathbf{b}_\ell \mathbf{a}_\ell^* \in \mathbb{C}^{K \times N}$
- $(\mathbf{h}_0, \mathbf{x}_0)$ - underlying truth, $y = \mathcal{A}(\mathbf{h}_0 \mathbf{x}_0^*) + \mathbf{e}$, $L_0 = \|\mathbf{h}_0\|^2 = \|\mathbf{x}_0\|^2$
- $(\mathbf{u}_0, \mathbf{v}_0)$ - initial estimate, $(\mathbf{u}_t, \mathbf{v}_t)$ - estimate during gradient descent
- $(\mathbf{h}, \mathbf{x})$ - obtained estimate after Algorithm, $\quad L = \|\mathbf{h}\| \cdot \|\mathbf{x}\|$
- $\delta = \delta(\mathbf{z}) = \delta(\mathbf{h}, \mathbf{x}) := \frac{\|\mathbf{h}\mathbf{x}^* - \mathbf{h}_0 \mathbf{x}_0^*\|_F}{L_0}$, μ_h - incoherence, $\mu_h^2 = \frac{L \|\mathbf{B}\mathbf{h}_0\|_\infty^2}{\|\mathbf{h}_0\|^2}$

- $N_{L_0} := \{(\mathbf{h}, \mathbf{x}) \mid \|\mathbf{h}\| \leq 2\sqrt{L_0}, \|\mathbf{x}\| \leq 2\sqrt{L_0}\}$
- $N_\mu := \{\mathbf{h} \mid \sqrt{d}\|\mathbf{B}\mathbf{h}\|_\infty \leq 4\sqrt{L_0}\mu\}, \quad \mu_h \leq \mu$
- $N_\epsilon := \{(\mathbf{h}, \mathbf{x}) \mid \|\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*\|_F \leq \epsilon L_0\}, \quad 0 < \epsilon \leq \frac{1}{15}$
- $N_{\tilde{F}} = \{(\mathbf{h}, \mathbf{x}) \mid \tilde{F}(\mathbf{h}, \mathbf{x}) \leq \frac{1}{3}\epsilon^2 L_0^2 + \|\mathbf{e}\|^2\}$
- $F(\mathbf{h}, \mathbf{x}) := \|\mathcal{A}(\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*) + \mathbf{e}\|^2, \quad F_0(\mathbf{h}, \mathbf{x}) := \|\mathcal{A}(\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*)\|^2$
- $F(\mathbf{h}, \mathbf{x}) = \|\mathbf{e}\|^2 + F_0(\mathbf{h}, \mathbf{x}) - 2Re(\langle \mathcal{A}^*(\mathbf{e}), \mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^* \rangle)$
- $G_0(\mathbf{z}) := \max\{z - 1, 0\}^2 = [z - 1]_+^2, \quad \rho \geq d^2 + 2\|\mathbf{e}\|^2$
- $G(\mathbf{h}, \mathbf{x}) := \rho\Big[G_0\Big(\frac{\|\mathbf{h}\|^2}{2L}\Big) + G_0\Big(\frac{\|\mathbf{x}\|^2}{2L}\Big) + \sum_{\ell=1}^d G_0\Big(\frac{d|\mathbf{b}_\ell^*\mathbf{h}|^2}{8L\mu^2}\Big)\Big]$
- $\tilde{F}(\mathbf{h}, \mathbf{x}) := F(\mathbf{h}, \mathbf{x}) + G(\mathbf{h}, \mathbf{x})$

Terminal Embeddings Notation

- **Tube**: $tube(\delta, \mathcal{M}) := \{\mathbf{x} \mid \exists \mathbf{y} \in \mathcal{M} \ \ with \ \|\mathbf{x} - \mathbf{y}\|_2 \leq \delta\}$
- **Euclidean Ball**: $B_{\ell^2}^N(\mathbf{x}, \gamma) := \{\mathbf{y} \in \mathbb{R}^N \mid \|\mathbf{x} - \mathbf{y}\|_2 < \gamma\}$
- $-S := \{-\mathbf{x} \mid \mathbf{x} \in S\}$, $S \pm S := \{\mathbf{x} \pm \mathbf{y} \mid \mathbf{x}, \mathbf{y} \in S\}$, $U(\mathbf{x}) := \mathbf{x}/\|\mathbf{x}\|_2$
- **Unit Secants**: $S_T := \overline{U\left((T - T) \setminus \{\mathbf{0}\}\right)} = \overline{\left\{\frac{\mathbf{x}-\mathbf{y}}{\|\mathbf{x}-\mathbf{y}\|_2} \mid \mathbf{x}, \mathbf{y} \in T, \mathbf{x} \neq \mathbf{y}\right\}}$.
- **ϵ-JL map**: $A \in \mathbb{C}^{m \times N}$, $T \subset \mathbb{R}^N$ into $\mathbb{C}^m$, $(1 - \epsilon)\|\mathbf{x}\|_2^2 \leq \|A\mathbf{x}\|_2^2 \leq (1 + \epsilon)\|\mathbf{x}\|_2^2$
- **ϵ-JL embedding**: $A \in \mathbb{C}^{m \times n}$ ϵ-JL map of $T - T := \{\mathbf{x} - \mathbf{y} \mid \mathbf{x}, \mathbf{y} \in T\}$
- **Radius**: $rad(T) := \sup_{x \in T} \|\mathbf{x}\|_2$
- **Diameter**: $diam(T) := rad(T - T) = \sup_{x,y \in T} \|\mathbf{x} - \mathbf{y}\|_2$,
- **δ-cover**: $\delta \in \mathbb{R}^+$, $S \subset T$, $\forall \mathbf{x} \in T$, $\exists \mathbf{y} \in S$ such that $\|\mathbf{x} - \mathbf{y}\|_2 \leq \delta$.
- **δ-Covering Number**: $\mathcal{N}(T, \delta) \in \mathbb{N}$, smallest achievable cardinality of a δ-cover of T
- **Gaussian Width**: $w(T) := \mathbb{E} \sup_{x \in T} \langle \mathbf{g}, \mathbf{x} \rangle$, $\mathbf{g} \in \mathbb{R}^N$, i.i.d. $\mu = 0, \sigma^2 = 1$, Gaussian entries
- **Reach**: $S \subset \mathbb{R}^N$, $\tau_S := \sup\{t \geq 0 \mid \forall \mathbf{x} \in \mathbb{R}^n, d(\mathbf{x}, S) < t, \ \mathbf{x} \ \exists \text{ unique closest point in } S\}$
- **Convex Hull**: $S \subset \mathbb{C}^N$, $\text{conv}(S) := \bigcup_{j=1}^\infty \left\{\sum_{\ell=1}^j \alpha_\ell \mathbf{x}_\ell \mid \mathbf{x}_\ell \in S, \ \alpha_\ell \in [0, 1], \sum_{\ell=1}^j \alpha_\ell = 1\right\}$
- **ϵ-Convex Hull Distortion**: $S \subset \mathbb{R}^N$, $|\|\Phi\mathbf{x}\|_2 - \|\mathbf{x}\|_2| \leq \epsilon$, $\forall \mathbf{x} \in \text{conv}(S)$

CHAPTER 1

INTRODUCTION TO PHASE RETRIEVAL AND DIMENSIONALITY REDUCTION

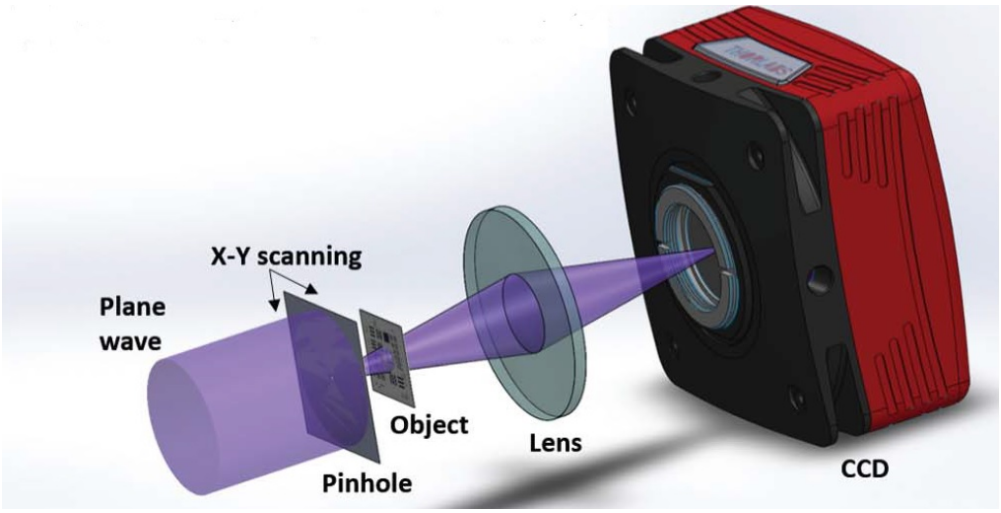

Figure 1.1 [99] An illustration of a conventional ptychography setup

1.1 Introduction

In many imaging systems, one can only measure the magnitude-square of the Fourier transform of the underlying signal, known as the **power spectral density**. For example, in an optical setting, detection devices like CCD (**charge-coupled device**) cameras and photosensitive films cannot measure the phase of a light wave only, instead measuring the photon flux (number of photons per second per unit area).

At a large enough distance from the imaging plane, the measurements are given by the Fourier transform of the image. Thus for capturing images of large distances, optical devices essentially measure the Fourier transform magnitude of the object being imaged. However, structural content about the image is contained in the phase, so this important information is lost. The problem of reconstructing a signal from its Fourier magnitude is known as **Fourier phase retrieval**. This problem arises in many areas of engineering and applied physics, including optics ([56], [36]), X-ray crystallography ([65],[120],[102],[24]), astronomical imaging ([115],[93],[105]), speech process-

ing ([90], [42]) and computational biology, just to name a few.

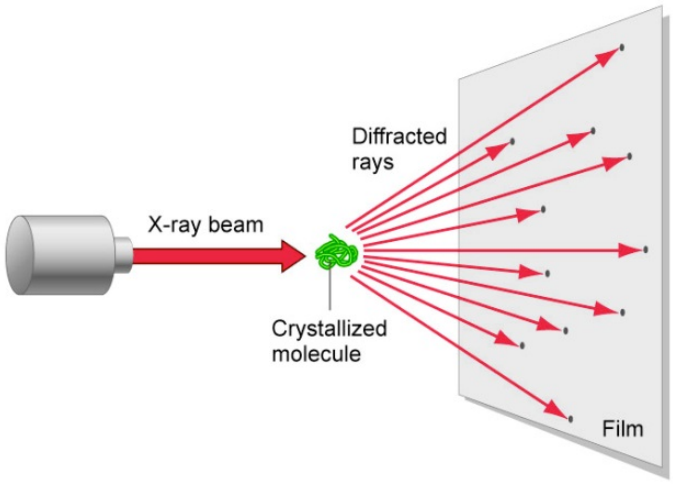

Figure 1.2 [92] In x-ray crystallography, the goal is to gain an image of the positions of atoms within a molecule by illuminating a crystallized sample with x-rays. The molecular structure is then deduced from the pattern of the radiation diffracted by the sample.

1.2 Phase Retrieval Problem

To mathematically solve the phase retrieval problem, we first focus on the discretized one-dimensional setting, which can the be generalized.

Definition 1.2.1. ***(Classical Phase Retrieval Problem)*** *Let* $\mathbf{x} \in \mathbb{C}^d$ *be the underlying signal we wish to recover. In Fourier phase retrieval, the measurements are given by*

$$y_k = \left| \sum_{n=0}^{d-1} x_n e^{-2\pi i k n/N} \right|^2, \quad k \in [N]_0, N = 2d - 1 \tag{1.1}$$

Here we are over-sampling by a factor of two by twice the length of the signal. Our goal is to recover $\mathbf{x}$.

There are many challenges involved in solving the phase retrieval problem, as discussed in [7]. First among these, is the fact that the true signal $\mathbf{x} \in \mathbb{C}^d$ cannot be recovered uniquely. For instance, the rotation, translation, or conjugate reflection do not modify the Fourier magnitudes. Without additional constraints, the unknown signal will only be determined up to what are called *classical*

ambiguities or *unavoidable trivial ambiguities*, which may not be of concern depending on the application.

There are also non-trivial ambiguities for the classical phase retrieval problem. For example

$$x_1 = (1, 0, -2, 0, -2)^T, \quad x_2 = ((1 - \sqrt{3}), 0, 1, 0, (1 + \sqrt{3}))^T \tag{1.2}$$

yield the same Fourier magnitudes y_k. We wish to categorise the number of non-trivial solutions, by exploring the relationship between the Fourier magnitudes and the autocorrelation measurements.

Definition 1.2.2. *Let* $\mathbf{x} \in \mathbb{C}^d$ *be the underlying signal with* $supp(x) \subseteq [d]_0$. *We define the* ***autocorrelation measurements*** *by*

$$a_m = \sum_{n=0}^{d} \bar{x}_n x_{n+m}, \quad -N + 1 \leq m \leq N - 1 \tag{1.3}$$

We consider the product of the polynomial $X(z) = \sum_{n=0}^{d-1} x_n z^n$ and the reversed polynomial $\tilde{X}(z) = z^{d-1}\bar{X}(z^{-1})$, where $\bar{X}$ denote the polynomial with conjugate coefficients. Assuming that $x[0], x[d-1] \neq 0$, we have that

$$X(z)\tilde{X}(z) = \sum_{n=0}^{d-1} x_n z^n \cdot z^{d-1} \sum_{\ell=0}^{d-1} \bar{x}_\ell z^{-\ell} = \sum_{n=0}^{2d-2} a_{n-d+1} z^n =: A(z) \tag{1.4}$$

where $A(z)$ is the autocorrelation polynomial of degree $2d - 2$. We can then rewrite the Fourier magnitude measurements

$$y_k = e^{2\pi ik(d-1)/N} X(e^{-2\pi ik/N})\tilde{X}(e^{-2\pi ik/N}) = e^{2\pi i(d-1)/N} A(e^{-2\pi ik/N}) \tag{1.5}$$

so that the autocorrelation polynomial is completely determined by the 2d - 1 samples y_k. The phase retrieval problem is thus equivalent to the recovery of $X(z)$ from $A(z) = X(z)\tilde{X}(z)$.

Comparing the roots of $X(z)$ and $\tilde{X}(z)$, we note that the roots of $A(z)$ occur in reflected pairs $(\gamma_j, \bar{\gamma}_j^{-1})$ with respect to the unit circle. The main problem in the recovery of $X(z)$ is deciding

whether γ_j or $\bar{\gamma}_j^{-1}$ is a root of $X(z)$. In [6], this approach is used to show that the number of non-trivial solutions is therefore bounded by 2^{d-2}.

1.2.1 Phase Retrieval Using Masks

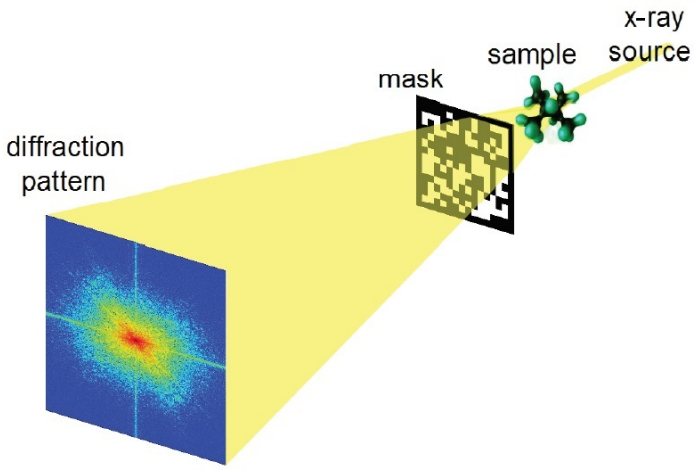

Figure 1.3 [12] An illustration of masked phase retrieval setup

We can adapt the phase retrieval problem by using masks, to eliminate some of the trivial and non-trivial ambiguities. There are several methods to applying this, some of which are listed below:
(i) **Masking**: The phase front after the sample is modified by the use of a mask or a phase plate;
(ii) **Diffraction grating**: The illuminating beam is modulated by the use of optical gratings;
(iii) **Oblique illuminations**: The illuminating beam is modulated to hit the sample at specific angles.

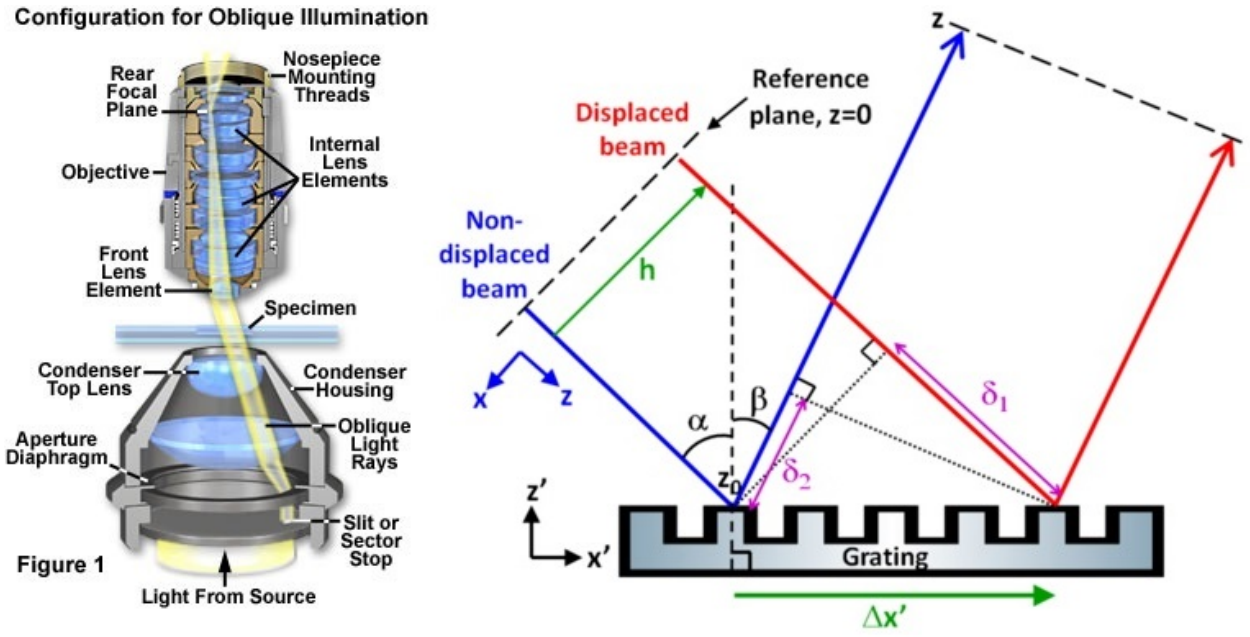

Figure 1.4 On the left ([86]) is an example of oblique illuminations, and on the right ([73]) is a diffraction grating which displaces the angle of the beam

The main area of research involving masked phase retrieval is split in two sectors, looking at random masks or deterministic masks (i.e. masks that have been specifically chosen).

Definition 1.2.3. ***(Phase Retrieval Using Masks)*** *Let* $\mathbf{x} \in \mathbb{C}^d$ *denote the signal,* $\{\mathbf{m}_\ell \mid \mathbf{m}_\ell \in \mathbb{C}^d, \ell \in [L]_0\}$, $L \geq 2$ *denote the collection of masks. The masked phase retrieval measurements will be of the form*

$$y_{\ell,k} = \left| \sum_{n=0}^{d-1} x_n [\mathbf{m}_\ell]_n e^{-2\pi i n k/d} \right|^2, \quad k \in [d]_0, \ell \in [L]_0. \tag{1.6}$$

The goal is then to recover $\mathbf{x}$. A further study of research is blind phase retrieval ([97], [1], [16], [17]), in which both the signal and mask are unknown, although some information may be known.

In Chapters 2 and 3, we look at types of masked phase retrieval, with Chapter 3 in particular, looking at the blind variant. In both situations, the set of masks are generated by a shift operator, which we will discuss more in the next section.

There is a constraint to the type of signal that we can recover.

Definition 1.2.4. *A signal* $\mathbf{x}$ *is said to be* ***non-vanishing*** *if* $x_n \neq 0$ *for each* $n \in [d]$.

In [54], deterministic masks are considered instead of random masks and they show that two masks are sufficient for the convex relaxation of the problem to uniquely recover non-vanishing signals up to a global phase when over-sampled by a factor of two.

1.2.2 PhaseLift

In [12], the classical phase retrieval problem is reformulated as a matrix completion problem. First, we need to define the Fourier transform of a vector.

Definition 1.2.5. *The* ***Fourier transform*** *of* $\mathbf{x} \in \mathbb{C}^d$*, denoted* $\hat{\mathbf{x}} \in \mathbb{C}^d$*, is defined component-wise via*

$$\hat{x}_k := (\mathbf{F}_d \mathbf{x})_k = \sum_{n=0}^{d-1} x_n e^{-2\pi i n k/d} \tag{1.7}$$

where $\mathbf{F}_d \in \mathbb{C}^{d\times d}$ *denotes the* $d \times d$ ***discrete Fourier transform (DFT) matrix*** *with entries*

$$(\mathbf{F}_d)_{\ell,k} = e^{-2\pi i \ell k/d}, \quad \forall (\ell, k) \in [d]_0 \times [d]_0 \tag{1.8}$$

Remark 1.2.1. *With this definition, we can rewrite the masked phase retrieval problem as*

$$y_{\ell,k} = |(\mathbf{F}_d(x \circ \mathbf{m}_\ell))_k|^2, \quad k \in [d]_0, \ell \in [L]_0. \tag{1.9}$$

Let $\mathbf{X} = \mathbf{x}\mathbf{x}^*$. For $\ell \in [L]_0$, let $\mathbf{D}_\ell$ be the diagonal matrix with the mask $\mathbf{m}_\ell$ on the diagonal and let $\mathbf{f}_k^*$ be the rows of the DFT matrix. We then have that the measurements in 1.6 can be written as

$$y_{\ell,k} = |\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}|^2, \quad k \in [d]_0, \ell \in [L]_0. \tag{1.10}$$

Let $\mathcal{A} : \mathcal{S}^{n\times n} \longrightarrow \mathbb{R}^{dL}$ denote the linear operator with entries given by

$$\{\mathcal{A}(\mathbf{X})\}_{\ell,k} = tr(\mathbf{x}^* \mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{D}_\ell \mathbf{f}_k \mathbf{x}) = tr(\mathbf{D}_\ell \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{X}), \tag{1.11}$$

where $\mathcal{S}^{n\times n}$ is the space of self-adjoint matrices. Then the phase retrieval problem can be formulated as

$$\text{Find} \quad \mathbf{X}, \qquad \text{subject to} \quad \mathcal{A}(\mathbf{X}) = \mathbf{y}, \mathbf{X} \succeq 0, \text{rank}(\mathbf{X}) = 1. \tag{1.12}$$

When the measurements in 1.6 are injective, then this is equivalent to

$$\text{minimize} \quad \text{rank}(\mathbf{X}), \qquad \text{subject to} \quad \mathcal{A}(\mathbf{X}) = \mathbf{y}, \mathbf{X} \succeq 0 \tag{1.13}$$

Due to the complexities of solving this problem, **PhaseLift** (Section 2.3, [12]) solves the convex surrogate, giving the semi-definite program

$$\text{minimize} \quad \text{Trace}(\mathbf{X}), \qquad \text{subject to} \quad \mathcal{A}(\mathbf{X}) = \mathbf{y}, \mathbf{X} \succeq 0 \tag{1.14}$$

The result will follow from looking at random masks, where the diagonal matrices $\mathbf{D}_\ell$ for $\ell \in [L]_0$ are i.i.d copies of a matrix $\mathbf{D}$, whose entries are i.i.d copies of a random variable p. These are known as **coded diffraction patterns**. It is shown in [13], that the solution to the convex relaxation is exact, with high probability, provided that we have sufficiently many coded diffraction patterns. It is further shown in the theorem below that the feasible set of solutions is given by

$$\{\mathbf{X} : \mathbf{X} \succeq 0, \mathcal{A}(\mathbf{X}) = \mathbf{y}\} = \{\mathbf{x}\mathbf{x}^*\} \tag{1.15}$$

Before we get to the result, we need a restriction on the random variable p which will allow us to recover $\mathbf{x}$.

Definition 1.2.6. *We say that p is **admissible** if*

(i) p is symmetric;

(ii) $|p| \leq N$;

(iii) $\mathbb{E}p = \mathbb{E}p^2 = 0$;

(iv) $\mathbb{E}|p|^4 = 2\mathbb{E}|p|^2$.

We can now state the theorem for recovering **x**.

Theorem 1.2.1. *(Theorem 1.1, [13]) Suppose that the modulation is admissible (i.e. the random masks are generated from an admissible random variable) and that the number L of coded diffraction patterns obeys* $L \geq c\gamma log^4 d$*, for some fixed numerical constant c. Then with probability at least* $1 - n^{-\gamma}$*, the set of solutions to the convex relaxation reduces to* $\mathbf{x}\mathbf{x}^*$*, and we thus recover* **x** *up to a global phase.*

In practice, physically generating these random masks with known entries is a hard task. In many settings, deterministic masks are more practical to use and used more in real world situations. The next section deals with these masks, in particular taking one mask and shifting it to generate a set of masks.

1.2.3 STFT Phase Retrieval

STFT phase retrieval is similar to masked phase retrieval in that it looks at partially blocking the signal measurements so that you can attempt to recover the signal at a more local level. The key idea is to introduce redundancy in the magnitude-only measurements by maintaining a substantial overlap between adjacent short-time sections of shifted masked measurements. In effect, we are taking a window and shifting it over time. This could involve physically shifting the specimen/window itself, or one could shift the beam to focus on a different local area of the specimen. First, we define the shift of a vector.

Definition 1.2.7. *Given* $\ell \in [d]_0$*, denote the* ***circulant shift operator*** $S_\ell : \mathbb{C}^d \longrightarrow \mathbb{C}^d$ *component-wise via*

$$(S_\ell \mathbf{x})_n = x_{(\ell+n) \bmod d}, \quad \forall n \in [d]_0 \tag{1.16}$$

Now we can introduce the STFT phase retrieval problem.

Definition 1.2.8. ***(STFT Phase Retrieval Problem)*** *Let* $\mathbf{x} \in \mathbb{C}^d, \mathbf{w} \in \mathbb{C}^d$ *denote the signal and window respectively. Let* $\mathbf{m}_\ell = S_\ell \mathbf{w}$ *denote the* ℓ*-shift of the window for* $\ell \in [L]_0$*. The STFT magnitude measurements will be of the form of the masked measurements from 1.6, where each of the masks is a shift of the original mask or window. Our goal is to recover* $\mathbf{x}$*.*

In a similar manner as to before, we say that a window $\mathbf{w}$ is **non-vanishing** if $w_n \neq 0$ for each $n \in [d]$. In [55], it is shown that up to a set of measure zero, non-vanishing signals are uniquely identifiable from their STFT magnitude measurements, up to a global phase, if the support of the signal is contained inside the support of the window, and as long as adjacent short-time sections overlap by any amount. In other research ([81], [28]), it is shown that all non-vanishing signals are uniquely identifiable from their STFT magnitude measurements, up to a global phase, for specific choices of $\mathbf{w}$ and L.

In Chapters 2 and 3, we will explore a couple of ptychographic phase retrieval problems, which can be modeled as an STFT phase retrieval problems.

1.2.4 Noise

In phase retrieval, noise refers to how a signal can be modified in a way that alters the final result. This occurs at all stages of the system, from capture and storage, to processing and transmission. Any real world system is affected by some level of noise. In analog photography or video capture, noise can come in the way of film grain, which is caused by the developing process of silver halide crystals dispersed in photographic emulsion ([39],[85]). In digital photography, noise can come in the way of compression artifacts that occur when the file is compressed to reduce file size ([23]). Background noise is a common occurrence in audio capture. In astronomy, it may result from cosmic background radiation, which is a faint glow of light occurring as a result of remnants from the Big Bang ([119], [8], [101]).

Figure 1.5 [96] Example of an image (left) in which a replication of film grain has been digitally added (right).

Additive White Gaussian Noise (AWGN) or simply **additive noise** is the basic noise model which will be applied in Chapters 2 and 3. This model assumes that for all our phase retrieval models, there will be added unknown Gaussian noise which forms part of the collected measurements, i.e.

$$\mathbf{Y} = \mathbf{X} + \mathbf{N} \tag{1.17}$$

where $\mathbf{X}$ are the "true" measurements, and $\mathbf{N}$ is the additive Gaussian noise. Both $\mathbf{X}$ and $\mathbf{N}$ are then assumed to be unknown with $\mathbf{Y}$ being the known collected measurements. For example, in the masked phase retrieval model, we would have that

$$y_{\ell,k} = |(\mathbf{F}_d(x \circ \mathbf{m}_\ell))_k|^2 + N_{k,\ell}, \quad k \in [d]_0, \ell \in [L]_0. \tag{1.18}$$

where $N \in \mathbb{C}^{d \times L}$ has complex Gaussian entries.

Although the noise is unknown, what can be modelled is the recovery of the signal against varying levels of the noise, with the noise level being measured as relative to the signal. This can be measured via the **signal-to-noise ratio** (**SNR**), which is the ratio of the power of the signal, P_s,

to the power of the noise, P_n. That is, $SNR = \dfrac{P_s}{P_n}$. Thus we have that the higher the signal-to-noise ratio, the less presence of noise relative to the signal. Typically, it is measured in decibels using a base 10 logarithm, that is

$$SNR_{db} := 10\log_{10}(SNR) = 10\log_{10}\left(\frac{P_s}{P_n}\right). \tag{1.19}$$

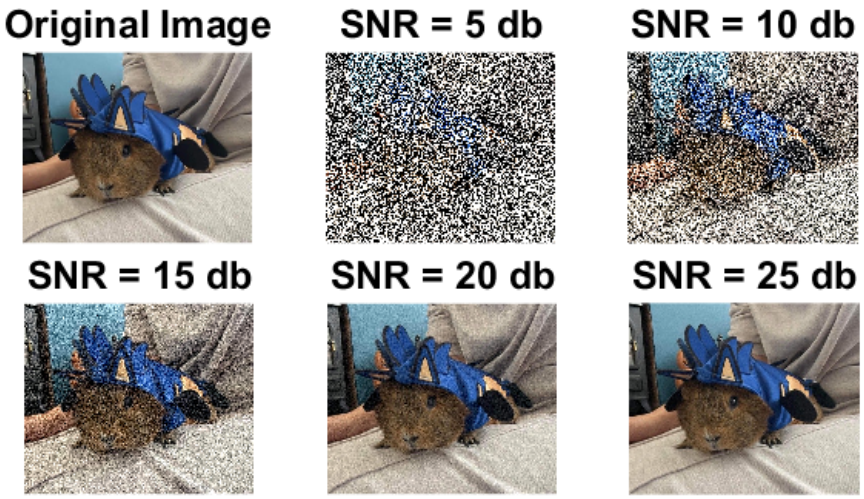

Figure 1.6 [96] Example of an image with varying levels of Gaussian noise applied.

1.2.5 Condition Number of a Matrix

In Chapter 2, we demonstrate a method for solving a phase retrieval problem involving rewriting the measurements as a matrix multiplication, for which we then invert the generated matrix. However, the presence of noise can affect this approach. To measure how much noise can effect the outcome, we require the following definition.

Definition 1.2.9. *The **condition number** of a matrix* $\mathbf{A}$ *is defined by*

$$\kappa = \kappa(\mathbf{A}) := \|\mathbf{A}\|\cdot\|\mathbf{A}^{-1}\| \tag{1.20}$$

where $\|\cdot\|$ is the operator norm. In particular, if $\|\cdot\|$ is the ℓ^2 norm, then

$$\kappa(\mathbf{A}) := \frac{\sigma_{max}(A)}{\sigma_{min}(A)} \tag{1.21}$$

where $\sigma_{max}(A), \sigma_{min}(A)$ are the maximal and minimal singular values respectively.

A matrix with a low condition number is said to be **well-conditioned**, while a matrix with a high condition number is said to be **ill-conditioned**. We apply these same definitions to the problem or system involving matrices i.e. a system is well-conditioned if the resultant matrix is well-conditioned.

Informally, the condition number measures how close a matrix is to being non-invertible. In practice, it measures the effect of perturbations. In our context, this perturbation will be the additive noise. To see where the condition number comes into play, suppose we have the system $\mathbf{y} = \mathbf{Ax}$ which has been effected by noise such that

$$\mathbf{y} + \delta\mathbf{y} = \mathbf{Ax} + \delta\mathbf{Ax} = \mathbf{A}(\mathbf{x} + \delta\mathbf{x}) \tag{1.22}$$

where $\delta\mathbf{y}$ is the noise which is relatively small compared to $\mathbf{y}$ (that is, it has a relatively large $SNR = \frac{\|\mathbf{y}\|}{\|\delta\mathbf{y}\|}$). Then we have that

$$\|\mathbf{y}\| = \|\mathbf{Ax}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{x}\| \Rightarrow \frac{1}{\|\mathbf{x}\|} \leq \frac{\|\mathbf{A}\|}{\|\mathbf{y}\|}, \tag{1.23}$$

and

$$\|\delta\mathbf{x}\| = \|\mathbf{A}^{-1}\delta\mathbf{y}\| \leq \|\mathbf{A}^{-1}\| \cdot \|\delta\mathbf{y}\|. \tag{1.24}$$

Thus combining these inequalities, we get that the relative error between the noise distributed part

of the signal, $\delta\mathbf{x}$, and the true signal, $\mathbf{x}$, is given by

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \|\mathbf{A}\|\cdot\|\mathbf{A}^{-1}\|\cdot\frac{\|\delta\mathbf{y}\|}{\|\mathbf{y}\|} = \frac{\kappa}{SNR}. \tag{1.25}$$

Thus our hope for a successful recovery, at least to a given margin of error, relies on SNR being relatively large, and κ being relatively small.

In Chapter 2, Lemma 2.3.2, we demonstrate a choice of matrix which allows for a well-conditioned system, and thus successful recovery of the signal. This is ensured by utilizing the bound of the condition number generated from [53], in which the maximal singular value is upper bounded whilst the minimal singular value is lower bounded.

1.3 Dimensionality Reduction

Dimensionality reduction ([98], [25], [58], [61]) is a general data analysis tool used in several disciplines, including statistics, data mining, pattern recognition, machine learning, artificial intelligence, and optimization. Dimensionality reduction refers to the act of transforming data from a high-dimensional space to a low-dimensional space. The goal is to be able to remove irrelevant or redundant data, and to be able to reduce the computational cost while still retaining meaningful properties of the original data. Dimension in this context can refer to many things, such as attributes, variables, features, pixels, etc.

Each application utilizes different dimension reduction techniques. In pattern recognition for example, the problem of dimensionality reduction is to extract a subset of features that recovers most of the variability of the data. In text mining, the problem is defined as selecting a subset of words or terms.

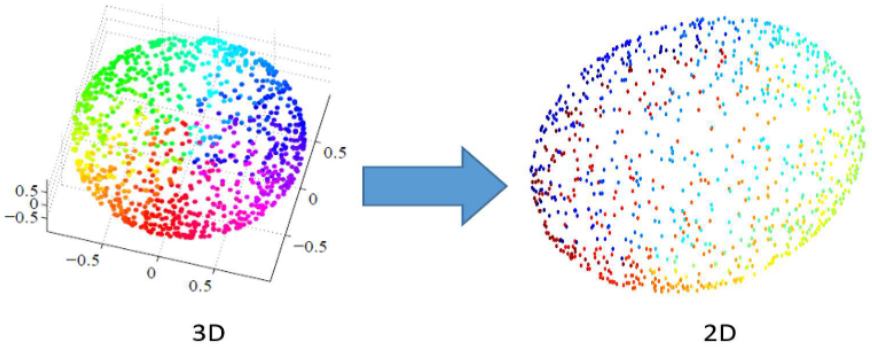

Figure 1.7 [96] Example of dimensionality reduction effect over a 3-dimensional spherical shell manifold. The resulting 2-dimensional embedded data is the attempt to unfold the original data.

1.3.1 k-Nearest Neighbors Classification

Suppose we vectorize training data in d-dimensions in such a way that the concept of distance between two points in $\mathbb{R}^d$ makes sense within the problem, and that data is classified into multiple different classes. The goal of the k-nearest neighbours algorithm (k-NN) is to identify a new image by applying the same vectorization and classifying based on the preset classification of its k-nearest neighbours in $\mathbb{R}^d$. Generally, k is chosen by testing and evaluating the results for different values.

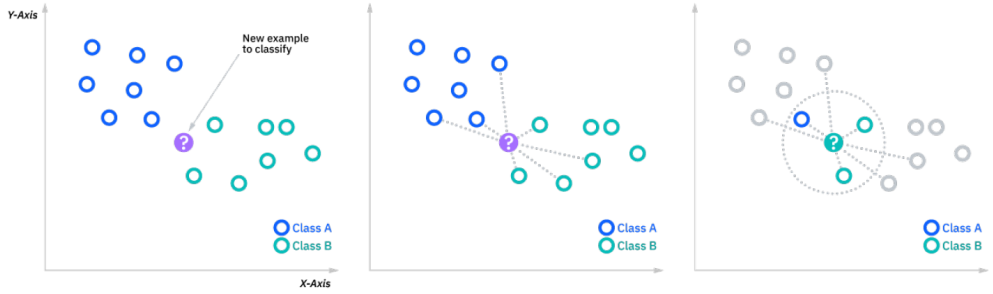

Figure 1.8 [48] Example of the k-nearest neighbors algorithm being applied on embedded data. The left figure demonstrates the data that is being attempted to be categorized. The central and right figure demonstrates the kNN algorithm being applied with $k = 3$.

1.3.2 Johnson-Lindenstrauss Lemma

Algorithms, for example, performing k-nearest neighbour classification, can be computational expensive in high dimensions. Thus it is often advantageous to reduce the dimension of the data

before carrying out such tasks. To ensure that applying dimensionality reduction does not highly distort the data, we will utilize the Johnson-Lindenstrauss lemma.

Lemma 1.3.1 (Johnson-Lindenstrauss Lemma)**.** *Let* $\epsilon \in (0,1)$ *and* $X \subset \mathbb{R}^d$ *be arbitrary with* $|X| = n > 1$. *There exists* $f : X \longrightarrow \mathbb{R}^m$ *with* $m = O(\epsilon^{-2} \log n)$ *such that* $\forall x \in X, \forall y \in X$

$$\|x - y\|_2 \leq \|f(x) - f(y)\|_2 \leq (1 + \epsilon)\|x - y\|_2. \tag{1.26}$$

Let X denote the set of training data in $\mathbb{R}^d$. Then Lemma 1.3.1 states that the distance between the training data will be preserved up to a small distortion. However, in order to successfully apply, for example, k-NN methods, we would prefer a stronger guarantee that the distances from points in the training data be approximately preserved to any point in $\mathbb{R}^d$.

Work by Elkin et al.in [29] demonstrated that y can be taken as an arbitrary point in $\mathbb{R}^d$ with $m = O(\log n)$ and distortion $\approx \sqrt{10}$. This embedding is called a **terminal embedding** with multiplicative factor on the right hand side referred to as the **terminal distortion**. Further work demonstrated that if m is sufficiently large, one may prove a result of the following type.

Theorem 1.3.1 (Lemma 1.1, [79])**.** *Let* $\epsilon \in (0,1)$ *and* $X \subset \mathbb{R}^d$ *be arbitrary with* $|X| = n > 1$. *There exists* $f : X \longrightarrow \mathbb{R}^m$ *with* $m = O(\epsilon^{-2} \log n)$ *such that* $\forall x \in X, \forall y \in \mathbb{R}^d$

$$\|x - y\|_2 \leq \|f(x) - f(y)\|_2 \leq (1 + \epsilon)\|x - y\|_2. \tag{1.27}$$

Thus if the points in X are mapped to $\mathbb{R}^m$ well, which occurs with high probability, then our final terminal embedding is guaranteed to have low terminal distortion as a map from all of $\mathbb{R}^d$ to $\mathbb{R}^m$. This terminal embedding is required to be non-linear. To see this, let $X \subset \mathbb{R}^d$ be arbitrary. Suppose for contradiction that $f : X \longrightarrow \mathbb{R}^m, d > m$ is a linear embedding with constant terminal distortion. By the Rank-Nullity theorem, $dim(ker(f)) \geq d - m \geq 1$. This means $\exists y \in ker(f) \setminus \{0\}$.

Let $x \in X$ be arbitrary. Since f is a linear embedding and $x - y \in \mathbb{R}^d$,

$$0 < \|y\|_2 = \|x - (x - y)\|_2 \leq \|f(x) - f(x - y)\|_2 = \|f(x) - f(x) + f(y)\|_2 = \|f(y)\|_2 = 0.$$

Thus we have arrived at a contradiction.

In Chapter 4, we will explore dimensionality reduction of manifolds. We also demonstrate numerically, a compressed classification classification algorithm for labelled data.

CHAPTER 2

TOWARD FAST AND PROVABLY ACCURATE NEAR-FIELD PTYCHOGRAPHIC PHASE RETRIEVAL

Abstract

Ptychography is an imaging technique that involves a sample being illuminated by a coherent, localized probe of illumination. When the probe interacts with the sample, the light is diffracted and a diffraction pattern is detected. Then the sample (or probe) is shifted laterally in space to illuminate a new area of the sample whilst ensuring sufficient overlap. Similarly, in Fourier ptychography a sample is illuminated at different angles of incidence (effectively shifting the sample's Fourier transform) after which a lens acts as a low-pass filter, thereby effectively providing localized Fourier information about the sample around frequencies dictated by each angle of illumination. Mathematically, one therefore obtains a similar set of overlapping measurements of the sample in both Fourier ptychography and ptychography, except in the different domains (Fourier for the former, and physical for the latter). In either case, one is then able to reconstruct an image of the sample from the measurements using similar methods.

Near-Field (Fourier) Ptychography (NFP) (see, e.g., [106, 107, 124]) occurs when the sample is placed at a short defocus distance having a large Fresnel number.

In this chapter, we prove that certain NFP measurements are robustly invertible (up to an unavoidable global phase ambiguity) for specific Point Spread Functions (PSFs) and physical masks which lead to well-conditioned lifted linear systems. We then apply a block phase retrieval algorithm using weighted angular synchronization and prove that the proposed approach accurately recovers the measured sample for these specific PSF and mask pairs. Finally, we also propose using a Wirtinger Flow for NFP problems and numerically evaluate that alternate approach both against our main proposed approach, as well as with NFP measurements for which our main approach does not apply.

2.1 Introduction

The task of recovering a complex signal $\mathbf{x} \in \mathbb{C}^d$ from phaseless magnitude measurements is called the *phase retrieval problem*. These types of problems appear in many applications such as optics [3, 118] and x-ray crystallography [9, 72]. Here, we are interested in phase retrieval problems arising from (Fourier) ptychography [95, 125]. Ptychography is an imaging technique involving a sample illuminated by a coherent and often localized probe of illumination. When the probe interacts with the sample, light is diffracted and a diffraction pattern is detected. The probe, or the sample, is then shifted laterally in space to illuminate a new area of the sample while ensuring there is sufficient overlap between each neighboring shift. The intensity of the diffraction pattern detected at position ℓ resulting from the k^{th} shift of the probe along the sample takes the general form of

$$\tilde{Y}_{k,\ell} = |(D(S_k\mathbf{m} \circ \mathbf{x}))_\ell|^2, \tag{2.1}$$

where $\mathbf{x} \in \mathbb{C}^d$ is the sample being imaged, $\mathbf{m} \in \mathbb{C}^d$ is a mask which represents the probe's incident illumination on (a portion of) the sample, $\circ$ denotes the Hadamard (pointwise) product, S_k is a shift operator, and $D : \mathbb{C}^d \to \mathbb{C}^d$ is a function that describes the diffraction of the probe radiation from the sample to the plane of the detector after possibly passing through, e.g, a lens. Similarly, Fourier ptychography ultimately results in the same type of measurements as in (2.1) except with $\mathbf{m}$ and $\mathbf{x}$ replaced by $\widehat{\mathbf{m}}$ and $\widehat{\mathbf{x}}$, respectively (see, e.g., [127]).

Prior work in the computational mathematics community related to (Fourier) ptychographic imaging has primarily focused on Far-Field[1] Ptychography (FFP) in which D is the action of a discrete (inverse) Fourier transform matrix (see, e.g., [94, 53, 49, 31, 91, 88]) in (2.1). Here, in contrast, we consider the less well studied setting of near-field ptychography (NFP) which describes situations where, e.g., the masked sample is too close to the source/detector to be well described by the FFP model. See, e.g., [106, 107, 124] for such imaging applications as well as for more detailed related discussions. In all of these NFP applications the acquired measurements can again

[1] Far-field versus near-field measurements are defined based on the *Fresnel number* of the imaging system. See, e.g., [56] for details.

be written in the form of (2.1) where D is now a convolution operator with a given Point Spread Function (PSF) $\mathbf{p} \in \mathbb{C}^d$.

Let $\mathbf{x} \in \mathbb{C}^d$ denote an unknown sample, $\mathbf{m} \in \mathbb{C}^d$ be a known mask, and $\mathbf{p} \in \mathbb{C}^d$ be a known PSF, respectively. For the remainder of this chapter we will suppose we have noisy discretized NFP measurements of the form

$$Y_{k,\ell} = Y_{k,\ell}(\mathbf{x}) := |(\mathbf{p} * (S_k \mathbf{m} \circ \mathbf{x}))_\ell|^2 + N_{k,\ell}, \quad (k,\ell) \in \mathcal{S} \subseteq [d]_0 \times [d]_0, \tag{2.2}$$

where S_k is a circular shift operator $(S_k \mathbf{x})_n = \mathbf{x}_{n+k \bmod d}$, $\mathbf{N} = (N_{k,\ell})$ is an additive noise matrix, and $[d]_0 := \{0, \dots, d-1\}$. Throughout this chapter we will always index vectors and matrices modulo d unless otherwise stated.

2.1.1 Results, Contributions, and Contents

Our main theorem guarantees the existence of a PSF $\mathbf{p} \in \mathbb{C}^d$ and a locally supported mask $\mathbf{m} \in \mathbb{C}^d$ with $supp(\mathbf{m}) \subseteq [\delta]_0 := \{0, \dots, \delta - 1\}$, $\delta \ll d$, for which the measurements (2.2) can be inverted up to a global phase factor by a computationally efficient and noise robust algorithm. In particular, we prove the following result which we believe to be the first theoretical error guarantee for a recovery algorithm in the setting of NFP.

Theorem 2.1.1 (Inversion of NFP Measurements)**.** *Choose $\delta \in [d]_0$ such that $2\delta - 1$ divides d. Then, there exists a PSF $\mathbf{p} \in \mathbb{C}^d$ and a mask $\mathbf{m} \in \mathbb{C}^d$ with $supp(\mathbf{m}) \subseteq [\delta]_0$ such that Algorithm 2.1 below, when provided with input measurements (2.2), will return an estimate $\mathbf{x}_{est} \in \mathbb{C}^d$ of $\mathbf{x}$ satisfying*

$$\min_{\phi \in [0,2\pi)} \|\mathbf{x}_{est} - e^{\mathbb{i}\phi}\mathbf{x}\|_2 \leq C\left(\|\mathbf{x}\|_\infty \frac{d\sqrt{\delta}\sqrt{\|\mathbf{x}_{est}\|_\infty^2 + \|\mathbf{x}_{est}\|_\infty^3}}{|\mathbf{x}_{est}|_{\min}^2} \cdot \|\mathbf{N}\|_F + \sqrt{d\delta}\|\mathbf{N}\|_F\right).$$

Here $C \in \mathbb{R}^+$ is an absolute constant[2]*, and $|\mathbf{x}_{est}|_{\min}$ denotes the smallest magnitude of any entry in $\mathbf{x}_{est}$.*

[2] In this chapter we will use C to denote absolute constants which may change from line to line.

Looking at Theorem 2.1.1 we can see, e.g., that in the noiseless setting where $\|\mathbf{N}\|_F = 0$ the output $\mathbf{x}_{\text{est}}$ of Algorithm 2.1 is guaranteed to match the measured signal $\mathbf{x}$ up to a global phase factor whenever $\mathbf{x}_{\text{est}}$ has no zeros.[3] Moreover, the method is also robust to small amounts of additive noise. The proof of Theorem 2.1.1 consists of two parts: First, in Section 2.3, we show that a specific PSF and mask choice results in NFP measurements (2.2) which are essentially equivalent to far-field ptychographic measurements (2.4) that are known to be robustly invertible by prior work [53, 49, 91]. This guarantees the existence of PSFs and masks which allow for the robust inversion of (2.2) up to a global phase. However, these prior works all prove error bounds on $\min_{\phi\in[0,2\pi)} \|\mathbf{x}_{\text{est}} - e^{\mathbb{i}\phi}\mathbf{x}\|_2$ which scale *quadratically* in d (see, e.g., Corollary 3 in [49] and Theorem 1 in [91]). This motivates the second part of the proof in Section 2.4, where we improve these results so that they only depend *linearly* on d. This is achieved by utilizing weighted angular synchronization error bounds from [33] which require, among other things, updated lower bounds for the second smallest eigenvalue of the unnormalized graph Laplacian of a given weighted graph obtained from $\mathbf{x}$ (derived in Section 2.4 with the help of auxiliary results proven in Appendix A.2). We also note that the improved dependence on d proven in Section 2.4 for the FFP methods previously analyzed in [53, 49, 91] may be of potential independent interest.

Theorem 2.1.1 is proven for a specific $(2\delta - 1)$-periodic PSF $\mathbf{p}$ and locally supported mask $\mathbf{m}$ whose induced lifted linear measurement operator (see (2.5) – (2.7) below together with Lemma 2.3.1) is provably well conditioned. See Lemma 2.3.2 for the definition of this particular $\mathbf{p}, \mathbf{m}$ pair as well as for their measurements' related condition number bound. However, we note that Algorithm 2.1 is guaranteed to work well much more generally for *any* PSF and mask pair which leads to well-conditioned measurements (up to, at worst, potentially having to change the shift and frequency pairs one samples if, e.g., $\mathbf{p}$ is not periodic – see Remark 2.3.1). Indeed, inspecting the proof of Theorem 2.1.1 we see that Lemma 2.4.1 decomposes the total error of Algorithm 2.1, $\min_{\phi\in[0,2\pi)}\left\|\mathbf{x} - e^{\mathbb{i}\phi}\mathbf{x}_{\text{est}}\right\|_2$, into terms involving the phase error, $\min_{\phi\in[0,2\pi)}\|\mathbf{x}_{\text{est}}^{(\theta)} - e^{\mathbb{i}\phi}\mathbf{x}^{(\theta)}\|_2$, and the magnitute error, $\|\mathbf{x}^{(\text{mag})} - \mathbf{x}_{\text{est}}^{(\text{mag})}\|_2$. The phase error is controlled by Theorem 2.4.2 and Lemma

[3] Note that prior work on far-field ptychography assumed that $\mathbf{x}$ itself was non-vanishing (see e.g. [53, 49]). However, requiring $\mathbf{x}_{\text{est}}$ to not vanish is more easily verifiable in practice.

2.4.3. The proof of these results only depends on the choice of $\mathbf{m}$ and $\mathbf{p}$ through $\sigma_{min}(\check{\mathbf{M}}^{(p,m)})$, the minimal singular value of their induced measurement operator. Similarly, the magnitude error is controlled by Lemma 2.4.2 and Theorem 2.4.1 which also only depend on $\mathbf{m}$ and $\mathbf{p}$ through $\sigma_{min}(\check{\mathbf{M}}^{(p,m)})$. Therefore, variants of these results can be derived for any invertible measurement system. Moreover, numerical experiments demonstrate that the proposed method works well for a wide variety of non-vanishing PSF and locally supported mask pairs.

In order to be able to handle even more general PSFs $\mathbf{p}$ which do however, e.g., vanish, in Section 2.5 we also propose a Wirtinger Flow based algorithm, Algorithm 2.2, for inverting NFP measurements (2.2). Though slower than Algorithm 2.1 and less well supported by theory for the PSF and mask pairs for which both methods work empirically, Algorithm 2.2 generally appears more flexible and, e.g., also requires fewer shifts than Algorithm 2.1 to work well in practice when a given mask is not locally supported. Similar to Algorithm 2.1, Algorithm 2.2 relies on the observation that the NFP measurements (2.2) are essentially equivalent to FFP measurements as shown in Section 2.3. In Section 2.6, we evaluate Algorithm 2.1 and Algorithm 2.2, numerically, both individually and in comparison to one another in the case of locally supported masks. Finally, in Section 2.8, we conclude with a brief discussion of future work.

2.2 Preliminaries: Prior Results for Far-Field Ptychography using Local Measurements

Our method, described in Algorithm 2.1, is based on relating the near-field ptychographic measurements (2.2) to far-field ptychographic measurements of the form

$$\widetilde{Y}_{k,\ell} = \widetilde{Y}_{k,\ell}\left(\mathbf{x}\right) \coloneqq \left|\sum_{n=0}^{d-1} m'_n x_{n+k} e^{-2\pi \mathbb{i} \ell n/d}\right|^2 + N_{k,\ell}, \tag{2.3}$$

where $\mathbf{m}'$ is a compactly supported mask. If we let $(\check{\mathbf{m}}_\ell)_n \coloneqq \overline{m'_n} e^{-2\pi \mathbb{i} \ell n/d}$, then these measurements can be written as

$$\widetilde{Y}_{k,\ell} = |\langle \check{\mathbf{m}}_\ell, S_k \mathbf{x} \rangle|^2 + N_{k,\ell}, \tag{2.4}$$

Table 2.1 Notational Reference Table

Notation	Definition	Notes
$[n]_0$	$[n]_0 = \{0, 1, 2, \ldots, n-1\}$	Zero indexing
$(\mathbf{x})_n$	$\mathbf{x} \in \mathbb{C}^d, (\mathbf{x})_n = x_{n \bmod d}$	Vector circular indexing
$(\mathbf{A})_{i,j}$	$\mathbf{A} \in \mathbb{C}^{m\times n}, (\mathbf{A})_{i,j} = A_{i \bmod m, j \bmod n}$	Matrix circular indexing
$\langle \mathbf{x}, \mathbf{y} \rangle$	$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{n=0}^{d-1} x_n \overline{y}_n = \mathbf{y}^* \mathbf{x}$	Complex inner product
$\mathrm{supp}(\mathbf{x})$	$\mathrm{supp}(\mathbf{x}) = \{n \in [d]_0 \mid x_n \neq 0\}$	Support
$\mathbf{F}_d$	$(\mathbf{F}_d)_{j,k} = e^{-2\pi \mathbb{i} jk/d}, \forall (j,k) \in [d]_0 \times [d]_0$	Discrete Fourier transform matrix
$\widehat{\mathbf{x}}$	$\widehat{x}_n = (\mathbf{F}_d \mathbf{x})_n = \sum_{k=0}^{d-1} x_k e^{-2\pi \mathbb{i} nk/d}$	Discrete Fourier transform
$\mathbf{F}_d^{-1}\mathbf{x}$	$(\mathbf{F}_d^{-1}\mathbf{x})_n = \frac{1}{d} \sum_{k=0}^{d-1} x_k e^{2\pi \mathbb{i} kn/d}$	Discrete inverse Fourier transform
$S_k(\mathbf{x})$	$(S_k \mathbf{x})_n = x_{(n+k) \ mod\ d}, \quad \forall n \in [d]_0$	Circular shift
$\widetilde{\mathbf{x}}$	$\widetilde{x}_n = x_{-n \ mod\ d}, \quad \forall n \in [d]_0$	Reversal
$\mathbf{x} * \mathbf{y}$	$(\mathbf{x} * \mathbf{y})_n = \sum_{k=0}^{d-1} x_k y_{n-k}$	Circular convolution
$\mathbf{x} \circ \mathbf{y}$	$(\mathbf{x} \circ \mathbf{y})_n = x_n y_n$	Hadamard (pointwise) product

where as above S_k denotes a circular shift of length k, i.e., $(S_k \mathbf{x})_n = x_{(n+k) \ mod\ d}$. In [53], phase retrieval measurements of this form are studied when $\mathbf{m}'$ is supported in an interval of length δ for some $\delta \ll d$. The fast phase retrieval (fpr) method used there relies on using a lifted linear system involving a block-circulant matrix to recover a portion of the autocorrelation matrix $\mathbf{x}\mathbf{x}^*$. Specifically, letting $D := d(2\delta - 1)$, the authors define a block-circulant matrix $\breve{\mathbf{M}} \in \mathbb{C}^{D\times D}$ by

$$\breve{\mathbf{M}} := \begin{pmatrix} \breve{\mathbf{M}}_0 & \breve{\mathbf{M}}_1 & \ldots & \breve{\mathbf{M}}_{\delta-1} & 0 & 0 & \ldots & 0 \\ 0 & \breve{\mathbf{M}}_0 & \breve{\mathbf{M}}_1 & \ldots & \breve{\mathbf{M}}_{\delta-1} & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \breve{\mathbf{M}}_1 & \ldots & \breve{\mathbf{M}}_{\delta-1} & 0 & 0 & 0 & \ldots & \breve{\mathbf{M}}_0 \end{pmatrix}. \tag{2.5}$$

where the matrices $\breve{\mathbf{M}}_k \in \mathbb{C}^{(2\delta-1)\times(2\delta-1)}$ are defined entry-wise by

$$(\breve{\mathbf{M}}_k)_{\ell j} \coloneqq \begin{cases} (\breve{\mathbf{m}}_\ell)_k \overline{(\breve{\mathbf{m}}_\ell)}_{j+k}, & 0 \le j \le \delta - k, \\ (\breve{\mathbf{m}}_\ell)_k \overline{(\breve{\mathbf{m}}_\ell)}_{j+k-2\delta}, & 2\delta - 1 + k \le j \le 2\delta - 2 \text{ and } k < \delta, \\ 0, & \text{otherwise.} \end{cases} \tag{2.6}$$

Letting $\mathbf{z} \in \mathbb{C}^d$ be a vector obtained by subsampling appropriate entries of $\mathrm{vec}(\mathbf{x}\mathbf{x}^*)$, the authors show that, in the noiseless setting,

$$\mathrm{vec}(\widetilde{\mathbf{Y}}) = \breve{\mathbf{M}}\mathbf{z}, \quad \widetilde{\mathbf{Y}} \in \mathbb{C}^{d\times(2\delta-1)}. \tag{2.7}$$

(See Equation (9) of [53] for explicit details on the arrangement of the entries.) For properly chosen $\mathbf{m}$, the matrix $\breve{\mathbf{M}}$ is invertible, and therefore one may solve for $\mathbf{z}$ by multiplying by $\breve{\mathbf{M}}^{-1}$, i.e., $\mathbf{z} = \breve{\mathbf{M}}^{-1}\mathrm{vec}(\mathbf{Y})$. Then, one may reshape $\mathbf{z}$ to recover a $d \times d$ matrix $\widehat{\mathbf{X}}$ whose non-zero entries are estimates of the autocorrelation matrix $\mathbf{x}\mathbf{x}^*$. One may then obtain a vector $\mathbf{x}_{\mathrm{est}}$ which approximates $\mathbf{x}$ by angular synchronization procedure such as the eigenvector-based method which we will discuss in Section 2.2.1.

In [53], it is shown that exponential masks $\breve{\mathbf{m}}_\ell^{(\mathrm{fpr})}$ defined by

$$(\breve{\mathbf{m}}_\ell^{(\mathrm{fpr})})_n = \begin{cases} \frac{e^{-(n+1)/a}}{\sqrt[4]{2\delta-1}} \cdot e^{\frac{2\pi \mathrm{i} n\ell}{2\delta-1}}, & n \in [\delta]_0 \\ 0, & \text{otherwise} \end{cases}, \quad a \coloneqq \max\left\{4, \frac{\delta-1}{2}\right\}, \tag{2.8}$$

lead to a lifted linear system which is well-conditioned and thus to provable recovery guarantees for the method described above. In particular, we may obtain the following upper bound for the condition number of block-circulant matrix $\breve{\mathbf{M}}^{(\mathrm{fpr})}$ obtained when one sets $\breve{\mathbf{m}}_\ell = \breve{\mathbf{m}}_\ell^{(\mathrm{fpr})}$.

Theorem 2.2.1 (Theorem 4 and Equation (33) in [53])**.** *The condition number of* $\breve{\mathbf{M}}^{(fpr)}$*, the matrix*

obtained by setting $\breve{\mathbf{m}}_\ell = \breve{\mathbf{m}}_\ell^{(fpr)}$ *in* (2.6)*, may be bounded by*

$$\kappa\left(\breve{\mathbf{M}}^{(fpr)}\right) < \max\left\{144e^2, \frac{9e^2(\delta-1)^2}{4}\right\} \leq C\delta^2, \quad C \in \mathbb{R}^+.$$

Furthermore, $\breve{\mathbf{M}}^{(fpr)}$ *can be inverted in* $O(\delta \cdot d \log d)$*-time and its smallest singular value* $\sigma_{min}\left(\breve{\mathbf{M}}^{(fpr)}\right)$ *is bounded from below by* C/δ*.*

2.2.1 Angular Synchronization

Inverting $\breve{\mathbf{M}}$ as described in the previous subsection allows one to obtain a portion of the autocorrelation matrix $\mathbf{x}\mathbf{x}^*$. This motivates us to consider angular synchronization, the process of recovering a vector $\mathbf{x}$ from (a portion of) its autocorrelation matrix $\mathbf{x}\mathbf{x}^*$ (or an estimate $\widehat{\mathbf{X}}$). One popular approach, which we discuss below, is based on upon first entry-wise normalizing this matrix and then taking the lead eigenvector. Specifically, we define a truncated autocorrelation matrix $\mathbf{X}$ corresponding to the true signal $\mathbf{x}$ by

$$X_{j,k} = \begin{cases} x_j\overline{x_k}, & |j-k| \bmod d < \delta \\ 0, & \text{otherwise.} \end{cases} \tag{2.9}$$

We also define a truncated autocorrelation matrix $\widehat{\mathbf{X}}$ corresponding to our estimate, $\mathbf{x}_{\text{est}}$, given by

$$\widehat{X}_{j,k} = \begin{cases} (x_{\text{est}})_j\overline{(x_{\text{est}})_k}, & |j-k| \bmod d < \delta \\ 0, & \text{otherwise.} \end{cases} \tag{2.10}$$

The method from [53] is based upon first solving for $\widehat{\mathbf{X}}$ and then solving for $\mathbf{x}_{\text{est}}$. If $\widehat{\mathbf{X}}$ is a good approximation of $\mathbf{X}$, then the results proved in [117] show that $\mathbf{x}_{\text{est}}$ will be a good approximation of $\mathbf{x}$.

Moving forward, prior works [53, 117] effectively decomposed $\mathbf{X} = \mathbf{X}^{(\theta)} \circ \mathbf{X}^{(\text{mag})}$ into its phase and magnitude matrices by setting $X^{(\text{mag})}_{j,k} = |X_{j,k}|$ and $X^{(\theta)}_{j,k} = X_{j,k}/|X_{j,k}|$ if $|X_{j,k}| \neq 0$ with $X^{(\theta)}_{j,k} = 0$

otherwise. One may then write $\widehat{\mathbf{X}} = \widehat{\mathbf{X}}^{(\theta)} \circ \widehat{\mathbf{X}}^{(\text{mag})}$. Note that by construction, if $\mathbf{x}$ is nonvanishing, then we have $|X^{(\theta)}_{j,k}| = 1$ and $X^{(\text{mag})}_{j,k} > 0$ whenever $|j-k| \bmod d < \delta$. Letting $\mathbf{u} \in \mathbb{C}^d$ be the leading eigenvector of $\widehat{\mathbf{X}}$ and letting $\text{diag}(\widehat{\mathbf{X}}) \in \mathbb{C}^d$ be the main diagonal of $\widehat{\mathbf{X}}$, the output of the resulting algorithm is then $\mathbf{x}_{est} := \text{diag}(\widehat{\mathbf{X}}) \circ \mathbf{u}$.

Example 2.2.1. *Let $d = 4, \delta = 2$. Then $\widehat{\mathbf{X}}$ defined as in (2.10) is given by*

$$\widehat{\mathbf{X}} = \begin{pmatrix} |(x_{est})_0|^2 & (x_{est})_0\overline{(x_{est})_1} & 0 & (x_{est})_0\overline{(x_{est})_3} \\ (x_{est})_1\overline{(x_{est})_1} & |(x_{est})_1|^2 & (x_{est})_1\overline{(x_{est})_2} & 0 \\ 0 & (x_{est})_2\overline{(x_{est})_1} & |(x_{est})_2|^2 & (x_{est})_2\overline{(x_{est})_3} \\ (x_{est})_3\overline{(x_{est})_0} & 0 & (x_{est})_3\overline{(x_{est})_2} & |(x_{est})_3|^2 \end{pmatrix}.$$

If we write $(x_{est})_n = |(x_{est})_n|e^{\mathbb{i}\theta_n}$, then we may compute

$$\widehat{\mathbf{X}}^{(\theta)} = \begin{pmatrix} 1 & e^{i(\theta_0-\theta_1)} & 0 & e^{i(\theta_0-\theta_3)} \\ e^{i(\theta_1-\theta_0)} & 1 & e^{i(\theta_1-\theta_2)} & 0 \\ 0 & e^{i(\theta_2-\theta_1)} & 1 & e^{i(\theta_2-\theta_3)} \\ e^{i(\theta_3-\theta_0)} & 0 & e^{i(\theta_3-\theta_2)} & 1 \end{pmatrix}.$$

One may verify that the lead eigenvector is $\mathbf{u} = (e^{\mathbb{i}\theta_0}\ e^{i\theta_1}\ e^{i\theta_2}\ e^{\mathbb{i}\theta_3})^T$ and therefore

$$\mathbf{x}_{est} = \sqrt{diag(\widehat{\mathbf{X}})} \circ \mathbf{u} = (|(x_{est})_0|e^{\mathbb{i}\theta_0}\ |(x_{est})_1|e^{\mathbb{i}\theta_1}\ |(x_{est})_2|e^{\mathbb{i}\theta_2}\ |(x_{est})_3|e^{\mathbb{i}\theta_3})^T.$$

In Section 2.4, we will discuss another slightly more sophisticated way for estimating the phases based on Algorithm 3 of [92] which involves taking the smallest eigenvector of an appropriately weighted graph Laplacian. Indeed, this new angular synchronization approach is what ultimately allows for the NFP error bound in Theorem 2.1.1 to have improved dependence on signal dimension d over prior FFP error bounds in [53, 49, 91]. The end result will be a more accurate method for computing $\widehat{\mathbf{X}}$ in (2.10) from given NFP measurements (2.2).

2.3 Near from Far: Guaranteed Near-Field Ptychographic Recovery via Far-Field Results

In this section, we show how to relate the near-field ptychographic measurements (2.2) to the far-field ptychographic measurements (2.4). This will allow us to recover $\mathbf{x}$ by using methods similar to those introduced in [53]. In order get nontrivial bounds, we will also need to prove the existence of an admissible PSF and mask pair, $\mathbf{p} \in \mathbb{C}^d$ and $\mathbf{m} \in \mathbb{C}^d$, which lead to a well conditioned linear system in (2.7). In particular, we will present a PSF and mask pair such that the resulting block-circulant matrix, denoted $\check{\mathbf{M}}^{(p,m)}$, will have the same condition number as the matrix $\check{\mathbf{M}}^{(\text{fpr})}$ constructed from the masks $\check{\mathbf{m}}_\ell^{(\text{fpr})}$ defined in (2.8). Therefore, Theorem 2.2.1 will allow us to obtain convergence guarantees for Algorithm 2.1.

Here, we will set the measurement index set $\mathcal{S}$ considered in (2.2) to be $\mathcal{S} = \mathcal{K} \times \mathcal{L}$ where $\mathcal{K} = [d]_0$ and $\mathcal{L} = [2\delta - 1]_0$. The following lemma proves that we can rewrite NFP measurements from (2.2) as local FFP measurements of the form (2.4) as long as the mask $\mathbf{m}$ has local support and the PSF is periodic. It will be based upon defining masks

$$\check{\mathbf{m}}_\ell^{(p,m)} := \overline{S_\ell \widetilde{\mathbf{p}} \circ \mathbf{m}} \in \mathbb{C}^d, [\delta]_0, \tag{2.11}$$

where $\widetilde{\mathbf{p}}$ is the reversal of $\mathbf{p}$ about its first entry modulo d, i.e., $\widetilde{p}_n = p_{-n \bmod d}$ Since the masks $\check{\mathbf{m}}_\ell^{(p,m)}$ have compact support, this will then yield a lifted set of linear measurements of the type considered in [53, 49, 91].

Lemma 2.3.1. *Let $\mathcal{S} = \mathcal{K} \times \mathcal{L} = [d]_0 \times [2\delta - 1]_0$, and recall the measurements*

$$Y_{k,\ell} = |(\mathbf{p} * (S_k \mathbf{m} \circ \mathbf{x}))_\ell|^2, \ (k, \ell) \in \mathcal{S},$$

defined in (2.2). Suppose that $2\delta - 1$ divides d, that $\mathbf{p} \in \mathbb{C}^d$ is $2\delta - 1$ periodic, and that $\mathbf{m} \in \mathbb{C}^d$ satisfies $supp(\mathbf{m}) \subseteq [\delta]_0$. Then, we may rearrange the measurements (2.2) into a matrix of

FFP-type measurements

$$\widetilde{Y}_{k,\ell} \coloneqq Y_{-k \mod d,\, k-\ell \mod 2\delta-1} = |\langle \breve{\mathbf{m}}_\ell^{(p,m)}, S_k \mathbf{x} \rangle|^2, \ (k,\ell) \in [d]_0 \times [2\delta-1]_0, \tag{2.12}$$

where $\breve{\mathbf{m}}_\ell^{(p,m)}$ *is defined as in* (2.11)*. As a consequence, recovering* $\mathbf{x}$ *is equivalent to inverting a block-circulant matrix as described in* (2.5) – (2.7)*.*

Proof. By Lemma A.1.3 part 1 , Lemma A.1.2, Lemma A.1.3 part 2, and Lemma A.1.1 from Appendix A.1, we have that

$$\begin{aligned} Y_{k,\ell} = |(\mathbf{p} * (S_k \mathbf{m} \circ \mathbf{x}))_\ell|^2 &= |\langle S_{-\ell} \widetilde{\mathbf{p}}, \overline{S_k \mathbf{m} \circ \mathbf{x}} \rangle|^2 \\ &= |\langle S_{-\ell} \widetilde{\mathbf{p}} \circ S_k \mathbf{m}, \overline{\mathbf{x}} \rangle|^2 \\ &= |\langle S_k (S_{-\ell-k} \widetilde{\mathbf{p}} \circ \mathbf{m}), \overline{\mathbf{x}} \rangle|^2 \\ &= |\langle S_k (\overline{S_{-\ell-k} \widetilde{\mathbf{p}} \circ \mathbf{m}}), \mathbf{x} \rangle|^2 \\ &= |\langle S_k (\overline{S_{-\ell-k \mod 2\delta-1} \widetilde{\mathbf{p}} \circ \mathbf{m}}), \mathbf{x} \rangle|^2, \end{aligned}$$

where the last equality uses the fact that $\mathbf{p}$ is $2\delta - 1$ periodic. We may now apply Lemma A.1.3 part 3 to see that

$$Y_{k,\ell} = |\langle S_k (\overline{S_{-\ell-k \mod 2\delta-1} \widetilde{\mathbf{p}} \circ \mathbf{m}}), \mathbf{x} \rangle|^2 = |\langle (\overline{S_{-\ell-k \mod 2\delta-1} \widetilde{\mathbf{p}} \circ \mathbf{m}}), S_{-k} \mathbf{x} \rangle|^2.$$

Finally, since $\breve{\mathbf{m}}_\ell^{(p,m)} = \overline{S_\ell \widetilde{\mathbf{p}} \circ \mathbf{m}}$, we see that for all $k \in [d]_0$ and all $\ell \in [2\delta-1]_0$, we have

$$\begin{aligned} \widetilde{Y}_{k,\ell} &= Y_{-k \mod d,\, k-\ell \mod 2\delta-1} \\ &= |\langle (\overline{S_{-(k-\ell)-(-k) \mod 2\delta-1} \widetilde{\mathbf{p}} \circ \mathbf{m}}), S_{-(-k)} \mathbf{x} \rangle|^2 \\ &= |\langle (\overline{S_{\ell \mod 2\delta-1} \widetilde{\mathbf{p}} \circ \mathbf{m}}), S_k \mathbf{x} \rangle|^2 \\ &= |\langle \breve{\mathbf{m}}_\ell^{(p,m)}, S_k \mathbf{x} \rangle|^2. \end{aligned}$$

□

Remark 2.3.1. *If we instead change the pairs $\mathcal{S}$ in Lemma 2.3.1 for which we collect NFP measurements (2.2) to be $\mathcal{S}' := \cup_{k\in[d]_0}\{d-k\}\times\{k-2\delta+2,\dots,k-1,k\} \mod d$, then we may remove the assumption that $\mathbf{p}$ is $2\delta-1$ periodic. In particular, for $(k,\ell)\in\mathcal{S}'$ one may substitute $k = d-k'$ and $\ell' = k'-i$ for some $0\le i\le 2\delta-2$ to see that then $(-k'-\ell') \bmod d = i$. Thus, since $0\le i\le 2\delta-2$, $(-\ell'-k') \bmod 2\delta-1 = (-\ell'-k') \bmod d$, and so we may use the same calculation as above without assuming that $\mathbf{p}$ is $2\delta-1$ periodic. Note, however, that $\mathcal{S}$ has a simple Cartesian product structure whereas $\mathcal{S}'$ does not. As a result, the entries of $\mathbf{p}*(S_k\mathbf{m}\circ\mathbf{x})$ that one must sample varies based on the mask shift k in the case of $\mathcal{S}'$, potentially complicating the collection of the associated NFP measurements (2.2) in some situations.*

Next, in Lemma 2.3.2 below, we will show how to choose a mask $\mathbf{m}$ and PSF $\mathbf{p}$ such that $\breve{\mathbf{m}}_\ell^{(p,m)}$ defined as in (2.11) and $\breve{\mathbf{m}}^{(\mathrm{fpr})}_{2\ell \bmod 2\delta-1}$ defined as in (2.8) will only differ by a global phase for each $\ell\in[2\delta-1]_0$. As a consequence, we obtain the desired result that the block-circulant matrix arising from the NFP measurements (2.2) is essentially equivalent (up to a row permutation and global phase shift) to the well-conditioned lifted linear measurement operator $\breve{\mathbf{M}}^{(\mathrm{fpr})}$ considered in Theorem 2.2.1.

Lemma 2.3.2. *Let $\mathbf{p},\mathbf{m}\in\mathbb{C}^d$ have entries given by*

$$p_n := e^{-\frac{2\pi\mathbb{i}n^2}{2\delta-1}}, \quad \text{and} \quad m_n := \begin{cases} \dfrac{e^{-n+1}/a}{\sqrt[4]{2\delta-1}}\cdot e^{\frac{2\pi\mathbb{i}n^2}{2\delta-1}}, & n\in[\delta]_0 \\ 0, & \text{otherwise} \end{cases},$$

where $a := \max\left\{4, \dfrac{\delta-1}{2}\right\}$. Then for all $\ell\in[2\delta-1]_0$, $\breve{\mathbf{m}}_\ell^{(p,m)} = \overline{S_\ell\widetilde{\mathbf{p}}\circ\mathbf{m}}$ satisfies

$$\breve{\mathbf{m}}_\ell^{(p,m)} = e^{\frac{2\pi\mathbb{i}\ell^2}{2\delta-1}}\cdot\breve{\mathbf{m}}^{(fpr)}_{2\ell \bmod 2\delta-1}, \tag{2.13}$$

where $\breve{\mathbf{m}}_\ell^{(fpr)}$ is defined as in (2.8). As a consequence, if we let $\breve{\mathbf{M}}^{(fpr)}$ and $\breve{\mathbf{M}}^{(p,m)}$ be the lifted linear

measurement matrices as per (2.5) obtained by setting each $\breve{\mathbf{m}}_\ell$ in (2.6) equal to $\breve{\mathbf{m}}_\ell^{(fpr)}$ and $\breve{\mathbf{m}}_\ell^{(p,m)}$, respectively, then we will have

$$\breve{\mathbf{M}}^{(p,m)} = \mathbf{P}\breve{\mathbf{M}}^{(fpr)}, \tag{2.14}$$

where $\mathbf{P}$ is a $D \times D$ block diagonal permutation matrix. Thus $\breve{\mathbf{M}}^{(p,m)}$ and $\breve{\mathbf{M}}^{(fpr)}$ have the same singular values and

$$\kappa\left(\breve{\mathbf{M}}^{(p,m)}\right) = \kappa\left(\breve{\mathbf{M}}^{(fpr)}\right) \leq C\delta^2,$$

where $\kappa(\cdot)$ denotes the condition number of a matrix.

Proof. Using the definition of the Hadamard product $\circ$, the circulant shift operator S_ℓ and the reversal operator $\mathbf{x} \mapsto \widetilde{\mathbf{x}}$, we see that

$$(\breve{\mathbf{m}}_\ell^{(p,m)})_n = \overline{(S_\ell\widetilde{\mathbf{p}} \circ \mathbf{m})_n} = \overline{(S_\ell\widetilde{\mathbf{p}})_n\mathbf{m}_n} = \widetilde{\overline{\mathbf{p}}}_{n+\ell}\overline{\mathbf{m}_n} = \overline{\mathbf{p}}_{-n-\ell}\overline{\mathbf{m}_n}.$$

Therefore, inserting the definitions of $\mathbf{p}$ and $\mathbf{m}$ above shows that for $n \in [\delta]_0$

$$\begin{aligned}
(\breve{\mathbf{m}}_\ell^{(p,m)})_n &= e^{\frac{2\pi\mathbb{i}(n+\ell)^2}{2\delta-1}} \cdot \frac{e^{-(n+1)/a}}{\sqrt[4]{2\delta-1}} \cdot e^{-\frac{2\pi\mathbb{i}n^2}{2\delta-1}} \\
&= e^{\frac{2\pi\mathbb{i}n^2}{2\delta-1}} \cdot e^{\frac{4\pi\mathbb{i}n\ell}{2\delta-1}} \cdot e^{\frac{2\pi\mathbb{i}\ell^2}{2\delta-1}} \cdot \frac{e^{-(n+1)/a}}{\sqrt[4]{2\delta-1}} \cdot e^{-\frac{2\pi\mathbb{i}n^2}{2\delta-1}} \\
&= e^{\frac{2\pi\mathbb{i}\ell^2}{2\delta-1}}\left(\frac{e^{-(n+1)/a}}{\sqrt[4]{2\delta-1}} \cdot e^{\frac{2\pi\mathbb{i}n(2\ell)}{2\delta-1}}\right) = e^{\frac{2\pi\mathbb{i}\ell^2}{2\delta-1}}\left(\breve{\mathbf{m}}^{(\mathrm{fpr})}_{2\ell \bmod 2\delta-1}\right)_n.
\end{aligned}$$

For $n \notin [\delta]_0$, we have $\left(\breve{\mathbf{m}}_\ell^{(p,m)}\right)_n = e^{\frac{2\pi\mathbb{i}\ell^2}{2\delta-1}}\left(\breve{\mathbf{m}}^{(\mathrm{fpr})}_{2\ell \bmod 2\delta-1}\right)_n = 0$. Thus (2.13) follows.

To prove (2.14), let $\breve{\mathbf{M}}^{(p,m)}$ and $\breve{\mathbf{M}}_k^{(p,m)}$ be the matrices obtained by using the mask $\breve{\mathbf{m}}_\ell^{p,m}$ in (2.5) and (2.6) and let $\breve{\mathbf{M}}^{(\mathrm{fpr})}$ and $\breve{\mathbf{M}}_k^{(\mathrm{fpr})}$ be the matrices obtained using $\mathbf{m}_\ell^{(\mathrm{fpr})}$ instead. Then combining (2.13) and (2.6) implies that $\left(\breve{\mathbf{M}}_k^{(p,m)}\right)_{i,j} = \left(\breve{\mathbf{M}}_k^{(\mathrm{fpr})}\right)_{2i \bmod 2\delta-1,j}$. For example, when $j \in [\delta-k+1]_0$

one may check

$$\begin{aligned}\left(\breve{\mathbf{M}}_k^{(p,m)}\right)_{i,j} &= e^{\frac{2\pi \mathbb{i} i^2}{2\delta-1}} \left(\breve{\mathbf{m}}^{(\mathrm{fpr})}_{2i \bmod 2\delta-1}\right)_k \overline{e^{\frac{2\pi \mathbb{i} i^2}{2\delta-1}} \left(\breve{\mathbf{m}}^{(\mathrm{fpr})}_{2i \bmod 2\delta-1}\right)_{j+k}} \\ &= \left(\breve{\mathbf{M}}_k^{(\mathrm{fpr})}\right)_{2i \bmod 2\delta-1,j}, \end{aligned} \tag{2.15}$$

and one may perform similar computations in the other cases. Since each $\breve{\mathbf{M}}_k^{(p,m)}$ and $\breve{\mathbf{M}}_k^{(\mathrm{fpr})}$ have $2\delta - 1$ rows and the mapping $i \to 2i$ is a bijection on $\mathbb{Z}/(2\delta - 1)\mathbb{Z}$ we see that each $\breve{\mathbf{M}}_k^{(p,m)}$ may be obtained by permuting the rows of $\breve{\mathbf{M}}_k^{(\mathrm{fpr})}$ (and that the permutation does not depend on k). Therefore, there exists a block diagonal permutation matrix $\mathbf{P}$ such that $\breve{\mathbf{M}}^{(p,m)} = \mathbf{P}\breve{\mathbf{M}}^{(\mathrm{fpr})}$. Finally, the condition number bound for $\breve{\mathbf{M}}^{(p,m)}$ now follows from Theorem 2.2.1 and the fact that permuting the rows of a matrix does not change its condition number or any of its singular values. □

Lemma 2.3.1 above demonstrates how to recast NFP problems involving locally supported masks and periodic PSFs as particular types of FFP problems. Then, Lemma 2.3.2 provides a particular PSF and mask combination for which the resulting FFP problem can be solved by inverting a well-conditioned linear system. Together they imply that, for properly chosen $\mathbf{m}$ and $\mathbf{p}$, one may robustly invert the measurements given in (2.2) by first recasting the NFP data as modified FFP data and then using the BlockPR approach from [53, 49, 91]. This is the main idea behind Algorithm 2.1. However, this approach will lead to theoretical error bounds which scale *quadratically* in d. To remedy this, the final step of Algorithm 2.1 uses an alternative angular synchronization method (which originally appeared in [92]) based on a weighted graph Laplacian as opposed to previous works which used methods based on, e.g., the methods outlined in Section 2.2.1. As we shall see in the next section, this will allow us to obtain bounds in Theorem 2.1.1 which depend linearly on d rather than quadratically.

Algorithm 2.1 NFP-BlockPR

Input:

1) Variables $d, \delta, D = d(2\delta - 1)$.
2) A $2\delta - 1$ periodic PSF $\mathbf{p} \in \mathbb{C}^d$, and a mask $\mathbf{m} \in \mathbb{C}^d$ with $supp(\mathbf{m}) \subseteq [\delta]_0$.
3) A near-field ptychographic measurement matrix $\mathbf{Y} \in \mathbb{C}^{d \times 2\delta - 1}$.

Output: $\mathbf{x}_{\text{est}}$ with $\mathbf{x}_{\text{est}} \approx e^{i\theta}\mathbf{x}$ for some $\theta \in [0, 2\pi]$.

1) Form masks $\breve{\mathbf{m}}_\ell^{(p,m)} = \overline{S_\ell \widetilde{\mathbf{p}} \circ \mathbf{m}}$ and matrix $\breve{\mathbf{M}}^{(p,m)}$ as per (2.5) and (2.6).
2) Compute $\mathbf{z} = \left(\breve{\mathbf{M}}^{(p,m)}\right)^{-1} \text{vec}(\mathbf{Y}) \in \mathbb{C}^D$.
3) Reshape $\mathbf{z}$ to get $\widehat{\mathbf{X}}$ as per Section 2.2.1 containing estimated entries of $\mathbf{x}\mathbf{x}^*$.
4) Use weighted angular synchronization (Algorithm 3, [92]) to obtain $\mathbf{x}_{\text{est}}$.

2.4 Error Analysis for Algorithm 2.1

In this section, we will prove our main result, Theorem 2.1.1, which provides accuracy and robustness guarantees for Algorithm 2.1. For $\mathbf{x} \in \mathbb{C}^d$, we write its n^{th} entry as $x_n =: |x_n| e^{\mathbb{i}\theta_n}$ and let $\mathbf{x}^{(\text{mag})} := (|x_0|, \ldots, |x_{d-1}|)^T$ and $\mathbf{x}^{(\theta)} := (e^{\mathbb{i}\theta_0}, e^{\mathbb{i}\theta_1}, \ldots, e^{\mathbb{i}\theta_{d-1}})^T$ so that we may decompose $\mathbf{x}$ as

$$\mathbf{x} = \mathbf{x}^{(\text{mag})} \circ \mathbf{x}^{(\theta)}. \tag{2.16}$$

The following lemma upper bounds the total estimation error in terms of its phase and magnitude errors. For a proof, please see Appendix A.1.

Lemma 2.4.1. *Let* $\mathbf{x}$ *be decomposed as in* (2.16)*, and similarly let* $\mathbf{x}_{est}$ *be decomposed* $\mathbf{x}_{est} = \mathbf{x}_{est}^{(mag)} \circ \mathbf{x}_{est}^{(\theta)}$*. Then, we have that*

$$\min_{\phi \in [0,2\pi)} \left\| \mathbf{x} - e^{\mathbb{i}\phi} \mathbf{x}_{est} \right\|_2 \leq \|\mathbf{x}\|_\infty \min_{\phi \in [0,2\pi)} \left\| \mathbf{x}_{est}^{(\theta)} - e^{\mathbb{i}\phi} \mathbf{x}^{(\theta)} \right\|_2 + \left\| \mathbf{x}^{(mag)} - \mathbf{x}_{est}^{(mag)} \right\|_2. \tag{2.17}$$

In light of Lemma 2.4.1, to bound the total error of our algorithm, it suffices to consider the phase and magnitude errors separately. In order to bound $\|\mathbf{x}^{(\text{mag})} - \mathbf{x}_{\text{est}}^{(\text{mag})}\|_2$, we may utilize the following lemma which is a restatement of Lemma 3 of [53].

Lemma 2.4.2 (Lemma 3 of [53])**.** *Let* $\sigma_{\min}\left(\breve{\mathbf{M}}^{(p,m)}\right)$ *denote the smallest singular value of the lifted*

measurement matrix $\breve{\mathbf{M}}^{(p,m)}$ *from line 1 of Algorithm 2.1. Then,*

$$\left\|\mathbf{x}^{(mag)} - \mathbf{x}_{est}^{(mag)}\right\|_\infty \leq C\sqrt{\frac{\|\mathbf{N}\|_F}{\sigma_{min}\left(\breve{\mathbf{M}}^{(p,m)}\right)}}.$$

Having obtained Lemma 2.4.2, we are now able to prove the following theorem bounding the total estimation error.

Theorem 2.4.1. *Let* $\mathbf{p}$ *and* $\mathbf{m}$ *be the admissible PSF, mask pair defined in Lemma 2.3.2. Then, we have that*

$$\min_{\phi\in[0,2\pi)}\left\|\mathbf{x} - e^{\mathbb{i}\phi}\mathbf{x}_{est}\right\|_2 \leq \|\mathbf{x}\|_\infty \min_{\phi\in[0,2\pi)}\left\|\mathbf{x}_{est}^{(\theta)} - e^{\mathbb{i}\phi}\mathbf{x}^{(\theta)}\right\|_2 + C\sqrt{d\delta\|\mathbf{N}\|_F}.$$

Proof. Combining Lemmas 2.4.1 and 2.4.2 along with the inequality $\|\mathbf{u}\|_2 \leq \sqrt{d}\|\mathbf{u}\|_\infty$, implies that

$$\min_{\phi\in[0,2\pi)}\|\mathbf{x} - e^{\mathbb{i}\phi}\mathbf{x}_{\text{est}}\|_2 \leq \|\mathbf{x}\|_\infty \min_{\phi\in[0,2\pi)}\left\|\mathbf{x}_{\text{est}}^{(\theta)} - e^{\mathbb{i}\phi}\mathbf{x}^{(\theta)}\right\|_2 + C\sqrt{\frac{d\|\mathbf{N}\|_F}{\sigma_{min}\left(\breve{\mathbf{M}}^{(p,m)}\right)}}. \tag{2.18}$$

As noted in Lemma 2.3.2, the singular values of $\breve{\mathbf{M}}^{(p,m)}$ are the same as those of $\breve{\mathbf{M}}^{(\text{fpr})}$. Therefore, applying Theorem 2.2.1 then finishes the proof. □

Remark 2.4.1. *Note that the inequality (2.18) in the proof of Theorem 2.4.1 holds any time* $\mathbf{m} \in \mathbb{C}^d$ *satisfies* $supp(\mathbf{m}) \subseteq [\delta]_0$ *and either* (i) $\mathbf{p} \in \mathbb{C}^d$ *is* $2\delta - 1$ *periodic for* $2\delta - 1$ *dividing* d*, or else* (ii) *one instead collects NFP measurements (2.2) at all* $(k, \ell) \in \mathcal{S}'$ *as in Remark 2.3.1. Therefore, results analogous to Theorem 2.4.1 may be produced for any such* $\mathbf{p}$ *and* $\mathbf{m}$ *pairs such that* $\sigma_{min}\left(\breve{\mathbf{M}}^{(p,m)}\right) > 0$*. Furthermore, the value of this minimal singular value is straightforward to check numerically in practice.*

In order to bound $\left\|\mathbf{x}_{\text{est}}^{(\theta)} - e^{\mathbb{i}\phi}\mathbf{x}^{(\theta)}\right\|_2$, we will need a few additional definitions. As in (2.9), let $\mathbf{X}$ denote the partial autocorrelation matrix corresponding to the true signal $\mathbf{x}$, as in (2.10), and let $\widehat{\mathbf{X}}$ denote the partial autocorrelation matrix corresponding to $\mathbf{x}_{\text{est}}$, i.e., the matrix obtained in step 3 of Algorithm 2.1. Let $G = (V, E, \mathbf{W})$ be a weighted graph whose vertices are given by $V = [d]_0$,

whose edge set E is taken to be the set of (i, j) such that $i \neq j$ and $|i - j| \bmod d < \delta$, and whose weight matrix $\mathbf{W}$ is defined entrywise by

$$W_{i,j} = \begin{cases} |\widehat{X}_{i,j}|^2, & 0 < |i - j| \bmod d < \delta \\ 0, & \text{otherwise} \end{cases}. \tag{2.19}$$

Letting $\mathbf{A}_G$ denote the *unweighted* adjacency matrix of G, we observe that by construction, we have $\mathbf{X} = (\mathbf{I} + \mathbf{A}_G) \circ \mathbf{x}\mathbf{x}^*$ and $\widehat{\mathbf{X}} = (\mathbf{I} + \mathbf{A}_G) \circ \mathbf{x}_{\text{est}}\mathbf{x}_{\text{est}}^*$. Letting $\mathbf{D}$ denote the *weighted* degree matrix, we define the *unnormalized graph Laplacian* by $\mathbf{L}_G := \mathbf{D} - \mathbf{W}$ and the *normalized graph Laplacian* by $\mathbf{L}_N := \mathbf{D}^{-1/2}\mathbf{L}_G\mathbf{D}^{-1/2}$. It is well known that both $\mathbf{L}_G$ and $\mathbf{L}_N$ are positive semi-definite with a minimal eigenvalue of zero (see, e.g., Section 3.1, [103]). We will let τ_G denote the spectral gap (second smallest eigenvalue) of $\mathbf{L}_G$. It is well known that if G is connected then τ_G is strictly positive (see, e.g., Lemma 3.1.1, [103]).

In [33], the authors used a weighted graph approach to prove the following result which bounds $\min_{\phi \in [0,2\pi)} \|\mathbf{x}_{\text{est}}^{(\theta)} - e^{\mathbb{i}\phi}\mathbf{x}^{(\theta)}\|_2$.

Theorem 2.4.2 (Corollary 3, [33])**.** *Consider the weighted graph $G = (V, E, \mathbf{W})$ described in the previous paragraph with weight matrix given as in (2.19). Let τ_G denote the spectral gap of the associated unnormalized Laplacian $\mathbf{L}_G$. Then we have that*

$$\min_{\phi \in [0,2\pi)} \left\| \mathbf{x}_{est}^{(\theta)} - e^{\mathbb{i}\phi}\mathbf{x}^{(\theta)} \right\|_2 \leq C\sqrt{1 + \|\mathbf{x}_{\,est}\|_\infty} \cdot \frac{\|\mathbf{X} - \widehat{\mathbf{X}}\|_F}{\sqrt{\tau_G}}, \; C \in \mathbb{R}^+.$$

Remark 2.4.2. *The $\sqrt{1 + \|\mathbf{x}_{est}\|_\infty}$ term is referred to in Theorem 4 of [33] as a tightness penalty which is applied when taking the non-convex constraint and performing an eigenvector relaxation, allowing us to use the method of angular synchronization involving the weighted Laplacian given in Algorithm 3 of [92].*

In order to utilize Theorem 2.4.2 we require both an upper bound of $\|\mathbf{X} - \widehat{\mathbf{X}}\|_F$ and a lower bound for the spectral gap τ_G. These are provided by the next two lemmas.

Lemma 2.4.3. *Let* $\mathbf{p}$ *and* $\mathbf{m}$ *be defined as in Lemma 2.3.2. Then,* $\|\mathbf{X} - \widehat{\mathbf{X}}\|_F \leq C\delta\|\mathbf{N}\|_F$.

Proof. Let $vec : \mathbb{C}^{d\times d} \to \mathbb{C}^D$ be the vectorization operator considered in (2.7). It follows from (2.4), (2.7), and Step 2 of Algorithm 2.1, that

$$\text{vec}(\mathbf{Y}) = \breve{\mathbf{M}}\text{vec}(\widehat{\mathbf{X}}) \quad \text{and} \quad \text{vec}(\mathbf{Y} - \mathbf{N}) = \breve{\mathbf{M}}\text{vec}(\mathbf{X}).$$

Therefore,

$$\left\|\mathbf{X} - \widehat{\mathbf{X}}\right\|_F \leq \left\|\left(\breve{\mathbf{M}}^{(p,m)}\right)^{-1} \text{vec}(\mathbf{N})\right\|_2 \leq \frac{\|\text{vec}(\mathbf{N})\|_2}{\sigma_{\,min}\left(\breve{\mathbf{M}}^{(p,m)}\right)} \leq C\delta\|\mathbf{N}\|_F,$$

where final inequality again utilizes Lemma 2.3.2 and Theorem 2.2.1. □

Lemma 2.4.4. *For the graph* G *considered in Theorem 2.4.2, we have that*

$$\tau_G \geq \frac{|\mathbf{x}_{est}|^4_{\min}}{\|\mathbf{x}_{est}\|^2_\infty}\frac{4(\delta-1)}{d^2}.$$

Proof. Letting $W_{\min}$ and $W_{\max}$ be the minimum and maximum value of any of the (nonzero) entries of $\mathbf{W}$, we have that $W_{\min} \geq |\mathbf{x}_{\text{est}}|^2_{\min}$, $W_{\max} \leq \|\mathbf{x}_{\text{est}}\|^2_\infty$, and $\text{diam}(G_{\text{unw}}) \geq d/(2\delta - 1)$ (where $\text{diam}(G_{\text{unw}})$ is the diameter of the unweighted version of G). Therefore, by Theorem A.2.1 in Appendix A.2, we have that

$$\tau_G \geq \frac{|\mathbf{x}_{\text{est}}|^4_{\min}}{\|\mathbf{x}_{\text{est}}\|^2_\infty}\frac{2}{(d-1)\text{diam}(G)} \geq \frac{|\mathbf{x}_{\text{est}}|^4_{\min}}{\|\mathbf{x}_{\text{est}}\|^2_\infty}\frac{4(\delta-1)}{d^2}.$$

□

We shall now finally prove our main result.

The Proof of Theorem 2.1.1. By Theorem 2.4.1, we have

$$\min_{\phi\in[0,2\pi)} \left\|\mathbf{x} - e^{\mathbb{i}\phi}\mathbf{x}_{\text{est}}\right\|_2 \leq \|\mathbf{x}\|_\infty \min_{\phi\in[0,2\pi)} \left\|\mathbf{x}^{(\theta)}_{\text{est}} - e^{\mathbb{i}\phi}\mathbf{x}^{(\theta)}\right\|_2 + C\sqrt{d\delta\|\mathbf{N}\|_F}.$$

Combining Theorem 2.4.2 with Lemmas 2.4.3 and 2.4.4 yields

$$\begin{aligned}\min_{\phi\in[0,2\pi)} \left\|\mathbf{x}_{\text{est}}^{(\theta)} - e^{\mathbb{i}\theta}\mathbf{x}^{(\theta)}\right\|_2 &\leq C\sqrt{1+\|\mathbf{x}_{est}\|_\infty}\cdot\frac{\|\mathbf{X}-\widehat{\mathbf{X}}\|_F}{\sqrt{\tau_G}}\\ &\leq C\sqrt{1+\|\mathbf{x}_{est}\|_\infty}\cdot\frac{d\sqrt{\delta}\|\mathbf{x}_{\text{est}}\|_\infty\|\mathbf{N}\|_F}{|\mathbf{x}_{\text{est}}|_{\min}^2}.\end{aligned}$$

The result follows. □

2.5 An Alternate Approach: Near-Field Ptychography via Wirtinger Flow

In the previous sections we have demonstrated a particular point spread function and mask for which NFP measurements are guaranteed to allow image reconstruction via Algorithm 2.1. However, in many real-world scenarios the particular mask and PSF combination considered above are not of the type actually used in practice. For example, in the setting considered in [124] the PSF $\mathbf{p}$ ideally behaves like a low-pass filter (so that, e.g., $\widehat{\mathbf{p}}$ is supported in $\{k\in\mathbb{Z}|-K<k \bmod d<K\}$ for some $K\ll d$), and the mask m is globally supported in $[d]_0$. In contrast, the PSF considered above has its nonzero Discrete Fourier coefficients at frequencies in $\{kd/(2\delta-1)\}_{k\in[2\delta-1]_0}$ (and thus its Fourier support includes large frequencies), and the mask m has small physical support in $[\delta]_0$. This motivates us to explore a variant of the well known Wirtinger Flow algorithm [14] in this section. This method, Algorithm 2.2, can be applied to more general set of PSF and mask pairs than Algorithm 2.1 considered in the previous section.

Suppose we have noiseless NFP measurements of the form

$$Y_{k,\ell} = |(\mathbf{p} * (S_k\mathbf{m}\circ\mathbf{x}))_\ell|^2,\quad (k,\ell)\in\bigcup_{0\leq k\leq K-1}\{d-k\}\times\{K-L+1,\dots,k\} \bmod d,$$

where $K, L\in[d+1]_0\setminus\{0\}$. Then by the same argument used in Lemma 2.3.1 (see also in Remark 2.3.1), we can manipulate the measurements above so that we have

$$\widetilde{Y}_{k,\ell} = |\langle\breve{\mathbf{m}}_\ell^{(p,m)}, S_k\mathbf{x}\rangle|^2,\quad (k,\ell)\in[K]_0\times[L]_0,$$

where the masks $\breve{\mathbf{m}}_\ell^{(p,m)}$ are defined as in (2.11). We may then reshape these measurements into a vector $\mathbf{y} \in \mathbb{C}^{KL}$ with entries given by

$$y_n := |\langle \breve{\mathbf{m}}_{n \bmod L}^{(p,m)}, S_{\lfloor \frac{n}{L} \rfloor} \mathbf{x} \rangle|^2, \quad \forall n \in [KL]_0. \tag{2.20}$$

After this reformulation, we may then apply a standard Wirtinger Flow Algorithm with spectral initialization. Full details are given below in Algorithm 2.2.

Algorithm 2.2 NFP Wirtinger Flow

Input:

1) Size $d \in \mathbb{N}$, number of iterations T, stepsizes $\mu_{\tau+1}$ for $\tau \in [T]_0$.[4]
2) PSF $\mathbf{p} \in \mathbb{C}^d$, mask $\mathbf{m} \in \mathbb{C}^d$, $\breve{\mathbf{m}}_\ell^{(p,m)} = \overline{S_\ell \widetilde{\mathbf{p}}} \circ \mathbf{m}$.
3) Noisy measurements $Y_{k,\ell} = |(\mathbf{p} * (S_k \mathbf{m} \circ \mathbf{x}))_\ell|^2 + N_{k,\ell}$.

Output: $\mathbf{x}_{\text{est}} \in \mathbb{C}^d$ with $\mathbf{x}_{\text{est}} \approx e^{i\theta}\mathbf{x}$ for some $\theta \in [0, 2\pi]$

1) Rearrange measurement matrix to form measurement vector $\mathbf{y}$ in (2.20).
2) Compute $\mathbf{z}_0$ using spectral method (Algorithm 1 in [14]).
3) For $\tau \in [T]_0$, let $\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \dfrac{\mu_{\tau+1}}{\|\mathbf{z}_0\|^2} \nabla f(\mathbf{z}_\tau)$ where

$$f(\mathbf{z}) := \frac{1}{KL} \sum_{n=1}^{KL} \left(\left| \left(S_{-\lfloor \frac{n}{L} \rfloor} \breve{\mathbf{m}}_{n \bmod L}^{(p,m)} \right)^* \mathbf{z} \right|^2 - y_n \right)^2.$$

4) Return $\mathbf{x}_{\text{est}} = \mathbf{z}_T$.

2.6 Numerical Simulations

In this section, we evaluate Algorithms 2.1 and 2.2 with respect to both noise robustness and runtime. Every data point in the plots below reports an average reconstruction error or runtime over 100 tests. For each test, a new sample $\mathbf{x} \in \mathbb{C}^d$ is randomly generated by choosing each entry to have independent and identically distributed (i.i.d.) mean 0 and variance 1 Gaussian real and imaginary parts. We then attempt to recover this sample from the noisy measurements $Y_{k,\ell}(\mathbf{x})$ defined as in (2.2) where the additive noise matrices $\mathbf{N}$ also have i.i.d. mean 0 Gaussian entries.

In our noise robustness experiments, we plot the reconstruction error as a function of the

[4] For our numerical simulations in Section 2.6, we set $\mu_\tau = \min(1 - e^{-\tau/330}, 0.4)$ as suggested in [14].

Signal-to-Noise Ratio (SNR), where we define the reconstruction error by

$$Error(\mathbf{x}, \mathbf{x}_{\text{est}}) := 10 \log_{10} \left(\frac{\min_\phi \|\mathbf{x} - e^{\mathbb{i}\phi} \mathbf{x}_{\text{est}}\|_2^2}{\|\mathbf{x}\|_2^2} \right),$$

and the SNR by

$$SNR(\mathbf{Y}, \mathbf{N}) := 10 \log_{10} \left(\frac{\|\mathbf{Y} - \mathbf{N}\|_F}{\|\mathbf{N}\|_F} \right).$$

In these experiments, we re-scale the noise matrix $\mathbf{N}$ in order to achieve each desired SNR level. All simulations were performed using MATLAB R2021b on an Intel desktop with a 2.60GHz i7-10750H CPU and 16GB DDR4 2933MHz memory. All code used to generate the figures below is publicly available at https://github.com/MarkPhilipRoach/NearFieldPtychography.

2.6.1 Algorithms 2.1 and 2.2 for Locally Supported Masks and Periodic Point Spread Functions

In these experiments, we choose the measurement index set for (2.2) to be $\mathcal{S} = \mathcal{K} \times \mathcal{L}$ where $\mathcal{K} = [d]_0$ and $\mathcal{L} = [2\delta - 1]_0$. As a consequence we see that we consider all shifts $k \in [d]_0$ of the mask while observing only a portion of each resulting noisy near-field diffraction pattern $|\mathbf{p} * (S_k \mathbf{m} \circ \mathbf{x})|^2$ for each k. This corresponds to a physical imaging system where, e.g., the sample and (a smaller) detector are fixed while a localized probe with support size δ scans across the sample. Figure 2.1 evaluates the robustness and runtime of Algorithm 2.1 as a function of the SNR and mask support δ in this setting. Looking at Figure 2.1 one can see that noise robustness increases with the support size of the mask, δ, in exchange for mild increases in runtime.

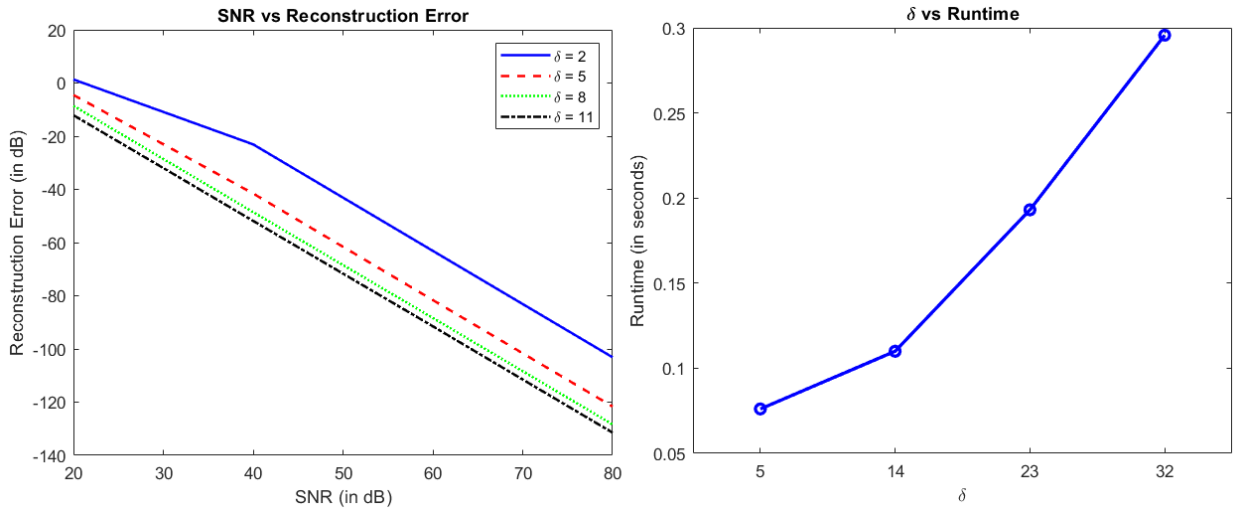

Figure 2.1 An evaluation of Algorithm 2.1 for the proposed PSF and mask with $d = 945$. Left: Reconstruction error vs SNR for various $\delta = |\ supp(\mathbf{m})|$. Right: Runtime as a function of δ.

Figure 2.2 compares the performance of Algorithm 2.1 and Algorithm 2.2 for the measurements proposed in Lemma 2.3.2. Looking at Figure 2.2 we can see that Algorithm 2.2 takes longer to achieve comparable errors to Algorithm 2.1 for these particular $\mathbf{p}$ and $\mathbf{m}$ as SNR increases. More specifically, we see, e.g., that BlockPR achieves a similar reconstruction error to 500 iterations of Wirtinger flow at an SNR of about 50 in a small fraction of the time. This supports the value of the BlockPR method as a fast initializer for more traditional optimization-based solution approaches.

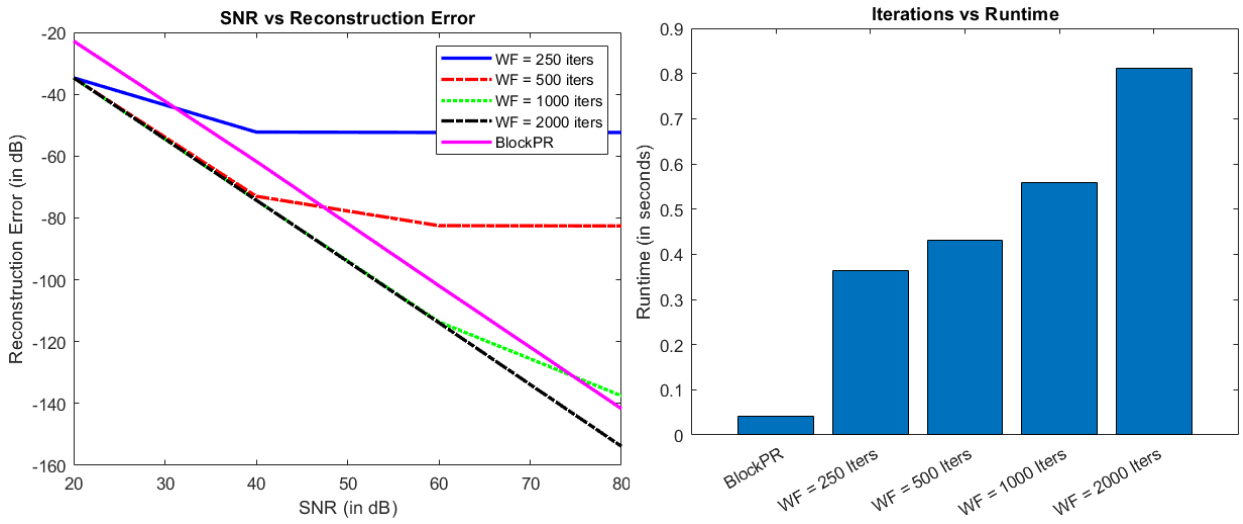

Figure 2.2 A comparison of Algorithms 2.1 and 2.2 for the proposed PSF and mask with $\delta = 26$ and $d = 102$. Left: Reconstruction error vs SNR for various numbers of Algorithm 2.2 iterations. Right: The corresponding average runtimes.

2.6.2 Algorithm 2.2 for Globally Supported Masks

As we saw in the previous section, Algorithm 2.1 is able to invert NFP measurements more efficiently than Algorithm 2.2 in situations where it is applicable. However, Algorithm 2.1 only applies to locally supported masks. In this section, we will show that Algorithm 2.2 remains effective even when the masks are globally supported, such as the masks considered in [124].

In Figure 2.3, we evaluate Algorithm 2.2 using noisy measurements of the form

$$Y_{k,\ell} = |(\mathbf{p} * (S_k \mathbf{m} \circ \mathbf{x}))_\ell|^2 + N_{k,l}, \quad (k,\ell) \in [K]_0 \times [d]_0. \tag{2.21}$$

Here $\mathbf{p} \in \mathbb{C}^d$ is a low-pass filter with $\widehat{\mathbf{p}} = S_{-(\gamma-1)/2}\mathbb{1}_\gamma$ where $\gamma = \frac{d}{3} + 1$ and $\mathbb{1}_\gamma \in \{0,1\}^d$ is a vector whose first γ entries are 1 and whose last $d - \gamma$ entries are 0. Here, we choose the mask $\mathbf{m}$ to have i.i.d. mean 0 variance 1 Gaussian entries. Thus, the measurements considered in (2.21) differ from those used in Section 2.6.1 in two crucial respects: i) the mask $\mathbf{m}$ here has global support. ii) we utilize a small number of mask shifts and observe the entire diffraction pattern resulting from each one (as opposed to observing just a portion of each diffraction pattern from all possible shifts, as

above). Examining Figure 2.3, one can see Algorithm 2.2 remains effective in this setting. We also observe, as expected, that using more shifts, i.e., collecting more measurements, results in lower reconstruction errors.

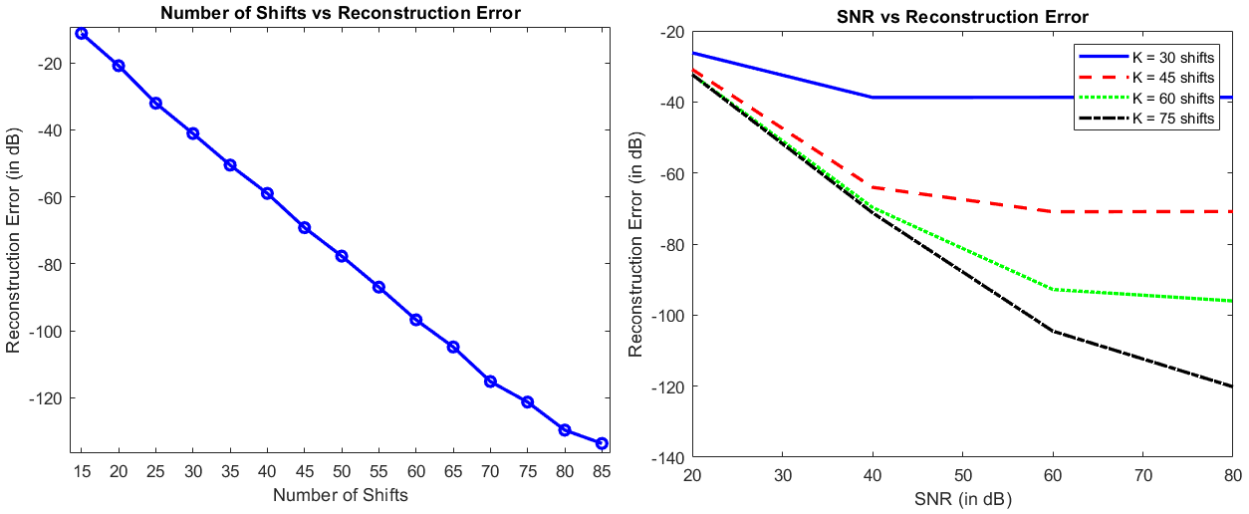

Figure 2.3 The reconstruction error of Algorithm 2.2 with $d = 102$, $\mathcal{L} = [d]_0$, and number of iterations $T = 2000$. Left: Reconstruction error vs the number of total shifts K for fixed $SNR = 80$. Right: Reconstruction error vs SNR for various numbers of shifts K.

2.6.3 Algorithm 2.1 for Non-Periodic PSFs via Remark 2.3.1

In these experiments, we choose the measurement index set for (2.2) to be $\mathcal{S}'$ from Remark 2.3.1 and consider two different non-periodic PSFs together with locally supported masks $\mathbf{m} \in \mathbb{C}^d$ having $supp(\mathbf{m}) \subseteq [\delta]_0$ for $\delta = 26$ and $d = 102$. Motivated again by [124], we first consider a PSF given by a low-pass filter (defined as in Section 2.6.2) plus small noise modeling imperfections (here the additive vector has i.i.d. $\mathcal{N}(0, 10^{-4})$ normal entries) in combination with a random symmetric mask. Here the mask's nonzero entries are created by reflecting $\delta/2 = 13$ random entries (chosen via i.i.d. mean 0 variance 1 Gaussians) across the middle of its support. The reconstruction error of Algorithm 2.1, as well as of Algorithm 2.2 initialized with the output of Algorithm 2.1, is plotted on the left in Figure 2.4 as a function of the NFP measurements' SNR for this PSF/mask pair.

For our second non-periodic PSF and locally supported mask pair, we let the PSF be a vector with unit magnitude entries having i.i.d. uniformly random phases, and let our locally supported

masks have δ nonzero i.i.d. mean 0 variance 1 Gaussian entries. The reconstruction errors of both Algorithm 2.1 and Algorithm 2.2 are plotted on the right in Figure 2.4 as a function of the NFP measurements' SNR in this case. In both experiments plotted in Figure 2.4, we note that both random initialization as well as the spectral initialization method from [14] appear insufficiently accurate to allow Algorithm 2.2 to converge. However, when the output of Algorithm 2.1 is used to compute z_0 in step 2 of Algorithm 2.2, Algorithm 2.2 then converges nicely to an accurate estimate of the true signal. This further reinforces the potential value of Algorithm 2.1 as fast and accurate initializer for more traditional optimization-based solution approaches.

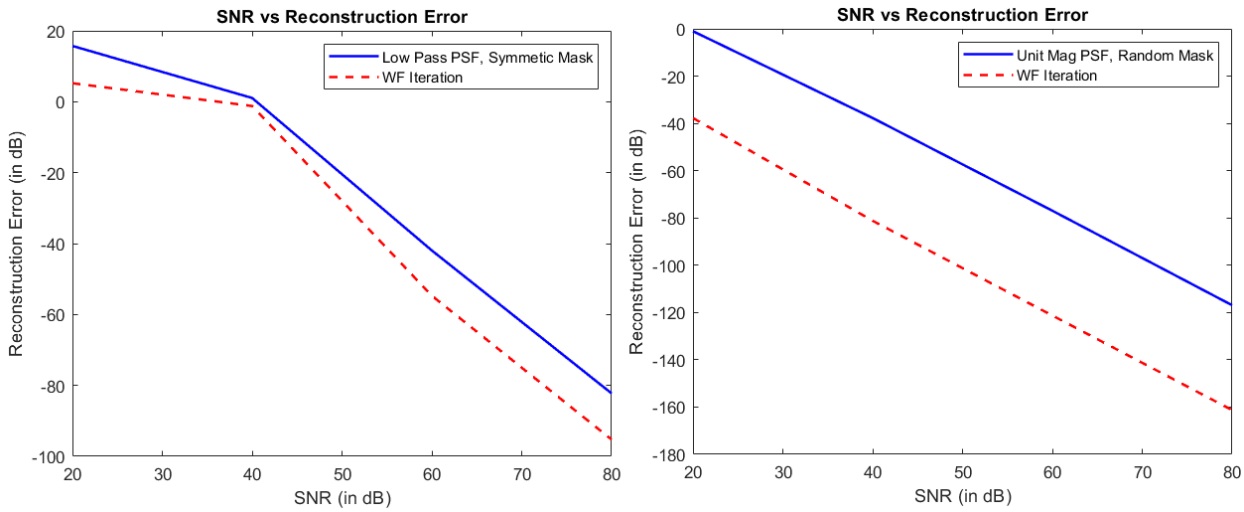

Figure 2.4 A simulation applying Algorithm 2.1 via Remark 2.3.1 and then using its generated estimate as the initial estimate z_0 in Algorithm 2.2. In both plots Algorithm 2.1 is plotted in solid blue, and Algorithm 2.2 is plotted in dashed red. Left: Reconstruction error vs SNR where the PSF is an low-pass filter with small additive noise, and the locally supported mask is symmetric and random. Here $T = 5000$ iterations are used in Algorithm 2.2. Right: Reconstruction error vs SNR where the PSF is an vector with randomized unit magnitude entries, and the locally supported mask is random. Here $T = 1000$ iterations are used in Algorithm 2.2.

2.7 Application of Algorithm 2.1

We aim to apply a real world application of Algorithm 2.1, where we aim to recover a nxn-pixel color image. The image is first broken down into its three colour channels on the RGB scale, and converting to three individual integer valued matrices with entries from 0 to 255 based on

the intensity of the color at each pixel. Each of these matrices is then separately reshaped into a column vector in $\mathbb{R}^{n^2}$, which is then used for the object x in Algorithm 2.1. Once we obtain our three estimates for the three column vectors, they are then reshaped back into $n \times n$ matrices and then combined together to form the color estimate of the original image.

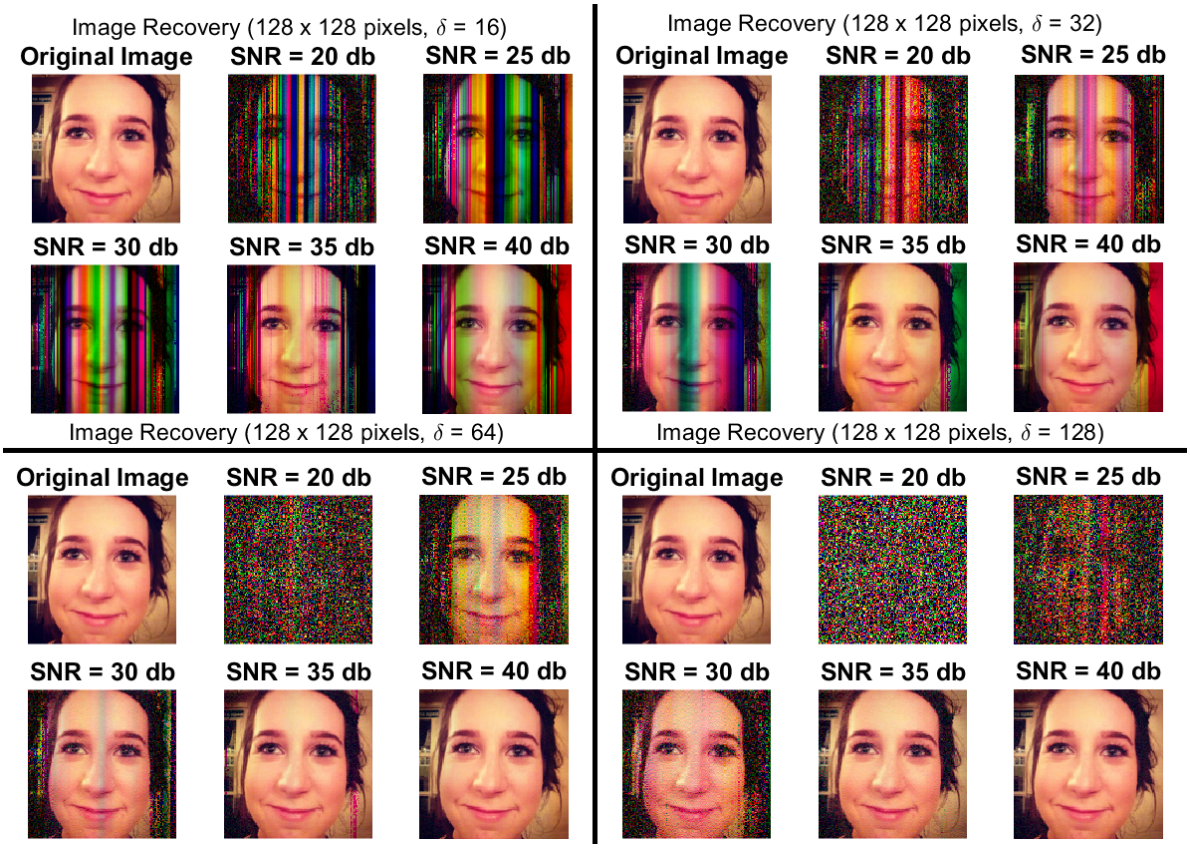

Figure 2.5 Here we have an example of this process in action. The original image is a 128×128 pixel color image. This would mean that $d = 128^2 = 16,384$ in Algorithm 2.1, however to ensure that d is divisible by $2\delta - 1$ for any given delta, we let d_{ext} be the smallest integer such that $d_{ext} \geq d$ and d_{ext} is divisible by $2\delta - 1$. We then extend the reshapen pixel data such that $x \in \mathbb{R}^{d_{ext}}$ by adding ones at the extended part and disregard this extension once we recover the estimate. We test this recovery for two levels of delta and for each delta, we apply varying levels of noise.

2.8 Conclusions and Future Work

We have introduced two new algorithms for recovering a specimen of interest from near-field ptychographic measurements. Both of these algorithms rely on first reformulating and reshaping our measurements so that they resemble widely-studied far-field ptychographic measurements. We

then recover our method using either Wirtinger Flow or using methods based on [53]. Algorithm 2.1 is computational efficient and, to the best of our knowledge, is the first algorithm with provable recovery guarantees for measurements of this form. Algorithm 2.2, on the other hand, has the advantage of being applied to more general masks with global support. Developing more efficient and provably accurate algorithms for this latter class of measurements remains an interesting avenue for future work.

CHAPTER 3

BLIND PTYCHOGRAPHY VIA BLIND DECONVOLUTION

Abstract

Ptychography involves a sample being illuminated by a coherent, localised probe of illumination. When the probe interacts with the sample, the light is diffracted and a diffraction pattern is detected. Then the probe or sample is shifted laterally in space to illuminate a new area of the sample while ensuring there is sufficient overlap. **Far-field Ptychography** occurs when there is a large enough distance (when the Fresnel number is $\ll 1$) to obtain magnitude-square Fourier transform measurements. In an attempt to remove ambiguities, masks are utilized to ensure unique outputs to any recovery algorithm are unique up to a global phase. In this paper, we assume that both the sample and the mask are unknown, and we apply blind deconvolutional techniques to solve for both. Numerical experiments demonstrate that the technique works well in practice, and is robust under noise.

This chapter is comprised of three sections. Section 3.2 introduces far-field Fourier ptychography, and an algorithm for solving given noisy ptychographic measurements. In particular of use to us will be Theorem 3.2.2 which reformulates the measurements into a convolution. Section 3.3 explores a method for solving a blind deconvolution problem, given certain appropriate real-world assumptions. Section 3.4 combines the previous two sections, taking the reformulated convolutional measurements, assuming that both components are unknown, and then applying the blind deconvolution recovery algorithm. The full algorithm is stated and numerical simulations are summarized, outlining good recovery which is robust under noise.

3.1 Introduction

Ptychography involves a sample being illuminated by a coherent, localised probe of illumination. When the probe interacts with the sample, the light is diffracted and a diffraction pattern is detected. Then the probe or sample is shifted laterally in space to illuminate a new area of the

sample while ensuring there is sufficient overlap. **Far-field Ptychography** occurs when there is a large enough distance (when the Fresnel number is $\ll 1$) to obtain magnitude-square Fourier transform measurements.

Ptychography was initially studied in the late 1960s ([45]), with the problem solidified in 1970 ([44]). The name "Ptychography" was coined in 1972 ([43]), after the Greek word *to fold* because the process involves an interference pattern such that the scattered waves fold into one anotherthe (coherent) Fourier diffraction pattern of the object. Initially developed to study crystalline objects under a scanning transmission electron microscope, since then the field has widen to setups such as using visible light ([19], [47], [84]), x-rays ([26], [114], [89]), or electrons ([122],[37][57]). It is benefited from being unaffected by lens-induced aberrations or diffraction effects unlike conventional lens imaging. Various types of ptychography are studied based on the optical configuration of the experiments. For instance, Bragg Ptychography ([38] [108], [46], [69]) measures strain in crystalline specimens by shifting the surface of the specimen.

Fourier ptychography ([126],[113],[87],[127]) consists of taking multiple images at a wide field-of-view then computationally synthesizing into a high-resolution image reconstruction in the Fourier domain. This results in an increased resolution compared to a conventional microscope.

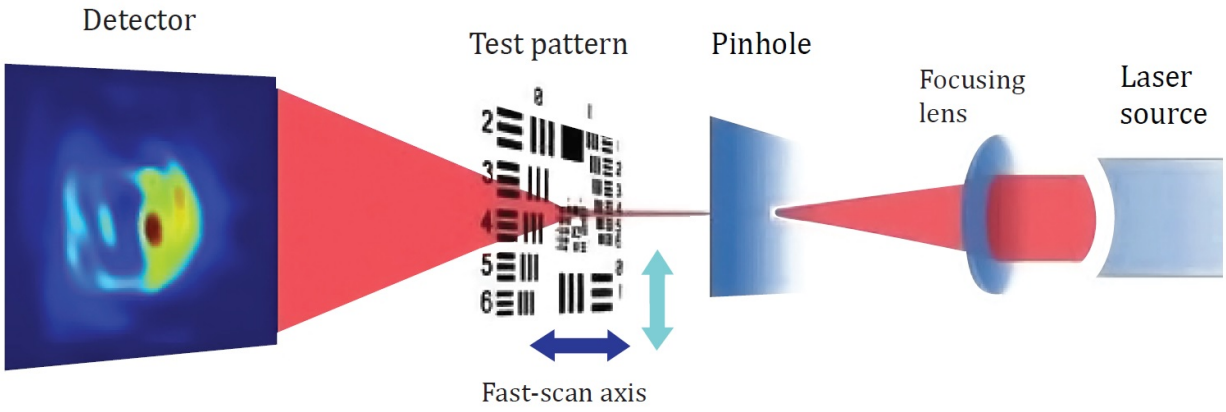

Figure 3.1 [47] Experimental setup for fly-scan ptychography

3.2 Far-field Fourier Ptychography

Let $\mathbf{x}, \mathbf{m} \in \mathbb{C}^d$ denote the unknown sample and known mask, respectively. We suppose that we have d^2 noisy ptychographic measurements of the form

$$(\mathbf{Y})_{\ell,k} = |(\mathbf{F}(\mathbf{x} \circ S_k\mathbf{m}))_\ell|^2 + (\mathbf{N})_{\ell,k}, \quad (\ell, k) \in [d]_0 \times [d]_0, \tag{3.1}$$

where $S_k, \circ, \mathbf{F} := \mathbf{F}_d$ denote k^{th} circular shift, Hadamard product, and d-dimensional discrete Fourier transform, and $\mathbf{N}$ is the matrix of additive noise.

In this section, we will define a discrete Wigner distribution deconvolution method for recovering a discrete signal. A modified Wigner distribution deconvolution approach is used to solve for an estimate of $\hat{\mathbf{x}}\hat{\mathbf{x}}^* \in \mathbb{C}^{d\times d}$ and then angular synchronization is performed to compute estimate of $\hat{\mathbf{x}}$ and thus $\mathbf{x}$.

In Section 3.2.1, we introduce definitions and techincal lemmas which will be of use. In particular, the decoupling lemma (Lemma 3.2.2) allows use to effectively 'separate' the mask and object from a convolution. In Section 3.2.2, these technical lemmas are applied to the ptychographic measurements to write the problem as a decoupled deconvolution problem, the blind variant of which will be studied later on. In Section 3.2.3, an additional Fourier transform is applied and the measurements have been rewritten to a form in which a pointwise division approach can be applied. Sub-sampled version of this theorem are also given. We then state the full algorithm for recovering the sample.

3.2.1 Properties of the Discrete Fourier Transform

We firstly define the modulation operator.

Definition 3.2.1. *Given $k \in [d]_0$, define the **modulation operator** $W_k : \mathbb{C}^d \longrightarrow \mathbb{C}^d$ component-wise via*

$$(W_k\mathbf{x})_n = x_n e^{2\pi ikn/d}, \quad \forall n \in [d]_0. \tag{3.2}$$

From this definition, we can develop some useful equalities which we will use in the main proofs of this section.

Lemma 3.2.1. ***(Technical Equalities)*** *(Lemma 1.3.1., [77]) The following equalities hold for all* $\mathbf{x} \in \mathbb{C}^d, [\ell] \in [d]_0$*:*

(i) $\mathbf{F}_d\hat{\mathbf{x}} = d \cdot \tilde{\mathbf{x}}$*;*

(ii) $\mathbf{F}_d(W_\ell \mathbf{x}) = S_{-\ell}\hat{\mathbf{x}}$*;*

(iii) $\mathbf{F}_d(S_\ell \mathbf{x}) = W_\ell \widehat{\mathbf{x}}$*;*

(iv) $W_{-\ell}\mathbf{F}_d(S_\ell \bar{\tilde{\mathbf{x}}}) = \bar{\hat{\mathbf{x}}}$*;*

(v) $\widetilde{\overline{S_\ell \mathbf{x}}} = S_{-\ell}\bar{\tilde{\mathbf{x}}}$*;*

(vi) $\mathbf{F}_d\bar{\mathbf{x}} = \overline{\mathbf{F}_d\tilde{\mathbf{x}}}$*;*

(vii) $\tilde{\hat{\mathbf{x}}} = \mathbf{F}_d\tilde{\mathbf{x}}$.

We wish to be able to convert between the convolution and the Hadamard product, so we will need this useful theorem.

Theorem 3.2.1. ***(Discretized Convolution Theorem)*** *(Lemma 1.3.2., [77]) Let* $\mathbf{x}, \mathbf{y} \in \mathbb{C}^d$*. We have that*

(i) $F_d^{-1}(\hat{\mathbf{x}} \circ \hat{\mathbf{y}}) = \mathbf{x} *_d \mathbf{y}$*;*

(ii) $(\mathbf{F}_d\mathbf{x}) *_d (\mathbf{F}_d\mathbf{y}) = d \cdot \mathbf{F}_d(\mathbf{x} \circ \mathbf{y})$.

Currently, the measurements we are dealing with will be having the specimen and the mask intertwined. We introduce the decoupling lemma to essentially detangle the two.

Lemma 3.2.2. ***(Decoupling Lemma)*** *(Lemma 1.3.3., [77])*

Let $\mathbf{x}, \mathbf{y} \in \mathbb{C}^d, \ell, k \in [d]_0$*. Then*

$$\left((\mathbf{x} \circ S_{-\ell}\mathbf{y}) *_d (\bar{\tilde{\mathbf{x}}} \circ S_\ell \bar{\tilde{\mathbf{y}}})\right)_k = \left((\mathbf{x} \circ S_{-k}\bar{\mathbf{x}}) *_d (\tilde{y} \circ S_k \bar{\tilde{\mathbf{y}}})\right)_\ell. \tag{3.3}$$

Proof. Let $\mathbf{x}, \mathbf{y} \in \mathbb{C}^d, \ell, k \in [d]_0$. By the definitions of the circular convolution, Hadamard product and shift operator, we have that

$$\begin{aligned}
\left((\mathbf{x} \circ S_{-\ell}\mathbf{y}) *_d (\bar{\tilde{\mathbf{x}}} \circ S_\ell \bar{\tilde{\mathbf{y}}})\right)_k &= \sum_{n=0}^{d-1} (\mathbf{x} \circ S_{-\ell}\mathbf{y})_n ((\bar{\tilde{\mathbf{x}}} \circ S_\ell \bar{\tilde{\mathbf{y}}})_{k-n} \\
&= \sum_{n=0}^{d-1} x_n y_{n-\ell} \bar{\tilde{x}}_{k-n} \bar{\tilde{y}}_{\ell+k-n} \\
&= \sum_{n=0}^{d-1} x_n \bar{x}_{n-k} \tilde{y}_{\ell-n} \bar{\tilde{y}}_{k+\ell-n} \\
&= \sum_{n=0}^{d-1} (\mathbf{x} \circ S_{-k}\mathbf{x})_n ((\tilde{\mathbf{y}} \circ S_k \bar{\tilde{\mathbf{y}}})_{\ell-n} \\
&= \left((\mathbf{x} \circ S_{-k}\bar{\mathbf{x}}) *_d (\tilde{y} \circ S_k \bar{\tilde{\mathbf{y}}})\right)_\ell.
\end{aligned} \tag{3.4}$$

□

Lastly before entering the main part of this subsection, we need a lemma involving looking at how the Fourier squared magnitude measurements will relate to a convolution.

Lemma 3.2.3. *Let* $\mathbf{x} \in \mathbb{C}^d$. *We have that*

$$|\mathbf{F}_d \mathbf{x}|^2 = \mathbf{F}_d(\mathbf{x} *_d \bar{\tilde{\mathbf{x}}}). \tag{3.5}$$

Proof. Let $\mathbf{x} \in \mathbb{C}^d$. Then we have that

$$|\mathbf{F}_d \mathbf{x}|^2 = (\mathbf{F}_d \mathbf{x}) \circ \overline{(\mathbf{F}_d \mathbf{x})} = (\mathbf{F}_d \mathbf{x}) \circ (\mathbf{F}_d \bar{\tilde{\mathbf{x}}}) = \mathbf{F}_d(\mathbf{x} *_d \bar{\tilde{\mathbf{x}}}). \tag{3.6}$$

□

3.2.2 Discretized Wigner Distribution Deconvolution

We now prove the Discretized Wigner Distribution Deconvolution theorem which will allow us to convert the measurements into a form in which we can algorithmically solve.

Theorem 3.2.2. *(Lemma 1.3.5., [77]) Let* $\mathbf{x}, \mathbf{m} \in \mathbb{C}^d$ *denote the unknown specimen and known mask, respectively. Suppose we have* d^2 *noisy ptychographic measurements of the form*

$$(\mathbf{y}_\ell)_k = \left| \sum_{n=0}^{d-1} x_n m_{n-\ell} e^{-2\pi i n k/d} \right|^2 + (\mathbf{N})_{\ell,k}, \quad (\ell, k) \in [d]_0 \times [d]_0. \tag{3.7}$$

Let $\mathbf{Y} \in \mathbb{R}^{d\times d}, \mathbf{N} \in \mathbb{C}^{d\times d}$ *be the matrices whose* ℓ^{th} *column is* $\mathbf{y}_\ell, \mathbf{N}_\ell$ *respectively. Then for any* $k \in [d]_0$*,*

$$\left(\mathbf{Y}^T \mathbf{F}_d^T\right)_k = d \cdot (\mathbf{x} \circ S_k \bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}} \circ S_{-k} \bar{\tilde{\mathbf{m}}}) + \left(\mathbf{N}^T \mathbf{F}_d^T\right)_k. \tag{3.8}$$

Proof. Let $\ell \in [d]_0$. We have that

$$\mathbf{y}_\ell = |F_d(\mathbf{x} \circ S_{-\ell}\mathbf{m})|^2 + \mathbf{N}_\ell = F_d\Big((\mathbf{x} \circ S_{-\ell}\mathbf{m}) *_d (\bar{\tilde{\mathbf{x}}} \circ S_\ell \bar{\tilde{\mathbf{m}}})\Big) + \mathbf{N}_\ell. \tag{3.9}$$

Taking Fourier transform of both sides at $k \in [d]_0$ and using that $\mathbf{F}_d \hat{\mathbf{x}} = d \cdot \tilde{\mathbf{x}}$ yields

$$\begin{aligned} (\mathbf{F}_d \mathbf{y}_\ell)_k &= d \cdot \Big((\mathbf{x} \circ S_{-\ell}\mathbf{m}) *_d (\bar{\tilde{\mathbf{x}}} \circ S_\ell \bar{\tilde{\mathbf{m}}})\Big)_{-k} + (\mathbf{F}_d \mathbf{N}_\ell)_k \\ &= d \cdot \Big((\mathbf{x} \circ S_k \bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}} \circ S_{-k} \bar{\tilde{\mathbf{m}}})\Big)_\ell + (\mathbf{F}_d \mathbf{N}_\ell)_k \end{aligned} \tag{3.10}$$

by previous lemma. For fixed $\ell \in [d]_0$, the vector $\mathbf{F}_d \mathbf{y}_\ell \in \mathbb{C}^d$ is the ℓ^{th} column of the matrix $\mathbf{F}_d \mathbf{Y}$, thus its transpose $\mathbf{y}_\ell^T \mathbf{F}_d^T \in \mathbb{C}^d$ is the ℓ^{th} row of the matrix $(\mathbf{F}_d \mathbf{Y})^T$. Similarly, $((\mathbf{N})_\ell)^T \mathbf{F}_d^T \in \mathbb{C}^d$ is the ℓ^{th} row of $(\mathbf{F}_d \mathbf{N})^T$. Thus we have that

$$\left(\left(\mathbf{Y}^T \mathbf{F}_d^T\right)_k\right)_\ell = d \cdot \Big((\mathbf{x} \circ S_k \bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}} \circ S_{-k} \bar{\tilde{\mathbf{m}}})\Big)_\ell + \left(\mathbf{N}^T \mathbf{F}_d^T\right)_{k,\ell}. \tag{3.11}$$

Thus we have that

$$\left(\mathbf{Y}^T \mathbf{F}_d^T\right)_k = d \cdot (\mathbf{x} \circ S_k \bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}} \circ S_{-k} \bar{\tilde{\mathbf{m}}}) + \left(\mathbf{N}^T \mathbf{F}_d^T\right)_k. \tag{3.12}$$

□

We note that $\mathbf{x} \circ S_k \bar{\mathbf{x}}$ is a diagonal of $\mathbf{x}\mathbf{x}^*$.

3.2.3 Wigner Distribution Deconvolution Algorithm

We suppose that the mask is known and the specimen is unknown. By taking an additional Fourier transform and using the discretized convolution theorem, we have these variances of the previous lemmas.

Theorem 3.2.3. ***(Discretized Wigner Distribution Deconvolution)*** *Let $\mathbf{x}, \mathbf{m} \in \mathbb{C}^d$ denote the unknown specimen and known mask, respectively. Suppose we have d^2 noisy spectrogram measurements of the form*

$$(\mathbf{y}_\ell)_k = \left| \sum_{n=0}^{d-1} x_n m_{n-\ell} e^{-2\pi i n k/d} \right|^2 + (\mathbf{N})_{\ell,k}, \quad (\ell, k) \in [d]_0 \times [d]_0. \tag{3.13}$$

Let $\mathbf{Y} \in \mathbb{R}^{d \times d}$ be the matrix whose ℓ^{th} column is $\mathbf{y}_\ell$. Then for any $k \in [d]_0$

$$\mathbf{F}_d \left(\mathbf{Y}^T \mathbf{F}_d^T \right)_k = d \cdot \mathbf{F}_d(\mathbf{x} \circ S_k \bar{\mathbf{x}}) \circ \mathbf{F}_d(\tilde{\mathbf{m}} \circ S_{-k} \bar{\tilde{\mathbf{m}}}) + \mathbf{F}_d \left(\mathbf{N}^T \mathbf{F}_d^T \right)_k. \tag{3.14}$$

We also have a similar result based on the work in Appendix B.

Lemma 3.2.4. ***(Sub-Sampling In Frequency)*** *Suppose that the spectrogram measurements are collected on a subset $\mathcal{K} \subseteq [d]_0$ of K equally spaced Fourier modes. Then for any $\omega \in [K]_0$*

$$\mathbf{F}_d \left((\mathbf{Y}_{K,d})^T \mathbf{F}_K^T \right)_\omega = K \sum_{r=0}^{\frac{d}{K}-1} \mathbf{F}_d(\mathbf{x} \circ S_{\ell L - \alpha} \bar{\mathbf{x}}) \circ \mathbf{F}_d(\tilde{\mathbf{m}} \circ S_{\alpha - \ell L} \bar{\tilde{\mathbf{m}}}) + \mathbf{F}_d \left((\mathbf{N}_{K,d})^T \mathbf{F}_K^T \right)_\omega$$

where $\mathbf{Y}_{K,d} \in \mathbb{C}^{K \times d}$ is the matrix of sub-sampled noiseless $K \cdot d$ measurements.

Lemma 3.2.5. ***(Sub-Sampling In Frequency And Space)*** *Suppose we have spectrogram measurements collected on a subset $\mathcal{K} \subseteq [d]_0$ of K equally spaced frequencies and a subset $\mathcal{L} \subseteq [d]_0$ of L*

equally spaced physical shifts. Then for any $\omega \in [K]_0, \alpha \in [L]_0$

$$\left(\mathbf{F}_L(\mathbf{Y}_{K,L})^T(\mathbf{F}_K^T)_\omega\right)_\alpha = \frac{KL}{d^3}\sum_{r=0}^{\frac{d}{K}-1}\sum_{\ell=0}^{\frac{d}{L}-1}\left(\mathbf{F}_d(\hat{\mathbf{x}}\circ S_{\ell L-\alpha}\bar{\hat{\mathbf{x}}})\right)_{\omega-rK}\left(F_d(\hat{\mathbf{m}}\circ S_{\alpha-\ell L}\bar{\hat{\mathbf{m}}})\right)_{\omega-rK} + \left(\mathbf{F}_L(\mathbf{N}_{K,L})^T(\mathbf{F}_K^T)_\omega\right)_\alpha,$$

where $\mathbf{Y}_{K,L} \in \mathbb{C}^{K\times L}$ *is the matrix of sub-sampled noiseless* $K \cdot L$ *measurements.*

Assume that $\mathbf{m}$ is band-limited with $supp(\hat{\mathbf{m}}) = [\delta]_0$ for some $\delta \ll d$. Then the algorithm below allows for the recovery of an estimate of $\hat{\mathbf{x}}$ from spectrogram measurements via Wigner distribution deconvolution and angular synchronization.

Algorithm 3.1 (Algorithm 1, [77]) Wigner Distribution Deconvolution Algorithm

Input: 1) $Y_{d,L} \in \mathbb{C}^{d\times L}$, matrix of noisy measurements.
2) Mask $\mathbf{m} \in \mathbb{C}^d$ with $supp(\hat{\mathbf{m}}) = [\delta]$.
3) Integer $\kappa \leq \delta$, so that $2\kappa - 1$ diagonals of $\hat{\mathbf{x}}\hat{\mathbf{x}}^*$ are estimated, and $L = \delta + \kappa - 1$.

Output: An estimate $\mathbf{x}_{est}$ of $\mathbf{x}$ up to a global phase.
1) Perform pointwise division to compute

$$\frac{1}{d}\frac{\mathbf{F}_d\left(\mathbf{Y}^T\mathbf{F}_d^T\right)_k}{\mathbf{F}_d^{-1}(\tilde{\mathbf{m}}\circ S_{-k}\bar{\tilde{\mathbf{m}}})}. \tag{3.15}$$

2) Invert the $(2\kappa - 1)$ Fourier transforms above.
3) Organize values from step 2 to form the diagonals of a banded matrix $Y_{2\kappa-1}$.
4) Perform angular synchronization on $Y_{2\kappa-1}$ to obtain $\hat{\mathbf{x}}_{est}$.
5) Let $\mathbf{x}_{est} = \mathbf{F}_d^{-1}\hat{\mathbf{x}}_{est}$.

When the mask is known with $supp(\hat{\mathbf{m}}) = [\delta]_0$, $\delta \ll d$, maximum error guarantees (Theorem 2.1.1., [77]) are given depending on $\mathbf{x}$, d, κ, L, $\|N_{d,L}\|_F$ (the matrix formed by the noise) and the mask dependent constant $\mu > 0$,

$$\mu := \min_{|p|\leq\kappa-1, q\in[d]_0}\left|(F_d(\hat{\mathbf{m}}\circ S_p\bar{\hat{\mathbf{m}}}))_q\right|. \tag{3.16}$$

In the next section, we look at the situation in which both the specimen and mask are unknown. Since we have already shown that we can rewrite the Fourier squared magnitude measurements as convolutions between the shifted autocorrelations, then the next obvious step is when both the

specimen and mask are unknown. This is the topic of **blind deconvolution**, which seeks to recover vectors from their deconvolution. In particular, we will look at a couple of approaches which involve making assumptions based on real world applications.

3.3 Blind Deconvolution

3.3.1 Introduction

Blind Deconvolution is a problem that has been mathematically considered for decades, from more past work ([4], [60], [111], [34][78], [64]) to more recent work ([15], [71], [2], [35] [70]), and summarized in [68]. The goal is to recover a sharp image from an initial blurry image. The first application to compressive sensing was considered in [2].

We consider one-dimensional, discrete, noisy measurements of the form $\mathbf{y} = \mathbf{f} * \mathbf{g} + \mathbf{n}$, where the $\mathbf{f}$ is considered to be an object, signal, or image of consideration. $\mathbf{g}$ is considered to be a blurring, masking, or point-spread function. $\mathbf{n}$ is considered to be the noise vector. $*$ refers to circulant convolution[1]. We consider situations when both $\mathbf{f}$ and $\mathbf{g}$ are unknown. The process of recovering the object and blurring function can be generalized to two-dimensional measurements.

The problem of estimating the unknown blurring function and unknown object simultaneously is known as **blind image restoration** ([123], [121], [63], [104]). Although strictly speaking, **blind deconvolution** refers to the noiseless model of recovering $\mathbf{f}$ and $\mathbf{g}$ from $\mathbf{y} = \mathbf{f} * \mathbf{g}$, the noisy model is commonly refered to as blind deconvolution itself, and this notation will be continued in this chapter.

As we will show later, the problem is ill-posed and ambiguities lead to no unique solution to the pair being viable from any approach.

In Section 3.3.2, we consider the underlying measuerements and assumptions that we will consider. We then show how through manipulation, that we can re-write the original problem as a minimization of a non-convex linear function. In Section 3.3.3, we demonstrate an iterative approach to the minimization problem, in particular, by applying Wirtinger Gradient Descent. In

[1] $*$ should refer to ordinary convolution, but for $\mathbf{g}$ that will be considered in this chapter, circulant convolution will be sufficient.

Section 3.3.5, we outline the initial estimate used for this gradient descent and fully layout the algorithm that will apply in our numerical simulations. In Section 3.3.6, we will look at the recovery guarantees that currently exist for this approach. Finally, in Section 3.3.7, we consider the key conditions used to generate the main recovery theorem, and where further work could be done to generalize these conditions and ultimately, allow more guarantees of recovery.

3.3.2 Blind Deconvolution Model

We now want to approach the blind ptychography problem in which both the mask and specimen are unknown. Using the lemmas in the previous section, we can see that this would reduce to solving a blind deconvolution problem.

Definition 3.3.1. *We consider the blind deconvolution model*

$$\mathbf{y}' = \mathbf{f} * \mathbf{g} + \mathbf{n}, \quad \mathbf{y}, \mathbf{f}, \mathbf{g}, \mathbf{n} \in \mathbb{C}^d,$$

where $\mathbf{y}$ *are blind deconvolutional measurements,* $\mathbf{f}$ *is the unknown blurring function (which serves a similar role as to our phase retrieval masks),* $\mathbf{n}$ *is the noise, and* $\mathbf{g}$ *is the signal (which serves a similar role as to our phase retrieval object). Here* $*$ *denotes circular convolution.*

We will base our work on the algorithm suggested in [70], considering the assumptions used. In [70], the authors impose general conditions on $\mathbf{f}$ and $\mathbf{g}$ that are not restricted to any particular application but allows for flexibility. They also assume that $\mathbf{f}$ and $\mathbf{g}$ belong to known linear subspaces.

For the blurring function, it is assumed that $\mathbf{f}$ is either compactly supported, or that $\mathbf{f}$ decays sufficiently fast so that it can be well approximated by a compactly supported function. Therefore, we make the assumption that $\mathbf{f} \in \mathbb{C}^d$ satisfies

$$\mathbf{f} := \begin{bmatrix} \mathbf{h} \\ \mathbf{0}_{d-K} \end{bmatrix},$$

for some $k \ll d, \mathbf{h} \in \mathbb{C}^K$. This again reinforces the notion that the blurring function is analogous to our masking function since both are compactly supported.

Figure 3.2 [35] An example of an image deblurring by solving the deconvolution

For the signal, it is assumed that $\mathbf{g}$ belongs to a linear subspace spanned by the columns of a known matrix $\mathbf{C}$, i.e., $\mathbf{g} = \mathbf{C}\bar{\mathbf{x}}$ for some matrix $\mathbf{C} \in \mathbb{C}^{d\times N}, N \ll d$. This will lead to an additional restriction we have to place on our blind ptychography but one for which there are real world applications for which this assumption makes reasonable sense.

In [70], the authors use that $\mathbf{C}$ is a Gaussian random matrix for theoretical guarantees although they demonstrated in numerical simulations that this assumption is not necessary to gain results. In particular, they found good results for when $\mathbf{C}$ represents a wavelet subspace (suitable for images) or when $\mathbf{C}$ is a Hadamard-type matrix (suitable for communications).

We assume the noise is complex Gaussian, i.e. $\mathbf{n} \sim \mathcal{N}(0, \frac{\sigma^2 L_0^2}{2} I_d) + i\mathcal{N}(0, \frac{\sigma^2 L_0^2}{2} I_d)$ is a complex Gaussian noise vector, where $L_0 = \|\mathbf{h}_0\| \cdot \|\mathbf{x}_0\|$, and $\mathbf{h}_0, \mathbf{x}_0$ are the true blurring function and signal. σ^{-2} represents the SNR.

The goal is convert the problem into one which can be algorithmically solvable via gradient descent.

Proposition 3.3.1. *[70] Let $\mathbf{F}_d \in C^{d\times d}$ be DFT matrix. Let $\mathbf{B} \in \mathbb{C}^{d\times K}$ denote the first K columns of $\mathbf{F}_d$. Then we have that*

$$\mathbf{y} = \mathbf{Bh} \circ \overline{\mathbf{Ax}} + \mathbf{e}, \tag{3.17}$$

where $\mathbf{y} = \frac{1}{\sqrt{d}}\widehat{\mathbf{y}'}$, $\bar{\mathbf{A}} = \mathbf{FC} \in \mathbb{C}^{d\times N}$, and $\mathbf{e} = \frac{1}{\sqrt{d}}\mathbf{F_d n}$ represents noise.

Proof. By the unitary property of $\mathbf{F}_d$, we have that $\mathbf{B}^*\mathbf{B} = \mathbf{I}_K$.By applying the scaled DFT matrix $\sqrt{L}\mathbf{F}_d$ to both sides of the convolution, we have that

$$\sqrt{L}\mathbf{F_d y} = (\sqrt{L}\mathbf{F_d f}) \circ (\sqrt{L}\mathbf{F_d g}) + \sqrt{L}\mathbf{F_d n}.$$

Additionally, we have that

$$\mathbf{F_d f} = \begin{bmatrix}\mathbf{B} & \mathbf{M}\end{bmatrix}\begin{bmatrix}\mathbf{h} \\ \mathbf{0}_{d-K}\end{bmatrix} = \mathbf{Bh},$$

and we let $\bar{\mathbf{A}} = \mathbf{FC} \in \mathbb{C}^{L\times N}$. Since $\mathbf{C}$ is Gaussian, then $\bar{\mathbf{A}} = \mathbf{FC}$ is also Gaussian. In particular,

$$\bar{\mathbf{A}}_{ij} \sim \mathcal{N}(0, \frac{1}{2}) + i\mathcal{N}(0, \frac{1}{2}).$$

Thus by dividing by d, the problem converts to

$$\frac{1}{\sqrt{d}}\widehat{\mathbf{y}'} = \mathbf{Bh} \circ \overline{\mathbf{Ax}} + \mathbf{e},$$

where $\mathbf{e} = \frac{1}{\sqrt{d}}\mathbf{F_L n} \sim \mathcal{N}(0, \frac{\sigma^2 L_0^2}{2L}\mathbf{I}_d) + i\mathcal{N}(0, \frac{\sigma^2 L_0^2}{2L}\mathbf{I}_d)$ serves as complex Gaussian noise. Hence by letting $\mathbf{y} = \frac{1}{\sqrt{d}}\widehat{\mathbf{y}'}$, we arrive at

$$\mathbf{y} = \mathbf{Bh} \circ \overline{\mathbf{Ax}} + \mathbf{e}.$$

□

We have thus transformed the original blind deconvolution model as a hadamard product. This form of the problem is used in the rest of the section, where $\mathbf{y} \in \mathbb{C}^d, \mathbf{B} \in \mathbb{C}^{d\times K}, \mathbf{A} \in \mathbb{C}^{d\times N}$ are given. Our goal is to recover $\mathbf{h}_0$ and $\mathbf{x}_0$.

There are inherent ambiguities to the problem however. If $(\mathbf{h}_0, \mathbf{x}_0)$ is a solution to the blind deconvolution problem, then so is $(\alpha\mathbf{h}_0, \alpha^{-1}\mathbf{x}_0)$ for any non-zero constant α. For most real world applications, this is not an issue. Thus for uniformity, it is assumed that $\|\mathbf{h}_0\| = \|\mathbf{x}_0\| = \sqrt{L_0}$.

Definition 3.3.2. *We define the matrix-valued linear operator $\mathcal{A} : \mathbb{C}^{K\times N} \longrightarrow \mathbb{C}^d$ by*

$$\mathcal{A}(Z) := \{\mathbf{b}_\ell^* Z \mathbf{a}_\ell\}_{\ell=1}^d,$$

where $\mathbf{b}_k$ denotes the k-th column of $\mathbf{B}^*$, and $\mathbf{a}_k$ is the k-th column of $\mathbf{A}^*$. We also define the corresponding adjoint operator $\mathcal{A}^* : \mathbb{C}^d \longrightarrow \mathbb{C}^{K\times N}$, given by

$$\mathcal{A}^*(\mathbf{z}) := \sum_{k=1}^d \mathbf{z}_k \mathbf{b}_k \mathbf{a}_k^*.$$

We see that this translates to a lifting problem, where

$$\sum_{k=1}^d \mathbf{b}_\ell \mathbf{b}_\ell^* = \mathbf{B}^*\mathbf{B} = \mathbf{I}_K, \quad \|\mathbf{b}_\ell\| = \frac{K}{d}, \quad \mathbb{E}(\mathbf{a}_\ell \mathbf{a}_\ell^*) = \mathbf{I}_N, \quad \forall k \in [d].$$

Lemma 3.3.1. *Let y be defined as in Proposition 3.3.1. Then*

$$\mathbf{y} = \mathcal{A}(\mathbf{h}_0\mathbf{x}_0^*) + \mathbf{e}. \tag{3.18}$$

This equivalent model to Proposition 3.3.1 will the model worked with for the rest of the chapter. We aim to recover $(\mathbf{h}_0, \mathbf{x}_0)$ by solving the minimization problem

$$\min_{(\mathbf{h},\mathbf{x})} F(\mathbf{h}, \mathbf{x}), \quad F(\mathbf{h}, \mathbf{x}) := \|\mathcal{A}(\mathbf{h}\mathbf{x}^*) - \mathbf{y}\|^2 = \|\mathcal{A}(\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*) - \mathbf{e}\|^2.$$

We also define

$$F_0(\mathbf{h},\mathbf{x}) := \|\mathcal{A}(\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*)\|^2, \quad \delta = \delta(\mathbf{h},\mathbf{x}) = \frac{\|\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*\|_F}{d_0}.$$

$F(\mathbf{h},\mathbf{x})$ is highly non-convex and thus attempts at minimization such as alternating minimization and gradient descent, can get easily trapped in some local minima.

3.3.2.1 Main Theorems

Theorem 3.3.1. ***(Existence Of Unique Solution)*** *([2], Theorem 1) Fix $\alpha \geq 1$. Then there exists a constant $C_\alpha = O(\alpha)$, such that if*

$$\max(K \cdot \mu_{\max}^2, N \cdot \mu_h^2) \leq \frac{d}{C_\alpha \log^3 d},$$

then $\mathbf{X}_0 = \mathbf{h}_0\mathbf{x}_0^*$ is the unique solution to our minimization problem with probability $1 - O(d^{-\alpha+1})$, thus we can separate $\mathbf{y} = \mathbf{f} * \mathbf{g}$ up to a scalar multiple. When coherence is low, this is tight within a logarithmic factor, as we always have $\max(K,N) \leq d$.

Theorem 3.3.2. ***(Stability From Noise)*** *([2], Theorem 2) Let $\mathbf{X}_0 = \mathbf{h}_0\mathbf{m}_0^*$ and suppose the condition of previous theorem holds. We observe that*

$$\mathbf{y} = \mathcal{A}(\mathbf{X}_0) + \mathbf{e},$$

where $\mathbf{e} \in \mathbb{R}^d$ is an unknown noise vector with $\|z\|_2 \leq \delta$, and estimate $\mathbf{X}_0$ by solving

$$\min \quad \|\mathbf{X}\|_*, \qquad \textit{subject to} \quad \|\widehat{y} - \mathcal{A}(\mathbf{X})\|_2 \leq \delta.$$

Let $\lambda_{\min}, \lambda_{\max}$ be the smallest/largest non-zero eigenvalue of $\mathcal{A}\mathcal{A}^*$. Then with probability $1 - d^{-\alpha+1}$, the solution $\mathbf{X}$ will obey

$$\|\mathbf{X} - \mathbf{X}_0\|_F \leq C\frac{\lambda_{\max}}{\lambda_{\min}}\sqrt{\min(K,N)}\delta,$$

for a fixed constant C.

3.3.3 Wirtinger Gradient Descent

In [70], the approach is to solve the minimization problem (Equation 3.19) using Wirtinger gradient descent. In this subsection, the algorithm is introduced as well as the main theorems which establish convergence of the proposed algorithm to the true solution.

The algorithm consists of two parts: first an initial guess, and secondly, a variation of gradient descent, starting at the initial guess to converge to the true solution. Theoretical results are established for avoiding getting stuck in local minima.

This is ensured by determining that the iterates are inside some properly chosen basin of attraction of the true solution.

3.3.4 Basin of Attraction

Proposition 3.3.2. ***Basin of Attraction****: (Section 3.1, [70]) Three neighbourhoods are introduced whose intersection will form the basis of attraction of the solution:*
(i) ***Non-uniqueness****: Due to the scale ambiguity, for numerical stability we introduce the following neighbourhood*

$$N_{L_0} := \{(\mathbf{h}, \mathbf{x}) \mid \|\mathbf{h}\| \leq 2\sqrt{L_0}, \|\mathbf{x}\| \leq 2\sqrt{L_0}\}, \quad L_0 = \|\mathbf{h}_0\| \cdot \|\mathbf{x}_0\|.$$

(ii) ***Incoherence****: The number of measurements required for solving the blind deconvolution problem depends on how much* $\mathbf{h}_0$ *is correlated with the rows of the matrix* $\mathbf{B}$*, with the hopes of minimizing the correlation. We define the incoherence between the rows of* $\mathbf{B}$ *and* $\mathbf{h}_0$*, via*

$$\mu_{\mathbf{h}}^2 = \frac{d\|\mathbf{B}\mathbf{h}_0\|_\infty^2}{\|\mathbf{h}_0\|^2}.$$

To ensure that the incoherence of the solution is under control, we introduce the neighborhood

$$N_\mu := \{\mathbf{h} \mid \sqrt{d}\|\mathbf{B}\mathbf{h}\|_\infty \leq 4\sqrt{L_0}\mu\}, \quad \mu_{\mathbf{h}} \leq \mu. \tag{3.19}$$

(iii) ***Initial guess****: A carefully chosen initial guess is required due to the non-convexity of the function we wish to minimize. The distance to the true solution is defined via the following neighborhood*

$$N_\epsilon := \{(\mathbf{h}, \mathbf{x}) \mid \|\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*\|_F \leq \epsilon L_0\}, \quad 0 < \epsilon \leq \frac{1}{15}. \tag{3.20}$$

Thus the basin of attraction is chosen as $N_{d_0} \cap N_\mu \cap N_\epsilon$, where the true solution lies.

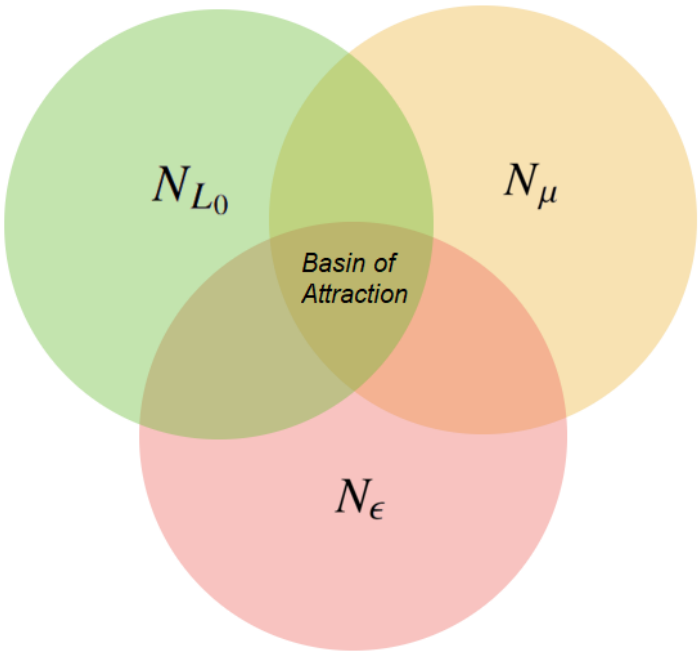

Figure 3.3 Basin of Attraction: $N_{L_0} \cap N_\mu \cap N_\epsilon$

Our approach consists of two parts: We first construct an initial guess that is inside the basin of attraction $N_{L_0} \cap N_\mu \cap N_\epsilon$. We then apply a regularized Wirtinger gradient descent algorithm that will ensure that all the iterates remain inside $N_{L_0} \cap N_\mu \cap N_\epsilon$. To achieve that, we add a regularizing function $G(\mathbf{h}, \mathbf{x})$ to the objective function $F(\mathbf{h}, \mathbf{x})$ to enforce that the iterates remain inside $N_{L_0} \cap N_\mu$. Hence we aim the minimize the following regularized objective function, in order to solve the blind deconvolution problem:

$$\tilde{F}(\mathbf{h}, \mathbf{x}) := F(\mathbf{h}, \mathbf{x}) + G(\mathbf{h}, \mathbf{x}),$$

where $F(\mathbf{h},\mathbf{x}) := \|\mathcal{A}(\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*) - \mathbf{e}\|^2$ is defined as before and $G(\mathbf{h},\mathbf{x})$ is the penalty function, of the form

$$G(\mathbf{h},\mathbf{x}) := \rho\Big[G_0\Big(\frac{\|\mathbf{h}\|^2}{2L}\Big) + G_0\Big(\frac{\|\mathbf{x}\|^2}{2L}\Big) + \sum_{\ell=1}^{d} G_0\Big(\frac{d|\mathbf{b}_\ell^*\mathbf{h}|^2}{8L\mu^2}\Big)\Big],$$

where

$$G_0(z) := \max\{z-1,0\}^2, \quad \rho \geq L^2 + 2\|\mathbf{e}\|^2.$$

It is assumed $\frac{9}{10}L_0 \leq L \leq \frac{11}{10}L_0$ and $\mu \geq \mu_h$.

Remark 3.3.1. *The matrix $\mathcal{A}^*(\mathbf{e}) = \sum_{k=1}^{d} \mathbf{e}_k\mathbf{b}_k\mathbf{a}_k^*$ as a sum of d rank-1 random matrices, has nice concentration of measure properties. Asymptotically, $\|\mathcal{A}^*(e)\|$ converges to 0 with rate $O(d^{1/2})$. Note that*

$$F(\mathbf{h},\mathbf{x}) = \|\mathbf{e}\|^2 + \|\mathcal{A}(\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*)\|_F^2 - 2Re(\langle\mathcal{A}^*(\mathbf{e}), \mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*\rangle).$$

If one lets $d \longrightarrow \infty$, then $\|\mathbf{e}\|^2 \sim \frac{\sigma^2 d_0^2}{2d}\chi_{2d}^2$ will converge almost surely to $\sigma^2 d_0^2$ [2] and the cross term $Re(\langle\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*, \mathcal{A}^*(\mathbf{e})\rangle)$ will converge to 0. In other words, asymptotically

$$\lim_{d\to\infty} F(\mathbf{h},\mathbf{x}) = F_0(\mathbf{h},\mathbf{x}) + \sigma^2 L_0^2,$$

for all fixed $(\mathbf{h},\mathbf{x})$. This implies that if the number of measurements is large, then $F(\mathbf{h},\mathbf{x})$ behaves "almost like" $F_0(\mathbf{h},\mathbf{x}) = \|\mathcal{A}(\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*)\|^2$, the noiseless version of $F(\mathbf{h},\mathbf{x})$. So for large d, we effectively ignore the noise.

Theorem 3.3.3. *For any given $Z \in \mathbb{C}^{K\times N}$, we have that*

$$\mathbb{E}(\mathcal{A}^*(\mathcal{A}(\mathbf{Z}))) = \mathbf{Z}.$$

[2] *This is due to the law of large numbers*

Proof. By linearity, sufficient to prove for $Z \in \mathbb{C}^{K\times N}$ where $Z_{i,j} = 1$, 0 otherwise. Then we have that

$$\mathbb{E}((\mathcal{A}^*\mathcal{A}(\mathbf{Z}))) = \mathbb{E}(\sum_{k=1}^{L} b_{ki}^* a_{kj} \mathbf{b}_k \mathbf{a}_k^*)) = \sum_{k=1}^{d} (b_{ki}^* \mathbf{b}_k \mathbb{E}(a_{kj}\mathbf{a}_k^*)) = \sum_{k=1}^{d} (b_{ki}^* \mathbf{b}_k) z_j^* = \mathbf{e}_i \mathbf{e}_j^* = Z,$$

Thus we have that

$$\mathbb{E}(\mathcal{A}^*(\mathbf{y})) = \mathbb{E}(\mathcal{A}^*(\mathcal{A}(\mathbf{h}_0\mathbf{x}_0^*) + \mathbf{e}) = \mathbb{E}(\mathcal{A}^*(\mathcal{A}(\mathbf{h}_0\mathbf{x}_0^*))) + \mathbb{E}(\mathcal{A}^*(\mathbf{e})) = \mathbf{h}_0\mathbf{x}_0^*,$$

since $\mathbb{E}(\mathcal{A}^*(\mathbf{e}))$ by the definition of $\mathbf{e}$. □

Hence it makes logical sense that the leading singular value and vectors of $\mathcal{A}^*(\mathbf{y})$ would be a good approximation of L_0 and $(\mathbf{h}_0, \mathbf{x}_0)$ respectively.

3.3.5 Algorithms

We can now state the algorithm for generating an initial estimate.

Algorithm 3.2 Blind Deconvolution Initial Estimate

Input: Blind Deconvolutional measurements $\mathbf{y}$, $K = \text{supp}(f)$.

Output: Estimate underlying signal and blurring function.

1) Compute $\mathcal{A}^*(\mathbf{y})$ and find the leading singular value, left and right singular vectors of $\mathcal{A}^*(\mathbf{y})$, denoted by d, $\tilde{\mathbf{h}}_0$, and $\tilde{\mathbf{x}}_0$ respectively.

2) Solve the following optimization problem

$$\mathbf{u_0} := \underset{z}{argmin} \|\mathbf{z} - \sqrt{d}\tilde{\mathbf{h}}_0\|^2, \quad \text{subject to } \sqrt{d}\|B\mathbf{z}\|_\infty \leq 2\sqrt{L}\mu,$$

and $\mathbf{x_0} = \sqrt{L}\tilde{\mathbf{x}}_0$.

Since we are dealing with complex variables, for the gradient descent, Wirtinger derivatives are utilized. Since $\tilde{F}$ is a real-valued function, we only need to consider the derivative of $\tilde{F}$, with respect to $\bar{\mathbf{h}}$ and $\bar{\mathbf{x}}$, and the corresponding updates of $\mathbf{h}$ and $\mathbf{x}$ since

$$\frac{\partial \tilde{F}}{\partial \bar{\mathbf{h}}} = \overline{\frac{\partial \tilde{F}}{\partial \mathbf{h}}}, \quad \frac{\partial \tilde{F}}{\partial \bar{\mathbf{x}}} = \overline{\frac{\partial \tilde{F}}{\partial \mathbf{x}}}.$$

In particular, we denote

$$\nabla \tilde{F}_{\mathbf{h}} := \frac{\partial \tilde{F}}{\partial \bar{\mathbf{h}}}, \quad \nabla \tilde{F}_{\mathbf{x}} := \frac{\partial \tilde{F}}{\partial \bar{\mathbf{x}}}. \tag{3.21}$$

We can now state the full algorithm.

Algorithm 3.3 Wirtinger Gradient Descent Blind Deconvolution Algorithm

Input: Blind Deconvolutional measurements $\mathbf{y}$, $K = \text{supp}(f)$.

Output: Estimate underlying signal and blurring function.

1) Compute $\mathcal{A}^*(\mathbf{y})$ and find the leading singular value, left and right singular vectors of $\mathcal{A}^*(\mathbf{y})$, denoted by d, $\tilde{\mathbf{h}}_0$, and $\tilde{\mathbf{x}}_0$ respectively.

2) Solve the following optimization problem

$$\mathbf{u_0} := \underset{z}{argmin} \|\mathbf{z} - \sqrt{L}\tilde{\mathbf{h}}_0\|^2, \quad \text{subject to } \sqrt{d}\|B\mathbf{z}\|_\infty \leq 2\sqrt{L}\mu,$$

and $\mathbf{x_0} = \sqrt{L}\tilde{\mathbf{x}}_0$.

3) Compute Wirtinger Gradient Descent

while halting criterion false **do**

$\mathbf{u_t} = \mathbf{u_{t-1}} - \eta \nabla \tilde{F}_{\mathbf{h}}(\mathbf{u_{t-1}}, \mathbf{u_{t-1}})$

$\mathbf{v_t} = \mathbf{v_{t-1}} - \eta \nabla \tilde{F}_{\mathbf{x}}(\mathbf{v_{t-1}}, \mathbf{v_{t-1}})$

end while

4) Set $(\mathbf{h}, \mathbf{x}) = (\mathbf{u_t}, \mathbf{v_t})$

In [70], the authors show that with a carefully chosen initial guess $(\mathbf{u}_0, \mathbf{v}_0)$, running Wirtinger gradient descent to minimize $\tilde{\mathbf{F}}(h, x)$ will guarantee linear convergence of the sequence $(\mathbf{u}_t, \mathbf{v}_t)$ to the global minimum $(\mathbf{h}_0, \mathbf{x}_0)$ in the noiseless case, and also provide robust recovery in the presence of noise. The results are summarized in the following two theorems.

3.3.6 Main Theorems

Theorem 3.3.4. ***(Main Theorem 1)*** *([70], Theorem 3.1) The initialization obtained via Algorithm 3.2 satisfies*

$$(\mathbf{u}_0, \mathbf{v}_0) \in \frac{1}{\sqrt{3}} N_{L_0} \cap \frac{1}{\sqrt{3}} N_\mu \cap N_{\frac{2}{5}\epsilon}, \quad \frac{9}{10} L_0 \leq L \leq \frac{11}{10} L_0,$$

with probability at least $1 - d^{-\gamma}$ *if the number of measurements is sufficient large, that is*

$$d \geq C_\gamma(\mu_h^2 + \sigma^2)\max\{K, N\}\log^2 d/\epsilon^2,$$

where ϵ *is a predetermined constant on* $\left(0, \frac{1}{15}\right]$*, and* C_γ *is a constant only linearly depending on* γ *with* $\gamma \geq 1$.

The following theorem establishes that as long as the initial guess lies inside the basin of attraction of the true solution, regularized gradient descent will converge to this solution (or to a nearby solution in case of noisy data).

Theorem 3.3.5. ***(Main Theorem 2)*** *([70], Theorem 3.2) Assume that the initialization* $(\mathbf{u}_0, \mathbf{v}_0) \in \frac{1}{\sqrt{3}}N_{L_0} \cap \frac{1}{\sqrt{3}}N_\mu \cap N_{\frac{2}{5}\epsilon}$*, and that* $d \geq C_\gamma(\mu^2 + \sigma^2)\max\{K, N\}\log^2(L)/\epsilon^2$*. Then Algorithm 3.3 will create a sequence* $(\mathbf{u}_t, \mathbf{v}_t) \in N_{d_0} \cap N_\mu \cap N_\epsilon$ *which converges geometrically to* $(\mathbf{h}_0, \mathbf{x}_0)$ *in the sense that with probability at least* $1 - 4d^{-\gamma} - \frac{1}{\gamma}e^{-(K+N)}$*, we have that*

$$\max\{sin\angle(\mathbf{u}_t, \mathbf{h}_0), sin\angle(\mathbf{v}_t, \mathbf{x}_0)\} \leq \frac{1}{L_t}\left(\frac{2}{3}(1 - \eta\omega)^{t/2}\eta L_0 + 50\|\mathcal{A}^*(\mathbf{e})\|\right),$$

and

$$|L_t - L_0| \leq \frac{2}{3}(1 - \eta\omega)^{t/2}\epsilon L_0 + 50\|\mathcal{A}^*(\mathbf{e})\|,$$

where $L_t := \|\mathbf{u}_t\| \cdot \|\mathbf{v}_t\|, \omega > 0, \eta$ *is the fixed stepsize. Here*

$$\|\mathcal{A}^*(\mathbf{e})\| \leq C_0\sigma d_0 \max\left\{\sqrt{\frac{(\gamma+1)\max\{K, N\}\log d}{d}}, \frac{(\gamma + d)\sqrt{KN}\log^2 d}{d}\right\},$$

holds with probability $1 - d^{-\gamma}$.

It has been shown with high probability that as long as the initial guess lies inside the basin of attraction of the true solution, Wirtinger gradient descent will converge towards the solution.

3.3.7 Key Conditions

Theorem 3.3.6. ***Four Key Conditions:***

(i) ***(Local RIP Condition)*** *([70], Condition 5.1) The following local Restricted Isometry Property (RIP) for A holds uniformly for all* $(\mathbf{h}, \mathbf{x})$ *in the basin of attraction* $(N_{L_0} \cap N_\mu \cap N_\epsilon)$

$$\frac{3}{4}\|\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*\|_F^2 \leq \|\mathcal{A}(\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*)\|^2 \leq \frac{5}{4}\|\mathbf{h}\mathbf{x}^* - \mathbf{h}_0\mathbf{x}_0^*\|_F^2.$$

(ii) ***(Robustness Condition)*** *([70], Condition 5.2) For the complex Gaussian noise* $\mathbf{e}$*, with high probability*

$$\|\mathcal{A}^*(\mathbf{e})\| \leq \frac{\epsilon L_0}{10\sqrt{2}},$$

for d sufficiently large, that is, $d \geq C_\gamma(\frac{\sigma^2}{\epsilon^2} + \frac{\sigma}{\epsilon})\max\{K, N\} log\ d$*;*

(iii) ***(Local Regularity Condition)*** *([70], Condition 5.3) There exists a regularity constant* $\omega = \frac{d_0}{5000} > 0$ *such that*

$$\|\nabla \tilde{F}(\mathbf{h}, \mathbf{x})\|^2 \geq \omega[\tilde{F}(\mathbf{h}, \mathbf{x}) - c]_+, \quad c = \|\mathbf{e}\|^2 + 1700\|\mathcal{A}^*(\mathbf{e})\|^2,$$

for all $(\mathbf{h}, \mathbf{x}) \in N_{L_0} \cap N_\mu \cap N_\epsilon$*;*

(iv) ***(Local Smoothness Condition)*** *([70], Condition 5.4) Denote* $\mathbf{z} := (\mathbf{h}, \mathbf{x})$*. There exists a constant* C_d *such that*

$$\|\nabla f(\mathbf{z} + t\Delta\mathbf{z}) - \nabla f(\mathbf{z})\| \leq C_d t\|\Delta\mathbf{z}\|, \quad 0 \leq t \leq 1,$$

for all $\{(\mathbf{z}, \Delta\mathbf{z}) \mid \mathbf{z} + t\Delta\mathbf{z} \in N_\epsilon \cap N_{\tilde{F}}, \forall 0 \leq t \leq 1\}$*, i.e., the whole segment connecting* $(\mathbf{h}, \mathbf{x})$ *and* $\nabla(\mathbf{h}, \mathbf{x})$ *belongs to the non-convex set* $N_\epsilon \cap N_{\tilde{F}}$*.*

3.4 Blind Ptychography

3.4.1 Introduction

A more recent area of study is blind ptychography, in which both the object and the mask are considered unknown, up to reasonable assumptions. The first successful recovery was given in ([110],[109]) , further study into the sufficient overlap ([10], [76], [75]), and summarized in [30].

Let $\mathbf{x}, \mathbf{m} \in \mathbb{C}^d$ denote the unknown sample and mask, respectively. We suppose that we have d^2 noisy ptychographic measurements of the form

$$(\mathbf{Y})_{\ell,k} = |(\mathbf{F}(\mathbf{x} \circ S_k\mathbf{m}))_\ell|^2 + (\mathbf{N})_{\ell,k}, \quad (\ell, k) \in [d]_0 \times [d]_0, \tag{3.22}$$

where $S_k, \circ, \mathbf{F}$ denote k^{th} circular shift, Hadamard product, and d-dimensional discrete Fourier transform, and $\mathbf{N}$ is the matrix of additive noise. By Theorem 3.2.2, we have shown we can rewrite the measurements as

$$\left(\mathbf{Y}^T\mathbf{F}^T\right)_k = d \cdot (\mathbf{x} \circ S_k\bar{\mathbf{x}}) * (\tilde{\mathbf{m}} \circ S_{-k}\bar{\tilde{\mathbf{m}}}) + \left(\mathbf{N}^T\mathbf{F}^T\right)_k, \tag{3.23}$$

where $*$ denotes the d-dimensional discrete convolution, and $\tilde{\mathbf{m}}$ denotes the reversal of $\mathbf{m}$ about its first entry. This is now a scaled blind deconvolution problem which has been studied in [2],[70].

3.4.2 Main Results

3.4.2.1 Recovering the Sample

To recover the sample, we will need to assume that $\mathbf{x}$ belongs to a known subspace. Initially we solve algorithmically for the zero shift case ($k = 0$) and then generalize the method to solve for the estimate which utilizes all the obtained shifts.

Our assumptions are as follows: $\mathbf{x} \in \mathbb{C}^d$ unknown, $\mathbf{x} = C\mathbf{x}'$, $C \in \mathbb{C}^{d\times N}$, $N \ll L$ known, $\mathbf{x}' \in \mathbb{C}^N$ or $\mathbb{R}^N$ unknown $\mathbf{m} \in \mathbb{C}^d$ unknown, $supp(\mathbf{m}) \subseteq [\delta]_0$, K known, $\|\mathbf{m}\|_2$ known. Known noisy measurements $\mathbf{Y}$.

Our first goal is to compute an estimate $\mathbf{x}_{est}$ of $\mathbf{x}$, true up to a global phase. We will use this

estimate to then produce an estimate $\mathbf{m}_{est}$ of $\mathbf{m}$, again true up to a global phase.

Firstly, we let $\mathbf{y}$ be the first column of $\frac{1}{\sqrt{d}} \cdot \mathbf{F}(\widetilde{\overline{(\mathbf{FY})^T}})$, $\mathbf{f} = \tilde{\mathbf{m}} \circ \overline{\tilde{\mathbf{m}}}$ (so $\|\mathbf{f}\|_2$ known). We next set $\mathbf{g} = \mathbf{x} \circ \bar{\mathbf{x}}$ but to fully utilize the blind deconvolution algorithm, we will need a lemma concerning hadamard products of products of matrices. Firstly, we need define some products between matrices.

Definition 3.4.1. *Let $\mathbf{A} \in \mathbb{C}^{m\times n}$ and $\mathbf{B} \in \mathbb{C}^{p\times q}$. Then the **Kronecker product** $A \otimes B \in \mathbb{C}^{mp\times nq}$ is defined by*

$$(A \otimes B)_{pr+v,qs+w} = a_{rs} b_{vw}. \tag{3.24}$$

Definition 3.4.2. *Let $\mathbf{A} \in \mathbb{C}^{m\times n}$ and $\mathbf{B} \in \mathbb{C}^{p\times n}$ with columns $\mathbf{a}_i, \mathbf{b}_i$ for $i \in [n]_0$. Then the **Khatri-Rao product** $A \odot B \in \mathbb{C}^{mp\times n}$ is defined by*

$$A \odot B = [\mathbf{a}_0 \otimes \mathbf{b}_0 \ \mathbf{a}_1 \otimes \mathbf{b}_1 \ldots \mathbf{a}_{n-1} \otimes \mathbf{b}_{n-1}]. \tag{3.25}$$

Definition 3.4.3. *Let $\mathbf{A} \in \mathbb{C}^{m\times n}$ and $\mathbf{B} \in \mathbb{C}^{m\times p}$ be matrices with rows $\mathbf{A}_i, \mathbf{B}_i$ for $i \in [m]$. Then the **transposed Khatri-Rao product** (or **face-splitting product**), denoted $\bullet$, is the matrix whose rows are Kronecker products of the columns of $\mathbf{A}$ and $\mathbf{B}$ i.e. the rows of $\mathbf{A} \cdot \mathbf{B} \in \mathbb{C}^{m\times np}$ are given by*

$$(\mathbf{A} \cdot \mathbf{B})_i = A_i \otimes B_i, \quad i \in [m] \tag{3.26}$$

We then utilize the following lemma concerning the transposed Khatri-Rao product.

Lemma 3.4.1 (Theorem 1, [100])**.** *Let $\mathbf{A} \in \mathbb{C}^{m\times n}, \mathbf{B} \in \mathbb{C}^{n\times p}, \mathbf{C} \in \mathbb{C}^{m\times q}, \mathbf{D} \in \mathbb{C}^{q\times p}$. Then we have that*

$$(\mathbf{AB}) \circ (\mathbf{CD}) = (\mathbf{A} \bullet \mathbf{C})(\mathbf{B} \odot \mathbf{D}),$$

where $\circ$ is the Hadamard product, $\odot$ is the standard Khatri-Rao product, and $\bullet$ is the transposed

Khatri-Rao product.

Thus by Lemma 3.4.1 we have that for $\mathbf{g} = \mathbf{x} \circ \bar{\mathbf{x}} = \mathbf{C}\mathbf{x}' \circ \bar{\mathbf{C}}\bar{\mathbf{x}}'$. Then $g = \mathbf{C}'\mathbf{x}''$ where $\mathbf{C}' \in \mathbb{C}^{L\times N^2}, \mathbf{x}'' \in \mathbb{C}^{N^2}$ are given by

$$\mathbf{C}' = \mathbf{C} \bullet \bar{\mathbf{C}}, \quad \mathbf{x}'' = \mathbf{x}' \odot \bar{\mathbf{x}}'.$$

We now compute RRR Blind Deconvolution (Algorithm 3.3) with $\mathbf{y}, \mathbf{f}, \mathbf{g}, \mathbf{C}, K = \delta$ as above (**B** last K columns of DFT matrix) to obtain estimate for $\mathbf{x}' \odot \bar{\mathbf{x}}'$. Use angular synchronisation to solve for $\mathbf{x}'$, and thus solve for $\mathbf{x}$.

Algorithm 3.4 Blind Ptychography (Zero Shift)

Input:

1) $\mathbf{x} \in \mathbb{C}^d$ unknown, $\mathbf{x} = C\mathbf{x}', C \in \mathbb{C}^{d\times N}, N \ll L$ known, $\mathbf{x}' \in \mathbb{C}^N$ or $\mathbb{R}^N$ unknown.
2) $\mathbf{m} \in \mathbb{C}^d$ unknown, $supp(\mathbf{m}) \subseteq [\delta]_0$, K known, $\|\mathbf{m}\|_2$ known Known noisy measurements $\mathbf{Y}$.
3) Known noisy measurements $\mathbf{Y}$.

Output: Estimate $\mathbf{x}_{est}$ of $\mathbf{x}$ true up to a global phase.

1) Let $\mathbf{y}$ be the first column of $\frac{1}{\sqrt{d}} \cdot \mathbf{F}(\widetilde{(\mathbf{F}\mathbf{Y})^T})$, $\mathbf{f} = \tilde{\mathbf{m}} \circ \overline{\tilde{\mathbf{m}}}$ (so $\|\mathbf{f}\|_2$ known)
2) Let $\mathbf{g} = \mathbf{x} \circ \bar{\mathbf{x}} = \mathbf{C}\mathbf{x}' \circ \bar{\mathbf{C}}\bar{\mathbf{x}}'$. Then $g = \mathbf{C}'\mathbf{x}''$ where $\mathbf{C}' \in \mathbb{C}^{d\times N^2}, \mathbf{x}'' \in \mathbb{C}^{N^2}$ are given by

$$\mathbf{C}' = \mathbf{C} \bullet \bar{\mathbf{C}}, \quad \mathbf{x}'' = \mathbf{x}' \odot \bar{\mathbf{x}}'.$$

3) Compute RRR Blind Deconvolution (Algorithm 1 & 2, [70]) with $\mathbf{y}, \mathbf{f}, \mathbf{g}, \mathbf{C}, K = \delta$ as above (**B** last K columns of DFT matrix) to obtain estimate for $\mathbf{x}' \odot \bar{\mathbf{x}}'$.
4) Use angular synchronisation to solve for $\mathbf{x}'$, and thus compute $\mathbf{x}_{est}$.

3.4.2.2 Recovering the Mask

Once the estimate of $\mathbf{x}$ has been found, denoted $\mathbf{x}_{est}$, we use this estimate to find $\mathbf{m}_{est}$. We first compute $\mathbf{g}_{est} = \mathbf{x}_{est} \circ \overline{\mathbf{x}}_{est}$, and then we use point-wise division to find

$$\mathbf{F}(\tilde{\mathbf{m}} \circ \bar{\tilde{\mathbf{m}}}) = \frac{\overline{\mathbf{F}^{-1}(\widetilde{(\mathbf{F}\mathbf{Y})^T})}}{\mathbf{F}(\mathbf{x}_{est} \circ \overline{\mathbf{x}}_{est})}. \tag{3.27}$$

Then use an inverse Fourier transform, a reversal and then angular synchronization, similar to obtaining $\mathbf{x}_{est}$.

Algorithm 3.5 Recovering The Mask

Input: 1) $\mathbf{x}_{est}$ generated by Algorithm 3.4.
2) Known noisy measurements $\mathbf{Y}$.
3) $supp(\mathbf{m}) \subseteq [\delta]_0$, K known, $\|\mathbf{m}\|_2$ known.

Output: Estimate $\mathbf{m}_{est}$ of $\mathbf{m}$ true up to a global phase.

1) Compute $\mathbf{g}_{est} = \mathbf{x}_{est} \circ \bar{\mathbf{x}}_{est}$ and $2\delta - 1$ perform point-wise divisions to obtain

$$\mathbf{F}(\tilde{\mathbf{m}}_{est} \circ S_{-k}\bar{\tilde{\mathbf{m}}}_{est}) = \frac{\overline{\mathbf{F}^{-1}(\widetilde{(\mathbf{FY})^T})_k}}{\mathbf{F}(\mathbf{x}_{est} \circ S_k\bar{\mathbf{x}}_{est})}. \tag{3.28}$$

2) Compute inverse Fourier transform to obtain $\tilde{\mathbf{m}}_{est} \circ S_{-k}\bar{\tilde{\mathbf{m}}}_{est}$ and use these to form the diagonals of a banded matrix.

3) Use angular synchronisation to solve for $\tilde{\mathbf{m}}_{est}$, and thus perform a reversal to compute $\mathbf{m}_{est}$.

3.4.2.3 Multiple Shifts

To generalize the setup, we let $\mathbf{y}_{(k)}$ denote the k^{th} column of $\frac{1}{\sqrt{d}} \cdot \mathbf{F}(\overline{\widetilde{(\mathbf{FY})^T}})$, $\mathbf{f}_{(k)} = \tilde{\mathbf{m}} \circ S_{-k}\overline{\tilde{\mathbf{m}}}$. Let $\mathbf{g}_{(k)} = \mathbf{x} \circ S_k\bar{\mathbf{x}} = \mathbf{C}\mathbf{x}' \circ S_k\bar{\mathbf{C}}\bar{\mathbf{x}}'$. Then again by another application of Lemma 3.4.1, $\mathbf{g}_{(k)} = \mathbf{C}'_{(k)}\mathbf{x}''$ where $\mathbf{C}' \in \mathbb{C}^{d\times N^2}, \mathbf{x}'' \in \mathbb{C}^{N^2}$ are given by

$$\mathbf{C}'_{(k)} = \mathbf{C} \bullet S_k\bar{\mathbf{C}}, \quad 0 \le k \le K, d - K + 1 \le k \le d, \quad \mathbf{x}'' = \mathbf{x}' \odot \bar{\mathbf{x}}' = vec(\mathbf{x}'(\mathbf{x}')*).$$

We then perform $2\delta - 1$ blind deconvolutions to obtain $2\delta - 1$ estimates of $\mathbf{x}$ and $\mathbf{m}$, labelled $\mathbf{x}^i_{est}, \mathbf{m}^j_{est}$ respectively for $i, j \in [2\delta - 1]_0$.

Ideally, we would want to select the estimates which generates the minimum error for each $\mathbf{x}$ and $\mathbf{m}$ but that implies prior knowledge of $\mathbf{x}$ and $\mathbf{m}$. Instead, compute $(2\delta - 1)^2$ estimates of the Fourier measurements by

$$(\mathbf{Y}^{i,j}_{est})_{\ell,k} = |(\mathbf{F}(\mathbf{x}^i_{est} \circ S_k\mathbf{m}^j_{est}))_\ell|^2, \quad i, j \in [2\delta - 1]_0. \tag{3.29}$$

We then compute the associated error

$$(i', j') = \underset{(i,j)}{\operatorname{argmin}} \frac{\|\mathbf{Y}^{i,j}_{est} - \mathbf{Y}\|_F^2}{\|\mathbf{Y}\|_F^2}, \quad i, j \in [2\delta - 1]_0. \tag{3.30}$$

Then let $\mathbf{x}_{est} = \mathbf{x}^{i'}_{est}, \mathbf{m}_{est} = \mathbf{m}^{j'}_{est}$.

Algorithm 3.6 Blind Ptychography (Multiple Shifts)

Input:

1) $\mathbf{x} \in \mathbb{C}^d$ unknown, $\mathbf{x} = C\mathbf{x}', C \in \mathbb{C}^{d\times N}, N \ll d$ known, $\mathbf{x}' \in \mathbb{C}^N$ or $\mathbb{R}^N$ unknown.

2) $\mathbf{m} \in \mathbb{C}^d$ unknown, $supp(\mathbf{m}) \subseteq [\delta]_0$, K known, $\|\mathbf{m}\|_2$.

3) Known noisy measurements $\mathbf{Y}$.

Output: Estimate $\mathbf{x}_{est}$ of $\mathbf{x}$, true up to a global phase

1) Let $\mathbf{y}_{(k)}$ denote the k^{th} column of $\frac{1}{\sqrt{d}} \cdot \mathbf{F}(\overline{(\widetilde{\mathbf{FY}})^T})$, $\mathbf{f}_{(k)} = \tilde{\mathbf{m}} \circ S_{-k}\overline{\tilde{\mathbf{m}}}$ (so $\|\mathbf{f}_{(k)}\|_2$ known).

2) Let $\mathbf{g}_{(k)} = \mathbf{x} \circ S_k\bar{\mathbf{x}} = \mathbf{C}\mathbf{x}' \circ S_k\bar{\mathbf{C}}\bar{\mathbf{x}}'$. Then $g_{(k)} = \mathbf{C}'\mathbf{x}''$ where $\mathbf{C}' \in \mathbb{C}^{d\times N^2}, \mathbf{x}'' \in \mathbb{C}^{N^2}$ are given by

$$\mathbf{C}'_{(k)} = C \bullet S_k\bar{C}, \quad 0 \le k \le K, d - K + 1 \le K \le d, \quad \mathbf{x}'' = \mathbf{x}' \odot \bar{\mathbf{x}}'.$$

3) Perform $2\delta - 1$ RRR Blind Deconvolutions (Algorithm 1 & 2, [70]) with $\mathbf{y}_{(k)}, \mathbf{f}_{(k)}, \mathbf{g}_{(k)}, \mathbf{C}$, as above to obtain $2\delta - 1$ estimates for $\mathbf{x}' \odot \bar{\mathbf{x}}'$.

4) Use angular synchronisation to solve for $2\delta - 1$ estimates $\mathbf{x}'_{est}$, and thus solve for $2\delta - 1$ estimates $\mathbf{x}^i_{est} = \mathbf{C}\mathbf{x}'_{est}, i \in [2\delta - 1]_0$.

5) Use these estimates $\mathbf{x}^i_{est}$ to compute $2\delta - 1$ estimates $\mathbf{m}^j_{est}, j \in [2\delta - 1]_0$.

6) Compute $(2\delta - 1)^2$ estimates of the Fourier measurements by

$$(\mathbf{Y}^{i,j}_{est})_{\ell,k} = |(\mathbf{F}(\mathbf{x}^i_{est} \circ S_k\mathbf{m}^j_{est}))_\ell|^2, \quad i, j \in [2\delta - 1]_0. \tag{3.31}$$

We then compute the associated error $(i', j') = \underset{(i,j)}{\text{argmin}} \dfrac{\|\mathbf{Y}^{i,j}_{est} - \mathbf{Y}\|^2_F}{\|\mathbf{Y}\|^2_F}, \quad i, j \in [2\delta - 1]_0.$

7) Let $\mathbf{x}_{est} = \mathbf{x}^{i'}_{est}, \mathbf{m}_{est} = \mathbf{m}^{j'}_{est}$.

3.4.3 Numerical Simulations

All simulations were performed using MATLAB R2021b on an Intel desktop with a 2.60GHz i7-10750H CPU and 16GB DDR4 2933MHz memory. All code used to generate the figures below is publicly available at https://github.com/MarkPhilipRoach/BlindPtychography.

To be more precise, we have defined the immeasurable (in practice since $\mathbf{x}$ and $\mathbf{m}$ are both unknown) estimates

$$\text{Max Shift}^{(x)} = \underset{x^i_{est}}{\text{argmax}} \|\mathbf{x} - \mathbf{x}^i_{est}\|^2_2, \text{Max Shift}^{(m)} = \underset{m^j_{est}}{\text{argmax}} \|\mathbf{m} - \mathbf{m}^j_{est}\|^2_2, \quad i, j \in [2\delta - 1]_0.$$

$$\text{Min Shift}^{(x)} = \underset{x^i_{est}}{\text{argmin}} \|\mathbf{x} - \mathbf{x}^i_{est}\|^2_2, \text{Min Shift}^{(m)} = \underset{m^j_{est}}{\text{argmin}} \|\mathbf{m} - \mathbf{m}^j_{est}\|^2_2, \quad i, j \in [2\delta - 1]_0.$$

and the measurable estimates. First, No Shift$^{(x)}$ and No Shift$^{(m)}$ refer to the zero shift estimates outlined in Algorithm 3.4. Secondly, the estimates achieved in Algorithm 3.6

$$(\text{Argmin Shift}^{(x)}, \text{Argmin Shift}^{(m)}) = (\mathbf{x}^{i'}, \mathbf{m}^{j'}), \quad (i', j') = \underset{(i,j)}{\text{argmin}} \frac{\|\mathbf{Y}_{est}^{i,j} - \mathbf{Y}\|_F^2}{\|\mathbf{Y}\|_F^2}, \quad i, j \in [2\delta - 1]_0.$$

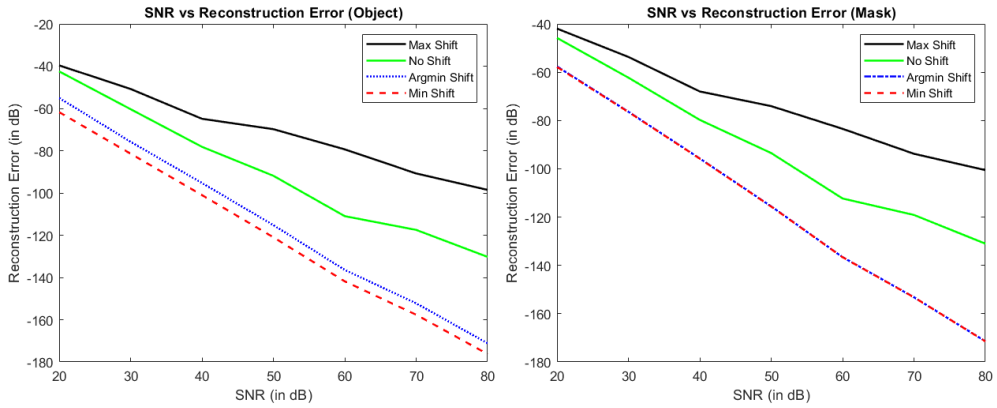

Figure 3.4 $d = 2^6, K = \delta = \log_2 d, N = 4, \mathbf{C}$ complex Gaussian. Max shift refers to the maximum error achieved from a blind deconvolution of a particular shift. Min shift refers to the maximum error achieved from a blind deconvolution of a particular shift. Argmin Shift refers to the choice of object and mask chosen in Step 6 of Algorithm 3.6. Averaged over 100 simulations. 1000 iterations.

Figure 3.4 demonstrates robust recovery under noise. It also demonstrates the impact of performing the $2\delta - 1$ blind deconvolutions and taking the Argmin Shift, versus simply taking the non-shifted object and mask. It also demonstrates how closely the reconstructions error from Argmin Shift and Min Shift are, in particular for the mask. Figure 3.5 demonstrates the impact even more, showing that with a higher value for the known subspace, the more accurate the Argmin Shift and Min Shift are, as well as demonstrating the large difference between the Max Shift and Min Shift.

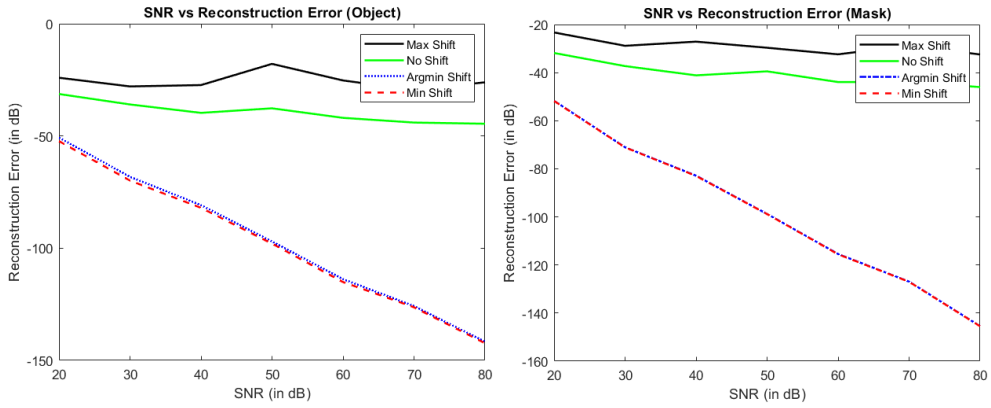

Figure 3.5 $d = 2^6, K = \delta = \log_2 d, N = 6, \mathbf{C}$ complex Gaussian. Max shift refers to the maximum error achieved from a blind deconvolution of a particular shift. Min shift refers to the maximum error achieved from a blind deconvolution of a particular shift. Argmin Shift refers to the choice of object and mask chosen in Step 6 of Algorithm 3.6. Averaged over 100 simulations. 1000 iterations.

The following figures demonstrate recovery against additional noise, with varying δ and N.

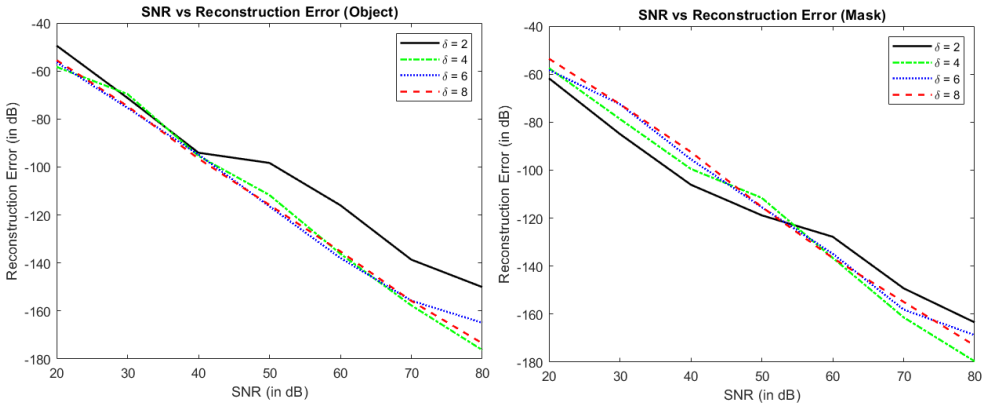

Figure 3.6 $d = 2^6, N = 4, \mathbf{C}$ complex Gaussian. Application of Algorithm 3.6 with varying $K = \delta$.

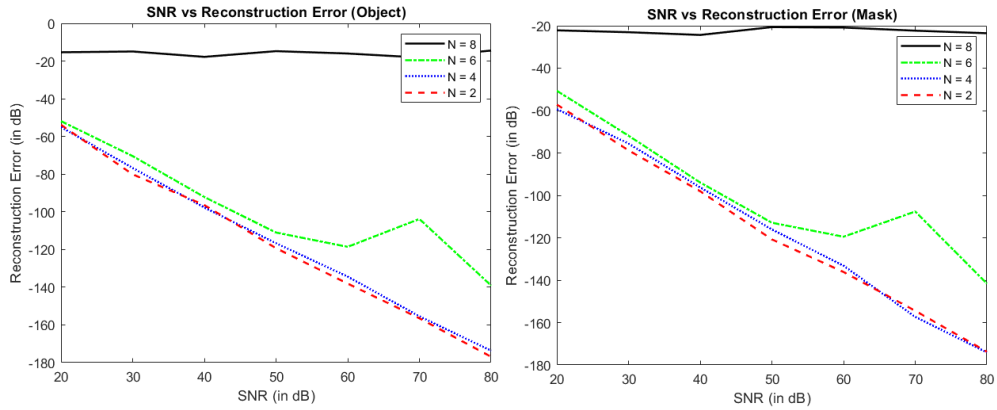

Figure 3.7 $d = 2^6, K = \delta = 6, \mathbf{C}$ complex Gaussian. Application of Algorithm 3.6 with varying N.

Next, we consider the frequency of the chosen index from performing the argmin function (step 6 of Algorithm 3.6) compared to the true minimizing indices for the object and mask separately.

Firstly, we have the frequency of the argmin indices.

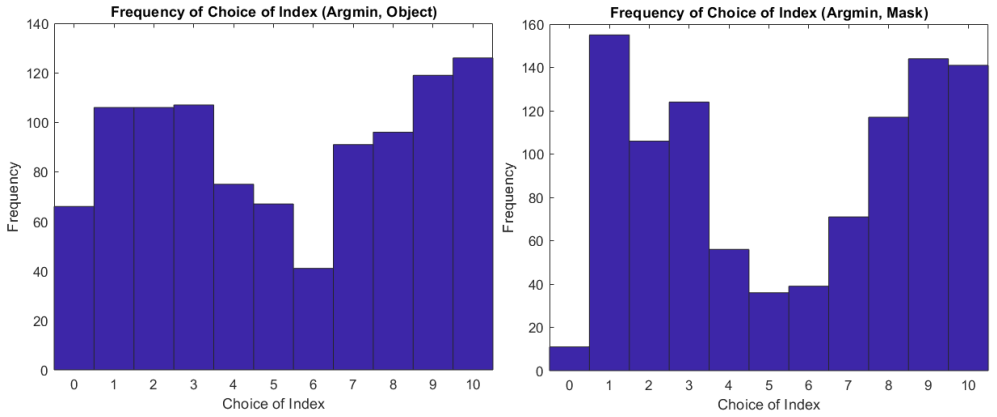

Figure 3.8 $d = 2^6, \delta = 6, N = 4, \mathbf{C}$ complex Gaussian. 1000 simulations. Frequency of index being chosen to compute Argmin Shift$^{(x)}$ and Argmin Shift$^{(m)}$.

Secondly, we have the frequency of the min shift for both of the object and mask. Both of Figure 3.8 and Figure 3.9 were computed on the same 1000 tests.

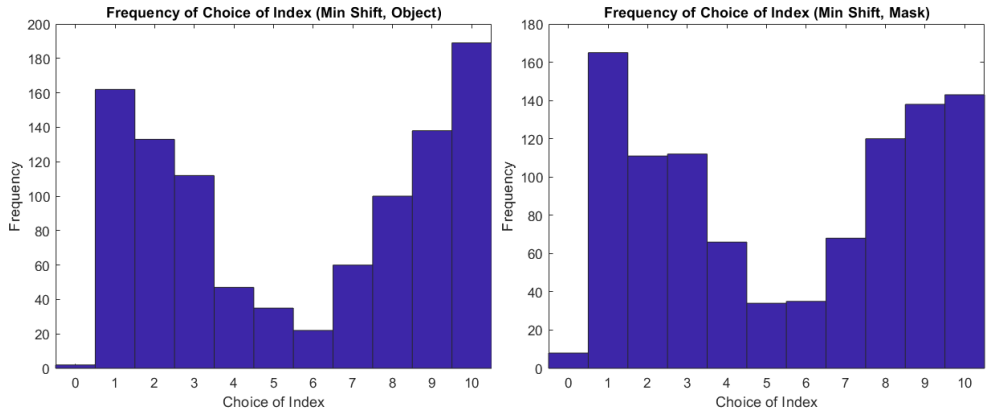

Figure 3.9 $d = 2^6, \delta = 6, N = 4, \mathbf{C}$ complex Gaussian. 1000 simulations. Frequency of index being chosen to compute Min Shift$^{(x)}$ and Min Shift$^{(m)}$.

Finally, we plot these choice of indices for both the Argmin Shift and Min Shift onto a two dimensional plot.

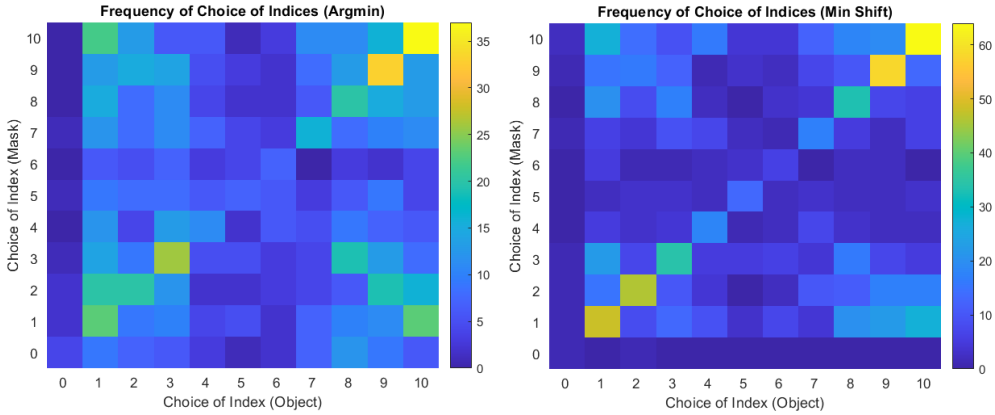

Figure 3.10 $d = 2^6, \delta = 6, N = 4, \mathbf{C}$ complex Gaussian. 1000 simulations. Frequency of indices being chosen to compute (Argmin Shift$^{(x)}$, Argmin Shift$^{(m)}$) and (Min Shift$^{(x)}$, Min Shift$^{(m)}$).

3.5 Conclusions and Future Work

We have introduced an algorithm for recovering a specimen of interest from blind far-field ptychographic measurements. This algorithm relies reformulating the measurements so that they resemble widely-studied blind deconvolutional measurements. This leads to Khatri-Rao product estimates of our specimen which are then able to be recovered by angular synchronization. We then use these estimates in applying inverse Fourier transforms, point-wise division, and angular synchronization

to recover estimates for the mask. Finally, we use a best error estimate sorting algorithm to find the final estimate of both the specimen and mask. As shown in numerical results, Algorithm 3.6 recovers both the sample and mask within a good margin of error. It also provides stability under noise. A further goal for this research would be to adapt the existing recovery guarantee theorems for the selected blind deconvolutional recovery algorithm, in which the assumed Gaussian matrix C is replaced with a transposed Khatri-Rao matrix $\mathbf{C}'_{(k)} = C \bullet S_k \bar{C}$. In particular, this would mean providing alternate inequalities for the four key conditions laid out in Theorem 3.3.6.

CHAPTER 4

ON OUTER BI-LIPSCHITZ EXTENSIONS OF LINEAR JOHNSON-LINDENSTRAUSS EMBEDDINGS OF LOW-DIMENSIONAL SUBMANIFOLDS OF $\mathbb{R}^N$

Abstract

Let $\mathcal{M}$ be a compact d-dimensional submanifold of $\mathbb{R}^N$ with reach τ and volume $V_{\mathcal{M}}$. Fix $\epsilon \in (0,1)$. In this chapter, it is proven that a nonlinear function $f : \mathbb{R}^N \to \mathbb{R}^m$ exists with $m \leq C\left(d/\epsilon^2\right) \log\left(\frac{\sqrt[d]{V_{\mathcal{M}}}}{\tau}\right)$ such that

$$(1-\epsilon)\|\mathbf{x}-\mathbf{y}\|_2 \leq \|f(\mathbf{x})-f(\mathbf{y})\|_2 \leq (1+\epsilon)\|\mathbf{x}-\mathbf{y}\|_2 \tag{4.1}$$

holds for all $\mathbf{x} \in \mathcal{M}$ and $\mathbf{y} \in \mathbb{R}^N$. In effect, f not only serves as a bi-Lipschitz function from $\mathcal{M}$ into $\mathbb{R}^m$ with bi-Lipschitz constants close to one, but also approximately preserves all distances from points not in $\mathcal{M}$ to all points in $\mathcal{M}$ in its image. Furthermore, the proof is constructive and yields an algorithm which works well in practice. In particular, it is empirically demonstrated herein that such nonlinear functions allow for more accurate compressive nearest neighbor classification than standard linear Johnson-Lindenstrauss embeddings do in practice.

4.1 Introduction

The classical Kirszbraun theorem [62] ensures that a Lipschitz continuous function $f : S \to \mathbb{R}^m$ from a subset $S \subset \mathbb{R}^N$ into $\mathbb{R}^m$ can always be extended to a function $\tilde{f} : \mathbb{R}^N \to \mathbb{R}^m$ with the same Lipschitz constant as f. More recently, similar results have been proven for bi-Lipschitz functions, $f : S \to \mathbb{R}^m$, from $S \subset \mathbb{R}^N$ into $\mathbb{R}^m$ in the theoretical computer science literature. In particular, it was shown in [74] that outer extensions of such bi-Lipschitz functions f, $\tilde{f} : \mathbb{R}^N \to \mathbb{R}^{m+1}$, exist which both (i) approximately preserve f's Lipschitz constants, and which (ii) satisfy $\tilde{f}(\mathbf{x}) = (f(\mathbf{x}), 0)$ for all $\mathbf{x} \in S$. Narayanan and Nelson [80] then applied similar outer extension methods to a special class of the linear bi-Lipschitz maps guaranteed to exist for any given finite set $S \subset \mathbb{R}^N$ by Johnson-

Lindenstrauss (JL) lemma [59] in order prove the following remarkable result: For each finite set $S \subset \mathbb{R}^N$ and $\epsilon \in (0,1)$ there exists a **terminal embedding of** S, $f : \mathbb{R}^N \to \mathbb{R}^{O\left(\log|S|/\epsilon^2\right)}$, with the property that

$$(1-\epsilon)\|\mathbf{x}-\mathbf{y}\|_2 \leq \|f(\mathbf{x})-f(\mathbf{y})\|_2 \leq (1+\epsilon)\|\mathbf{x}-\mathbf{y}\|_2 \tag{4.2}$$

holds $\forall \mathbf{x} \in S$ and $\forall \mathbf{y} \in \mathbb{R}^N$.

In this chapter, we generalize Narayanan and Nelson's theorem for finite sets to also hold for infinite subsets $S \subset \mathbb{R}^N$, and then give a specialized variant for the case where the infinite subset $S \subset \mathbb{R}^N$ in question is a compact and smooth submanifold of $\mathbb{R}^N$. As we shall see below, generalizing this result requires us to both alter the bi-Lipschitz extension methods of [74] as well as to replace the use of embedding techniques utilizing cardinality in [80] with different JL-type embedding methods involving alternate measures of set complexity which remain meaningful for infinite sets (i.e., the Gaussian width of the unit secants of the set S in question). In the special case where S is a submanifold of $\mathbb{R}^N$, recent results bounding the Gaussian widths of the unit secants of such sets in terms of other fundamental geometric quantities (e.g., their reach, dimension, volume, etc.) [52] can then be brought to bear in order to produce terminal manifold embeddings of S into $\mathbb{R}^m$ satisfying (4.2) with m near-optimally small. Note that a non-trivial terminal embedding, f, of S satisfying (4.2) for all $\mathbf{x} \in S$ and $\mathbf{y} \in \mathbb{R}^N$ must be nonlinear. In contrast, prior work on bi-Lipschitz maps of submanifolds of $\mathbb{R}^N$ into lower dimensional Euclidean space in the mathematical data science literature have all utilized *linear* maps (see, e.g., [5, 27, 52]). As a result, it is impossible for such previously considered linear maps to serve as terminal embeddings of submanifolds of $\mathbb{R}^N$ into lower-dimensional Euclidean space without substantial modification. Another way of viewing the work carried out herein is that it constructs outer bi-Lipschitz extensions of such prior linear JL embeddings of manifolds in a way that effectively preserves their near-optimal embedding dimension in the final resulting extension. Motivating applications of terminal embeddings of submanifolds of $\mathbb{R}^N$ related to compressive classification via manifold models [21] are discussed

next.

4.1.1 Universally Accurate Compressive Classification via Noisy Manifold Data

It is one of the sad facts of life that most everyone eventually comes to accept: everything living must eventually die, you can't always win, you aren't always right, and – worst of all to the most dedicated of data scientists – there is always noise contaminating your datasets. Nevertheless, there are mitigating circumstances and achievable victories implicit in every statement above – most pertinently here, there are mountains of empirical evidence that noisy training data still permits accurate learning. In particular, when the noise level is not too large, the mere existence of a low-dimensional data model which only approximately fits your noisy training data can still allow for successful, e.g., nearest-neighbor classification using only a highly compressed version of your original training dataset (even when you know very little about the model specifics) [21]. Better quantifying these empirical observations in the context of low-dimensional manifold models is the primary motivation for our main result below.

For example, let $\mathcal{M} \subset \mathbb{R}^N$ be a d-dimensional submanifold of $\mathbb{R}^N$ (our data model), fix $\delta \in \mathbb{R}^+$ (our effective noise level), and choose $T \subseteq \ tube(\delta, \mathcal{M}) := \{\mathbf{x} \mid \exists \mathbf{y} \in \mathcal{M} \ \ with \ \|\mathbf{x} - \mathbf{y}\|_2 \leq \delta\}$ (our "noisy" and potentially high-dimensional training data). Fix $\epsilon \in (0, 1)$. For a terminal embedding $f : \mathbb{R}^N \to \mathbb{R}^m$ of $\mathcal{M}$ as per (4.2), one can see that

$$(1-\epsilon)\left\|\mathbf{z}-\mathbf{t}\right\|_2 - 2(1-\epsilon)\delta \leq \left\|f(\mathbf{z})-f(\mathbf{t})\right\|_2 \leq (1+\epsilon)\left\|\mathbf{z}-\mathbf{t}\right\|_2 + 2(1+\epsilon)\delta \tag{4.3}$$

will hold simultaneously for all $\mathbf{z} \in \mathbb{R}^N$ and $\mathbf{t} \in T$, *where f has an embedding dimension that only depends on the geometric properties of $\mathcal{M}$ (and not necessarily on $|T|$).*[1] Thus, if T includes a sufficiently dense external cover of $\mathcal{M}$, then f will allow us to approximate the distance of all $\mathbf{z} \in \mathbb{R}^N$ to $\mathcal{M}$ in the compressed embedding space via the estimator

$$\tilde{d}(f(\mathbf{z}), f(T)) := \inf_{\mathbf{t}\in T} \left\|f(\mathbf{z})-f(\mathbf{t})\right\|_2 \approx d(\mathbf{z}, \mathcal{M}) := \inf_{\mathbf{y}\in\mathcal{M}} \|\mathbf{z}-\mathbf{y}\|_2 \tag{4.4}$$

[1] One can prove (4.3) by comparing both $\mathbf{z}$ and $\mathbf{t}$ to a point $\mathbf{x}_t \in \mathcal{M}$ satisfying $\|\mathbf{t}-\mathbf{x}_t\|_2 \leq \delta$ via several applications of the (reverse) triangle inequality.

up to $O(\delta)$-error. As a result, if one has noisy data from two disjoint manifolds $\mathcal{M}_1, \mathcal{M}_2 \subset \mathbb{R}^N$, one can use this compressed $\tilde{d}$ estimator to correctly classify all data $\mathbf{z} \in \ tube(\delta, \mathcal{M}_1) \bigcup \ tube(\delta, \mathcal{M}_2)$ as being in either $T_1 := \ tube(\delta, \mathcal{M}_1)$ (class 1) or $T_2 := \ tube(\delta, \mathcal{M}_2)$ (class 2) as long as $\inf_{\mathbf{x} \in T_1, \mathbf{y} \in T_2} \|\mathbf{x} - \mathbf{y}\|_2$ is sufficiently large. In short, terminal manifold embeddings demonstrate that accurate compressive nearest-neighbor classification based on noisy manifold training data is always possible as long as the manifolds in question are sufficiently far apart (though not necessarily separable from one another by, e.g., a hyperplane, etc.). Note that in the discussion above we may in fact take $T = \ tube(\delta, \mathcal{M})$. In that case (4.3) will hold simultaneously for all $\mathbf{z} \in \mathbb{R}^N$ and $(\mathbf{t}, \delta) \in \mathbb{R}^N \times \mathbb{R}^+$ with $\mathbf{t} \in \ tube(\delta, \mathcal{M})$ so that $f : \mathbb{R}^N \to \mathbb{R}^m$ will approximately preserve the distances of all points $\mathbf{z} \in \mathbb{R}^N$ to $\ tube(\delta, \mathcal{M})$ up to errors on the order of $O(\epsilon) d(\mathbf{z}, \ tube(\delta, \mathcal{M})) + O(\delta)$ for all $\delta \in \mathbb{R}^+$. This is in fact rather remarkable when one recalls that the best achievable embedding dimension, m, here only depends on the geometric properties of the low-dimensional manifold $\mathcal{M}$ (see Theorem 4.1.1 for a detailed accounting of these dependences).

We further note that alternate applications of Theorem 4.3.2 (on which Theorem 4.1.1 depends) involving other data models are also possible. As a more explicit second example, suppose that $\mathcal{M}$ is a union of n d-dimensional affine subspaces so that its unit secants, $S_{\mathcal{M}}$ defined as per (4.7), are contained in the union of at most $\binom{n}{2} + n$ unit spheres $\subset \mathbb{S}^{N-1}$, each of dimension at most $2d + 1$. The Gaussian width (see Definition 4.2.1) of $S_{\mathcal{M}}$ can then be upper-bounded by $C\sqrt{d + \log n}$ using standard techniques, where $C \in \mathbb{R}^+$ is an absolute constant. An application of Theorem 4.3.2 now guarantees the existence of a terminal embedding $f : \mathbb{R}^N \to \mathbb{R}^{O\left(\frac{d+\log n}{\epsilon^2}\right)}$ which will allow approximate nearest subspace queries to be answered for any input point $\mathbf{z} \in \mathbb{R}^N$ using only $f(\mathbf{z})$ in the compressed $O\left(\frac{d+\log n}{\epsilon^2}\right)$-dimensional space. Even more specifically, if we choose, e.g., $\mathcal{M}$ to consist of all at most s-sparse vectors in $\mathbb{R}^N$ (i.e., so that $\mathcal{M}$ is the union of $n = \binom{N}{s}$ subspaces of $\mathbb{R}^N$), we can now see that Theorem 4.3.2 guarantees the existence of a deterministic compressed estimator (4.4) which allows for the accurate approximation of the best s-term approximation error $\inf_{\mathbf{y} \in \mathbb{R}^N \ \ at\ most\ s\ \ sparse} \|\mathbf{z} - \mathbf{y}\|_2$ for all $\mathbf{z} \in \mathbb{R}^N$ using only $f(\mathbf{z}) \in \mathbb{R}^{O(s \log(N/s))}$ as input. Note that this is only possible due to the non-linearity of f herein. In, e.g., the setting of classical compressive

sensing theory where f must be linear it is known that such good performance is impossible [20, Section 5].

4.1.2 The Main Result and a Brief Outline of Its Proof

The following theorem is proven in Section 4.4. Given a low-dimensional submanifold $\mathcal{M}$ of $\mathbb{R}^N$ it establishes the existence of a function $f : \mathbb{R}^N \to \mathbb{R}^m$ with $m \ll N$ that approximately preserves the Euclidean distances from all points in $\mathbb{R}^N$ to all points in $\mathcal{M}$. As a result, it guarantees the existence of a low-dimensional embedding which will, e.g., always allow for the correct compressed nearest-neighbor classification of images living near different well separated submanifolds of Euclidean space.

Theorem 4.1.1 (The Main Result)**.** *Let $\mathcal{M} \hookrightarrow \mathbb{R}^N$ be a compact d-dimensional submanifold of $\mathbb{R}^N$ with boundary $\partial\mathcal{M}$, finite reach $\tau_{\mathcal{M}}$ (see Definition 4.2.2), and volume $V_{\mathcal{M}}$. Enumerate the connected components of $\partial\mathcal{M}$ and let τ_i be the reach of the i^{th} connected component of $\partial\mathcal{M}$ as a submanifold of $\mathbb{R}^N$. Set $\tau := \min_i\{\tau_{\mathcal{M}}, \tau_i\}$, let $V_{\partial\mathcal{M}}$ be the volume of $\partial\mathcal{M}$, and denote the volume of the d-dimensional Euclidean ball of radius 1 by ω_d. Next,*

1. *if $d = 1$, define $\alpha_{\mathcal{M}} := \frac{20V_{\mathcal{M}}}{\tau} + V_{\partial\mathcal{M}}$, else*
2. *if $d \geq 2$, define $\alpha_{\mathcal{M}} := \frac{V_{\mathcal{M}}}{\omega_d}\left(\frac{41}{\tau}\right)^d + \frac{V_{\partial\mathcal{M}}}{\omega_{d-1}}\left(\frac{81}{\tau}\right)^{d-1}$.*

Finally, fix $\epsilon \in (0, 1)$ and define

$$\beta_{\mathcal{M}} := \left(\alpha_{\mathcal{M}}^2 + 3^d\alpha_{\mathcal{M}}\right). \tag{4.5}$$

Then, there exists a map $f : \mathbb{R}^N \to \mathbb{C}^m$ with $m \leq c\left(\ln\left(\beta_{\mathcal{M}}\right) + 4d\right)/\epsilon^2$ that satisfies

$$\left|\|f(\mathbf{x}) - f(\mathbf{y})\|_2^2 - \|\mathbf{x} - \mathbf{y}\|_2^2\right| \leq \epsilon\|\mathbf{x} - \mathbf{y}\|_2^2 \tag{4.6}$$

for all $\mathbf{x} \in \mathcal{M}$ and $\mathbf{y} \in \mathbb{R}^N$. Here $c \in \mathbb{R}^+$ is an absolute constant independent of all other quantities.

Proof. See Section 4.4. □

The remainder of the chapter is organized as follows. In Section 4.2 we review notation and state a result from [52] that bounds the Gaussian width of the unit secants of a given submanifold of $\mathbb{R}^N$ in terms of geometric quantities of the original submanifold. Next, in Section 4.3 we prove an optimal terminal embedding result for arbitrary subsets of $\mathbb{R}^N$ in terms of the Gaussian widths of their unit secants by generalizing results from the computer science literature concerning finite sets [74, 80]. See Theorem 4.3.2 therein. We then combine results from Sections 4.2 and 4.3 in order to prove our main theorem in Section 4.4. Finally, in Section 4.5 we conclude by demonstrating that terminal embeddings allow for more accurate compressive nearest neighbor classification than standard linear embeddings in practice.

4.2 Notation and Preliminaries

Below $B^N_{\ell^2}(\mathbf{x}, \gamma)$ will denote the open Euclidean ball around $\mathbf{x}$ of radius γ in $\mathbb{R}^N$. Given an arbitrary subset $S \subset \mathbb{R}^N$, we will further define $-S := \{-\mathbf{x} \mid \mathbf{x} \in S\}$ and $S \pm S := \{\mathbf{x} \pm \mathbf{y} \mid \mathbf{x}, \mathbf{y} \in S\}$. Finally, for a given $T \subset \mathbb{R}^N$ we will also let $\overline{T}$ denote its closure, and further define the normalization operator $U : \mathbb{R}^N \setminus \{\mathbf{0}\} \to \mathbb{S}^{N-1}$ to be such that $U(\mathbf{x}) := \mathbf{x}/\|\mathbf{x}\|_2$. With this notation in hand we can then define the **unit secants of** $T \subset \mathbb{R}^N$ to be

$$S_T := \overline{U\left((T - T) \setminus \{\mathbf{0}\}\right)} = \overline{\left\{\frac{\mathbf{x} - \mathbf{y}}{\|\mathbf{x} - \mathbf{y}\|_2} \mid \mathbf{x}, \mathbf{y} \in T, \mathbf{x} \neq \mathbf{y}\right\}}. \tag{4.7}$$

Note that S_T is always a compact subset of the unit sphere $\mathbb{S}^{N-1} \subset \mathbb{R}^N$, and that $S_T = -S_T$.

Herein we will call a matrix $A \in \mathbb{C}^{m \times N}$ an *ϵ-JL **map** of a set $T \subset \mathbb{R}^N$ into $\mathbb{C}^m$* if

$$(1 - \epsilon)\|\mathbf{x}\|_2^2 \leq \|A\mathbf{x}\|_2^2 \leq (1 + \epsilon)\|\mathbf{x}\|_2^2$$

holds for all $\mathbf{x} \in T$. Note that this is equivalent to $A \in \mathbb{C}^{m \times N}$ having the property that

$$\sup_{\mathbf{x} \in T \setminus \{\mathbf{0}\}} \left| \left\| A(\mathbf{x}/\|\mathbf{x}\|_2^2) \right\|_2^2 - 1 \right| = \sup_{\mathbf{x} \in U(T)} \left| \|A\mathbf{x}\|_2^2 - 1 \right| \leq \epsilon, \tag{4.8}$$

where $U(T) \subset \mathbb{R}^N$ is the normalized version of $T \setminus \{\mathbf{0}\} \subset \mathbb{R}^{\mathbf{N}}$ defined as above. Furthermore, we will say that a matrix $A \in \mathbb{C}^{m \times n}$ is an ϵ-*JL* ***embedding*** *of a set* $T \subset \mathbb{R}^n$ *into* $\mathbb{C}^m$ if A is an ϵ-JL map of

$$T - T := \{\mathbf{x} - \mathbf{y} \mid \mathbf{x}, \mathbf{y} \in T\}$$

into $\mathbb{C}^m$. Here we will be working with random matrices which will embed any fixed set T of bounded size (measured with respect to, e.g., Gaussian Width [116]) with high probability. Such matrix distributions are often called **oblivious** and discussed as randomized embeddings in the absence of any specific set T since their embedding quality can be determined independently of any properties of a given set T beyond its size. In particular, the class of oblivious **sub-Gaussian random matrices** having independent, isotropic, and sub-Gaussian rows will receive special attention below.

4.2.1 Some Common Measures of Set Size and Complexity with Associated Bounds

We will denote the cardinality of a finite set T by $|T|$. For a (potentially infinite) set $T \subset \mathbb{R}^N$ we define its **radius** and **diameter** to be

$$rad(T) := \sup_{\mathbf{x} \in T} \|\mathbf{x}\|_2$$

and

$$diam(T) := \ rad(T - T) = \sup_{\mathbf{x}, \mathbf{y} \in T} \|\mathbf{x} - \mathbf{y}\|_2,$$

respectively. Given a value $\delta \in \mathbb{R}^+$, a δ-**cover of** T (also sometimes called a δ-**net of** T) will be a subset $S \subset T$ such that the following holds

$$\forall \mathbf{x} \in T, \ \exists \mathbf{y} \in S \ \ such\ that\ \|\mathbf{x} - \mathbf{y}\|_2 \leq \delta.$$

The δ-**covering number of** T, denoted by $\mathcal{N}(T, \delta) \in \mathbb{N}$, is then the smallest achievable cardinality of a δ-cover of T. Finally, the **Gaussian width** of a set T is defined as follows.

Definition 4.2.1. *(**Gaussian Width** [116, Definition 7.5.1]). The Gaussian width of a set $T \subset \mathbb{R}^N$ is*

$$w(T) := \mathbb{E} \sup_{\mathbf{x} \in T} \langle \mathbf{g}, \mathbf{x} \rangle$$

where $\mathbf{g}$ *is a random vector with* N *independent and identically distributed (i.i.d.) mean* 0 *and variance* 1 *Gaussian entries. For a list of useful properties of the Gaussian width we refer the reader to [116, Proposition 7.5.2].*

Finally, reach is an extrinsic parameter of a subset S of Euclidean space defined based on how far away points can be from S while still having a unique closest point in S [32, 112]. The following formal definition of reach utilizes the Euclidean distance d between a given point $\mathbf{x} \in \mathbb{R}^N$ and subset $S \subset \mathbb{R}^N$.

Definition 4.2.2. *(**Reach** [32, Definition 4.1]). For a subset $S \subset \mathbb{R}^N$ of Euclidean space, the reach τ_S is*

$$\tau_S := \sup \{ t \geq 0 \mid \forall \mathbf{x} \in \mathbb{R}^n \text{ such that } d(\mathbf{x}, S) < t, \ \mathbf{x} \text{ has a unique closest point in } S \} .$$

he following theorem is a restatement of Theorem 20 in [52]. It bounds the Gaussian width of a smooth submanifold of $\mathbb{R}^N$ in terms of its dimension, reach, and volume.

Theorem 4.2.1 (Gaussian Width of the Unit Secants of a Submanifold of $\mathbb{R}^N$, Potentially with Boundary)**.** *Let $\mathcal{M} \hookrightarrow \mathbb{R}^N$ be a compact d-dimensional submanifold of $\mathbb{R}^N$ with boundary $\partial\mathcal{M}$, finite reach $\tau_{\mathcal{M}}$, and volume $V_{\mathcal{M}}$. Enumerate the connected components of $\partial\mathcal{M}$ and let τ_i be the reach of the i^{th} connected component of $\partial\mathcal{M}$ as a submanifold of $\mathbb{R}^N$. Set $\tau := \min_i\{\tau_{\mathcal{M}}, \tau_i\}$, let $V_{\partial\mathcal{M}}$ be the volume of $\partial\mathcal{M}$, and denote the volume of the d-dimensional Euclidean ball of radius 1 by ω_d. Next,*

1. *if $d = 1$, define $\alpha_{\mathcal{M}} := \frac{20V_{\mathcal{M}}}{\tau} + V_{\partial\mathcal{M}}$, else*

2. *if* $d \geq 2$, *define* $\alpha_{\mathcal{M}} := \frac{V_{\mathcal{M}}}{\omega_d}\left(\frac{41}{\tau}\right)^d + \frac{V_{\partial\mathcal{M}}}{\omega_{d-1}}\left(\frac{81}{\tau}\right)^{d-1}$.

Finally, define

$$\beta_{\mathcal{M}} := \left(\alpha_{\mathcal{M}}^2 + 3^d \alpha_{\mathcal{M}}\right). \tag{4.9}$$

Then, the Gaussian width of $\overline{U\left((\mathcal{M}-\mathcal{M})\setminus\{\mathbf{0}\}\right)}$ *satisfies*

$$w\left(S_{\mathcal{M}}\right) = w\left(\overline{U\left((\mathcal{M}-\mathcal{M})\setminus\{\mathbf{0}\}\right)}\right) \leq 8\sqrt{2}\sqrt{\ln\left(\beta_{\mathcal{M}}\right)+4d}.$$

With this Gaussian width bound in hand we can now begin the proof of our main result. The approach will be to combine Theorem 4.2.1 above with general theorems concerning the existence of outer bi-Lipschitz extensions of ϵ-JL embeddings of arbitrary subsets of $\mathbb{R}^N$ into lower-dimensional Euclidean space. These general existence theorems are proven in the next section.

4.3 The Main Bi-Lipschitz Extension Results and Their Proofs

Our first main technical result guarantees that any given JL map Φ of a special subset of $\mathbb{S}^{N-1}$ related to $\mathcal{M}$ will not only be a bi-Lipschitz map from $\mathcal{M} \subset \mathbb{R}^N$ into a lower dimensional Euclidean space $\mathbb{R}^m$, but will also have an outer bi-Lipschitz extension into $\mathbb{R}^{m+1}$. It is useful as a means of extending particular (structured) JL maps Φ of special interest in the context of, e.g., saving on memory costs [51].

Theorem 4.3.1. *Let* $\mathcal{M} \subset \mathbb{R}^N$, $\epsilon \in (0,1)$, *and suppose that* $\Phi \in \mathbb{C}^{m\times N}$ *is an* $\left(\frac{\epsilon^2}{2304}\right)$*-JL map of* $S_{\mathcal{M}} + S_{\mathcal{M}}$ *into* $\mathbb{C}^m$. *Then, there exists an outer bi-Lipschitz extension of* $\Phi : \mathcal{M} \to \mathbb{C}^m$, $f : \mathbb{R}^N \to \mathbb{C}^{m+1}$, *with the property that*

$$\left|\|f(\mathbf{x})-f(\mathbf{y})\|_2^2 - \|\mathbf{x}-\mathbf{y}\|_2^2\right| \leq \epsilon \|\mathbf{x}-\mathbf{y}\|_2^2$$

holds for all $\mathbf{x} \in \mathcal{M}$ *and* $\mathbf{y} \in \mathbb{R}^N$.

Proof. See Section 4.3.4. □

Looking at Theorem 4.3.1 we can see that an $\left(\frac{\epsilon^2}{2304}\right)$-JL map of $S_{\mathcal{M}} + S_{\mathcal{M}}$ is required in order to achieve the outer extension f of interest. This result is sub-optimal in two respects. First, the constant factor $1/2304$ is certainly not tight and can likely be improved substantially. More importantly though is the fact that ϵ is squared in the required map distortion which means that the terminal embedding dimension, $m+1$, will have to scale sub-optimally in ϵ (see Remark 4.3.1 below for details). Unfortunately, this is impossible to rectify when extending arbitrary maps Φ (see, e.g., [74]). For sub-gaussian Φ an improvement is in fact possible, however, which is the subject of our second main technical result just below. Using specialized theory for sub-gaussian matrices it demonstrates the existence of terminal JL embeddings for arbitrary subsets of $\mathbb{R}^N$ which achieve an optimal terminal embedding dimension up to constants.

Theorem 4.3.2. *Let $\mathcal{M} \subset \mathbb{R}^N$ and $\epsilon \in (0,1)$. There exists a map $f : \mathbb{R}^N \to \mathbb{C}^m$ with $m \leq c\left(\frac{w(S_{\mathcal{M}})}{\epsilon}\right)^2$ that satisfies*

$$\left| \|f(\mathbf{x}) - f(\mathbf{y})\|_2^2 - \|\mathbf{x}-\mathbf{y}\|_2^2 \right| \leq \epsilon \|\mathbf{x}-\mathbf{y}\|_2^2 \tag{4.10}$$

for all $\mathbf{x} \in \mathcal{M}$ and $\mathbf{y} \in \mathbb{R}^N$. Here $c \in \mathbb{R}^+$ is an absolute constant independent of all other quantities.

Proof. See Section 4.3.5. □

To see the optimality of the terminal embedding dimension m provided by Theorem 4.3.2 we note that functions f which satisfy (4.10) for all $\mathbf{x}, \mathbf{y} \in \mathcal{M}$ must in fact generally scale quadratically in both $w(S_{\mathcal{M}})$ and $1/\epsilon$ (see [50, Theorem 7] and [66]). We will now begin proving supporting results for both of the main technical theorems above. The first supporting results pertain to the so-called **convex hull distortion** of a given linear ϵ-JL map.

4.3.1 All Linear ϵJL Maps Provide $O(\sqrt{\epsilon})$-Convex Hull Distortion

A crucial component involved in proving our main results involves the approximate norm preservation of all points in the convex hull of a given set bounded set $S \subset \mathbb{R}^N$. Recall that the

convex hull of $S \subset \mathbb{C}^N$ is

$$\text{conv}(S) := \bigcup_{j=1}^{\infty} \left\{ \sum_{\ell=1}^{j} \alpha_\ell \mathbf{x}_\ell \mid \mathbf{x}_1, \ldots, \mathbf{x}_j \in S,\ \alpha_1, \ldots, \alpha_j \in [0,1] \text{ s.t. } \sum_{\ell=1}^{j} \alpha_\ell = 1 \right\}.$$

The next theorem states that each point in the convex hull of $S \subset \mathbb{R}^N$ can be expressed as a convex combination of at most $N+1$ points from S. Hence, the convex hulls of subsets of $\mathbb{R}^N$ are actually a bit simpler than they first appear.

Theorem 4.3.3 (Carathéadory, see, e.g., [11])**.** *Given* $S \in \mathbb{R}^N$, $\forall \mathbf{x} \in \text{conv}(S)$, $\exists \mathbf{y}_1, \ldots, \mathbf{y}_{\tilde{N}}$, $\tilde{N} = \min(|S|, N+1)$, *such that* $\mathbf{x} = \sum_{\ell=1}^{\tilde{N}} \alpha_\ell \mathbf{y}_\ell$ *for some* $\alpha_1, \ldots, \alpha_{\tilde{N}} \in [0,1]$, $\sum_{\ell=1}^{\tilde{N}} \alpha_\ell = 1$.

Finally, we say that a matrix $\Phi \in \mathbb{C}^{m \times N}$ provides **ϵ-convex hull distortion for** $S \subset \mathbb{R}^N$ if

$$|\|\Phi\mathbf{x}\|_2 - \|\mathbf{x}\|_2| \leq \epsilon$$

holds for all $\mathbf{x} \in \text{conv}(S)$. The main result of this subsection states that all linear ϵ-JL maps can provide ϵ-convex hull distortion for the unit secants of any given set. In particular, we have the following theorem which generalizes arguments in [74] for finite sets to arbitrary and potentially infinite sets.

Theorem 4.3.4. *Let* $\mathcal{M} \subset \mathbb{R}^N$, $\epsilon \in (0,1)$, *and suppose that* $\Phi \in \mathbb{C}^{m \times N}$ *is an* $\left(\frac{\epsilon^2}{4}\right)$*-JL map of* $S_\mathcal{M} + S_\mathcal{M}$ *into* $\mathbb{C}^m$. *Then,* Φ *will also provide* ϵ*-convex hull distortion for* $S_\mathcal{M}$.

The proof of Theorem 4.3.4 depends on two intermediate lemmas. The first lemma is a slight modification of Lemma 3 in [51].

Lemma 4.3.1. *Let* $S \subset \mathbb{R}^N$ *and* $\epsilon \in (0,1)$. *Then, an* ϵ*-JL map* $\Phi \in \mathbb{C}^{m \times N}$ *of the set*

$$S' = \left\{ \frac{\mathbf{x}}{\|\mathbf{x}\|_2} + \frac{\mathbf{y}}{\|\mathbf{y}\|_2}, \frac{\mathbf{x}}{\|\mathbf{x}\|_2} - \frac{\mathbf{y}}{\|\mathbf{y}\|_2} \mid \mathbf{x}, \mathbf{y} \in S \right\}$$

will satisfy

$$|\Re\left(\langle \Phi\mathbf{x}, \Phi\mathbf{y} \rangle\right) - \langle \mathbf{x}, \mathbf{y} \rangle| \leq 2\epsilon \|\mathbf{x}\|_2 \|\mathbf{y}\|_2$$

$\forall \mathbf{x}, \mathbf{y} \in S$.

Proof. If $\mathbf{x} = \mathbf{0}$ or $\mathbf{y} = \mathbf{0}$ the inequality holds trivially. Thus, suppose $\mathbf{x}, \mathbf{y} \neq 0$. Consider the normalizations $\mathbf{u} = \frac{\mathbf{x}}{\|\mathbf{x}\|_2}, \mathbf{v} = \frac{\mathbf{y}}{\|\mathbf{y}\|_2}$. The polarization identities for complex/real inner products imply that

$$\begin{aligned}
|\Re\left(\langle \Phi\mathbf{u}, \Phi\mathbf{v}\rangle\right) - \langle \mathbf{u}, \mathbf{v}\rangle| &= \frac{1}{4}\left|\Re\left(\sum_{\ell=0}^{3} i^{\ell}\left\|\Phi\mathbf{u} + i^{\ell}\Phi\mathbf{v}\right\|_2^2\right) - \left(\|\mathbf{u}+\mathbf{v}\|_2^2 - \|\mathbf{u}-\mathbf{v}\|_2^2\right)\right| \\
&= \frac{1}{4}\left|\left(\|\Phi\mathbf{u}+\Phi\mathbf{v}\|_2^2 - \|\Phi\mathbf{u}-\Phi\mathbf{v}\|_2^2\right) - \left(\|\mathbf{u}+\mathbf{v}\|_2^2 - \|\mathbf{u}-\mathbf{v}\|_2^2\right)\right| \\
&\leq \frac{1}{4}\left(\left|\|\Phi\mathbf{u}+\Phi\mathbf{v}\|_2^2 - \|\mathbf{u}+\mathbf{v}\|_2^2\right| + \left|\|\Phi\mathbf{u}-\Phi\mathbf{v}\|_2^2 - \|\mathbf{u}-\mathbf{v}\|_2^2\right|\right) \\
&\leq \frac{\epsilon}{4}\left(\|\mathbf{u}+\mathbf{v}\|_2^2 + \|\mathbf{u}-\mathbf{v}\|_2^2\right) \leq \frac{\epsilon}{2}\left(\|\mathbf{u}\|_2 + \|\mathbf{v}\|_2\right)^2 \leq 2\epsilon.
\end{aligned}$$

The result now follows by multiplying the inequality through by $\|\mathbf{x}\|_2\|\mathbf{y}\|_2$. □

Next, we see that and linear ϵ-JL maps are capable of preserving the angles between the elements of the convex hull of any bounded subset $S \subset \mathbb{R}^N$.

Lemma 4.3.2. *Suppose* $S \subset \overline{B_{\ell^2}^N(\mathbf{0}, \gamma)}$ *and* $\epsilon \in (0, 1)$. *Let* $\Phi \in \mathbb{C}^{m\times N}$ *be an* $\left(\frac{\epsilon}{2\gamma^2}\right)$*-JL map of the set* S' *defined as in Lemma 4.3.1 into* $\mathbb{C}^m$. *Then*

$$|\Re\left(\langle \Phi\mathbf{x}, \Phi\mathbf{y}\rangle\right) - \langle \mathbf{x}, \mathbf{y}\rangle| \leq \epsilon$$

holds $\forall \mathbf{x}, \mathbf{y} \in conv(S)$.

Proof. Let $\mathbf{x}, \mathbf{y} \in \text{conv}(S)$. By Theorem 4.3.3, $\exists \{\mathbf{y}_i\}_{i=1}^{\tilde{N}}, \{\mathbf{x}_i\}_{i=1}^{\tilde{N}} \subset S$ *and* $\{\alpha_\ell\}_{\ell=1}^{\tilde{N}}, \{\beta_\ell\}_{\ell=1}^{\tilde{N}} \subset [0, 1]$ with $\sum_{\ell=1}^{\tilde{N}} \alpha_\ell = \sum_{\ell=1}^{\tilde{N}} \beta_\ell = 1$ such that

$$\mathbf{x} = \sum_{\ell=1}^{\tilde{N}} \alpha_\ell \mathbf{x}_\ell, \quad \textit{and} \quad \mathbf{y} = \sum_{\ell=1}^{\tilde{N}} \beta_\ell \mathbf{y}_\ell.$$

Hence, by Lemma 4.3.1 we have that

$$\begin{aligned}|\Re\left(\langle\Phi\mathbf{x},\Phi\mathbf{y}\rangle\right)-\langle\mathbf{x},\mathbf{y}\rangle| &= \left|\sum_{\ell=1}^{\tilde{N}}\sum_{j=1}^{\tilde{N}}\alpha_\ell\beta_j\left(\Re\left(\langle\Phi\mathbf{x}_\ell,\Phi\mathbf{y}_j\rangle\right)-\langle\mathbf{x}_\ell,\mathbf{y}_j\rangle\right)\right| \\ &\leq 2\sum_{\ell=1}^{\tilde{N}}\sum_{j=1}^{\tilde{N}}\alpha_\ell\beta_j\left(\frac{\epsilon}{2\gamma^2}\right)\|\mathbf{x}_\ell\|_2\|\mathbf{y}_j\|_2 \\ &\leq \epsilon\left(\sum_{\ell=1}^{\tilde{N}}\alpha_\ell\right)\left(\sum_{j=1}^{\tilde{N}}\beta_j\right) \;=\; \epsilon.\end{aligned}$$

Here we have also used the mapping error $\left(\frac{\epsilon}{2\gamma^2}\right)$ and the fact that all norms of vectors in this case will be less than γ. □

We are now prepared to prove Theorem 4.3.4.

4.3.1.1 Proof of Theorem 4.3.4

Applying Lemma 4.3.2 with $S = S_{\mathcal{M}} = S_{\mathcal{M}} \cup -S_{\mathcal{M}}$, we note that $S' = S_{\mathcal{M}} + S_{\mathcal{M}} = (S_{\mathcal{M}} \cup -S_{\mathcal{M}}) + (S_{\mathcal{M}} \cup -S_{\mathcal{M}})$ since $S \subset \mathbb{S}^{N-1}$. Furthermore, $\gamma = 1$ in this case. Hence, $\Phi \in \mathbb{R}^{m\times N}$ being an $\left(\frac{\epsilon^2}{4}\right)$-JL map of $S_{\mathcal{M}} + S_{\mathcal{M}}$ into $\mathbb{R}^m$ implies that

$$|\Re\left(\langle\Phi\mathbf{x},\Phi\mathbf{y}\rangle\right)-\langle\mathbf{x},\mathbf{y}\rangle| \leq \frac{\epsilon^2}{2} \tag{4.11}$$

holds $\forall\mathbf{x},\mathbf{y} \in \text{conv}(S_{\mathcal{M}}) \subset \overline{B^N_{\ell^2}(\mathbf{0},1)}$. In particular, (4.11) with $\mathbf{x} = \mathbf{y}$ implies that

$$|\|\Phi\mathbf{x}\|_2-\|\mathbf{x}\|_2|\,|\|\Phi\mathbf{x}\|_2+\|\mathbf{x}\|_2| \;=\; \left|\|\Phi\mathbf{x}\|_2^2-\|\mathbf{x}\|_2^2\right| \leq \epsilon^2/2.$$

Noting that $|\|\Phi\mathbf{x}\|_2+\|\mathbf{x}\|_2| \geq \|\mathbf{x}\|_2$ we can see that the desired result holds automatically if $\|\mathbf{x}\|_2 \geq \epsilon/2$. Thus, it suffices to assume that that $\|\mathbf{x}\|_2 < \epsilon/2$, but then we are also finished since $|\|\Phi\mathbf{x}\|_2-\|\mathbf{x}\|_2| \leq \max\{\|\mathbf{x}\|_2, \|\Phi\mathbf{x}\|_2\} \leq \sqrt{\|\mathbf{x}\|_2^2+\epsilon^2/2} < \frac{\sqrt{3}}{2}\epsilon$ will hold in that case.

Remark 4.3.1. *Though Theorem 4.3.4 holds for arbitrary linear maps, we note that it has sub-optimal dependence on the distortion parameter ϵ. In particular, a linear $\left(\frac{\epsilon^2}{4}\right)$-JL map of an*

arbitrary set will generally embed that set into $\mathbb{C}^m$ with $m = \Omega(1/\epsilon^4)$ [66]. However, it has been shown in [80] that sub-Gaussian matrices will behave better with high probability, allowing for outer bi-Lipschitz extensions of JL-embeddings of finite sets into $\mathbb{R}^m$ with $m = O(1/\epsilon^2)$. In the next subsection we generalize those better scaling results for sub-Gaussian random matrices to (potentially) infinite sets.

4.3.2 Sub-Gaussian Matrices and ϵConvex Hull Distortion for Infinite Sets

Motivated by results in [80] for finite sets which achieve optimal dependence on the distortion parameter ϵ for sub-Gaussian matrices, in this section we will do the same for infinite sets using results from [116]. Our main tool will be the following result (see also [52, Theorem 4]).

Theorem 4.3.5 (See Theorem 9.1.1 and Exercise 9.1.8 in [116])**.** *Let Φ be $m \times N$ matrix whose rows are independent, isotropic, and sub-Gaussian random vectors in $\mathbb{R}^N$. Let $p \in (0,1)$ and $S \subset \mathbb{R}^N$. Then there exists a constant c depending only on the distribution of the rows of Φ such that*

$$\sup_{\mathbf{x}\in S} \left| \|\Phi\mathbf{x}\|_2 - \sqrt{m}\|\mathbf{x}\|_2 \right| \leq c \left[w(S) + \sqrt{\ln(2/p)} \cdot \ rad(S) \right]$$

holds with probability at least $1 - p$.

The main result of this section is a simple consequence of Theorem 4.3.5 together with standard results concerning Gaussian widths [116, Proposition 7.5.2].

Corollary 4.3.1. *Let $\mathcal{M} \subset \mathbb{R}^N$, $\epsilon, p \in (0,1)$, and $\Phi \in \mathbb{R}^{m\times N}$ be an $m \times N$ matrix whose rows are independent, isotropic, and sub-Gaussian random vectors in $\mathbb{R}^N$. Furthermore, suppose that*

$$m \geq \frac{c'}{\epsilon^2} \left(w\left(S_{\mathcal{M}}\right) + \sqrt{\ln(2/p)} \right)^2,$$

where c' is a constant depending only on the distribution of the rows of Φ. Then, with probability at least $1 - p$ the random matrix $\frac{1}{\sqrt{m}}\Phi$ will simultaneously be both an ϵ-JL embedding of $\mathcal{M}$ into

$\mathbb{R}^m$ *and also provide* ϵ*-convex hull distortion for* $S_{\mathcal{M}}$.

Proof. We apply Theorem 4.3.5 to $S = \text{conv}\,(S_{\mathcal{M}})$. In doing so we note that $w\,(\text{conv}\,(S_{\mathcal{M}})) = w\,(S_{\mathcal{M}})$ [116, Proposition 7.5.2], and that $rad\,(\text{conv}\,(S_{\mathcal{M}})) = 1$ since $\text{conv}\,(S_{\mathcal{M}}) \subseteq \overline{B^N_{\ell^2}(\mathbf{0}, 1)}$. The result will be that $\frac{1}{\sqrt{m}}\Phi$ provides ϵ-convex hull distortion for $S_{\mathcal{M}}$ as long as $c' \geq c^2$. Next, we note that providing ϵ-convex hull distortion for $S_{\mathcal{M}}$ implies that $\frac{1}{\sqrt{m}}\Phi$ will also approximately preserve the ℓ_2-norms of all the unit vectors in $S_{\mathcal{M}} \subset \text{conv}\,(S_{\mathcal{M}})$. In particular, $\frac{1}{\sqrt{m}}\Phi$ will be a 3ϵ-JL map of $S_{\mathcal{M}}$ into $\mathbb{R}^m$, which in turn implies that $\frac{1}{\sqrt{m}}\Phi$ will also be a 3ϵ-JL embedding of $\mathcal{M} - \mathcal{M}$ into $\mathbb{R}^m$ by linearity/rescaling. Adjusting the constant c' to account for the additional factor of 3 now yields the stated result. □

We are now prepared to prove our general theorems regarding outer bi-Lipschitz extensions of JL-embeddings of potentially infinite sets.

4.3.3 Outer Bi-Lipschitz Extension Results for JL-embeddings of General Sets

Before we can prove our final results for general sets we will need two supporting lemmas. They are adapted from the proofs of analogous results in [74, 80] for finite sets.

Lemma 4.3.3. *Let* $\mathcal{M} \subset \mathbb{R}^N$, $\epsilon \in (0, 1)$, *and suppose that* $\Phi \in \mathbb{C}^{m\times N}$ *provides* ϵ*-convex hull distortion for* $S_{\mathcal{M}}$. *Then, there exists a function* $g : \mathbb{R}^N \to \mathbb{C}^m$ *such that*

$$|\Re\left(\langle g(\mathbf{y}), \Phi\mathbf{x}\rangle\right) - \langle \mathbf{y}, \mathbf{x}\rangle| \leq 2\epsilon\|\mathbf{y}\|_2\|\mathbf{x}\|_2 \tag{4.12}$$

holds for all $\mathbf{x} \in \overline{\mathcal{M}} - \overline{\mathcal{M}}$ *and* $\mathbf{y} \in \mathbb{R}^N$.

Proof. First, we note that (4.12) holds trivially for $\mathbf{y} = \mathbf{0}$ as long as $g(\mathbf{0}) = \mathbf{0}$. Thus, it suffices to consider nonzero $\mathbf{y}$. Second, we claim that it suffices to prove the existence of a function $g : \mathbb{R}^N \to \mathbb{C}^m$ that satisfies both of the following properties

1. $\|g(\mathbf{y})\|_2 \leq \|\mathbf{y}\|_2$, and

2. $|\Re\left(\langle g(\mathbf{y}), \Phi\mathbf{x}'\rangle\right) - \langle \mathbf{y}, \mathbf{x}'\rangle| \leq \epsilon\|\mathbf{y}\|_2$ for all $\mathbf{x}'$ in a finite $(\epsilon/2\max\{1, \|\Phi\|_{2\to 2}\})$-cover C of $S_{\mathcal{M}}$,

for all $\mathbf{y} \in \mathbb{R}^N$. To see why, fix $\mathbf{y} \neq \mathbf{0}$, $\mathbf{x} \in S_{\mathcal{M}}$, and let $\mathbf{x}' \in C \subset S_{\mathcal{M}}$ satisfy $\|\mathbf{x} - \mathbf{x}'\|_2 \leq \epsilon/2\max\{1, \|\Phi\|_{2\to 2}\}$. We can see that any function g satisfying both of the properties above will have

$$\begin{aligned}
|\Re\left(\langle g(\mathbf{y}), \Phi\mathbf{x}\rangle\right) - \langle \mathbf{y}, \mathbf{x}\rangle| &= |\Re\left(\langle g(\mathbf{y}), \Phi\mathbf{x}'\rangle\right) + \Re\left(\langle g(\mathbf{y}), \Phi\left(\mathbf{x} - \mathbf{x}'\right)\rangle\right) - \langle \mathbf{y}, (\mathbf{x} - \mathbf{x}')\rangle - \langle \mathbf{y}, \mathbf{x}'\rangle| \\
&\leq |\Re\left(\langle g(\mathbf{y}), \Phi\mathbf{x}'\rangle\right) - \langle \mathbf{y}, \mathbf{x}'\rangle| + |\langle g(\mathbf{y}), \Phi\left(\mathbf{x} - \mathbf{x}'\right)\rangle| + |\langle \mathbf{y}, (\mathbf{x} - \mathbf{x}')\rangle| \\
&\leq \epsilon\|\mathbf{y}\|_2 + \|g(\mathbf{y})\|_2\|\Phi\|_{2\to 2}\|\mathbf{x} - \mathbf{x}'\|_2 + \|\mathbf{y}\|_2\|\|\mathbf{x} - \mathbf{x}'\|_2
\end{aligned}$$

where the second property was used in the last inequality above.

Appealing to the first property above we can now also see that $|\Re\left(\langle g(\mathbf{y}), \Phi\mathbf{x}\rangle\right) - \langle \mathbf{y}, \mathbf{x}\rangle| \leq 2\epsilon\|\mathbf{y}\|_2$ will hold. Finally, as a consequence of the definition of $S_{\mathcal{M}}$, we therefore have that (4.12) will hold for all $\mathbf{x} \in \mathcal{M} - \mathcal{M}$ and $\mathbf{y} \in \mathbb{R}^N$ whenever Properties 1 and 2 hold above. Showing that (4.12) holds all $\mathbf{x} \in \overline{\mathcal{M}} - \overline{\mathcal{M}}$ more generally can be proven by contradiction using a limiting argument combined with the fact that both the right and left hand sides of (4.12) are continuous in $\mathbf{x}$ for fixed $\mathbf{y}$. Hence, we have reduced the proof to constructing a function g that satisfies both Properties 1 and 2 above.

Let

$$g(\mathbf{y}) := \arg\min_{\mathbf{v} \in \overline{B^{2m}_{\ell^2}(\mathbf{0}, \|\mathbf{y}\|_2)}} \max_{\boldsymbol{\lambda} \in \overline{B^{|C|}_{\ell^1}(\mathbf{0}, 1)}} h_{\mathbf{y}}(\mathbf{v}, \boldsymbol{\lambda}), \quad \text{where} \tag{4.13}$$

$$h_{\mathbf{y}}(\mathbf{v}, \boldsymbol{\lambda}) := \sum_{\mathbf{u} \in C} \left(\lambda_{\mathbf{u}}\left(\langle \mathbf{y}, \mathbf{u}\rangle - \Re\left(\langle \mathbf{v}, \Phi\mathbf{u}\rangle\right)\right) - \epsilon\,|\lambda_{\mathbf{u}}| \cdot \|\mathbf{y}\|_2\right) \tag{4.14}$$

where we identify $\mathbb{C}^m$ with $\mathbb{R}^{2m}$ above. Note that Property 1 above is guaranteed by definition (4.13). Furthermore, we note that if

$$\max_{\boldsymbol{\lambda} \in \{\pm \mathbf{e}_j\}_{j=1}^{|C|}} h_{\mathbf{y}}(g(\mathbf{y}), \boldsymbol{\lambda}) = \max_{\mathbf{u} \in C}\left(|\langle \mathbf{y}, \mathbf{u}\rangle - \Re\left(\langle g(\mathbf{y}), \Phi\mathbf{u}\rangle\right)| - \epsilon\|\mathbf{y}\|_2\right) \leq \max_{\boldsymbol{\lambda} \in \overline{B^{|C|}_{\ell^1}(\mathbf{0}, 1)}} h_{\mathbf{y}}(g(\mathbf{y}), \boldsymbol{\lambda}) \leq 0$$

then Property 2 above will hold as well. Thus, it suffices to show that $\min_{\mathbf{v}\in\overline{B^{2m}_{\ell^2}(\mathbf{0},\|\mathbf{y}\|_2)}} \max_{\boldsymbol{\lambda}\in\overline{B^{|C|}_{\ell^1}(\mathbf{0},1)}} h_{\mathbf{y}}(\mathbf{v},\boldsymbol{\lambda}) \leq 0$ always holds in order to finish the proof.

Noting that $h_{\mathbf{y}} : \mathbb{R}^{2m+|C|} \mapsto \mathbb{R}$ defined in (4.14) is continuous, convex (affine) in $\mathbf{v}$, concave in $\boldsymbol{\lambda}$, and further noting that both $\overline{B^{|C|}_{\ell^1}(\mathbf{0},1)}$ and $\overline{B^{2m}_{\ell^2}(\mathbf{0},\|\mathbf{y}\|_2)}$ are compact and convex, we may apply Von Neumann's minimax theorem [83] to see that

$$\min_{\mathbf{v}\in\overline{B^{2m}_{\ell^2}(\mathbf{0},\|\mathbf{y}\|_2)}} \max_{\boldsymbol{\lambda}\in\overline{B^{|C|}_{\ell^1}(\mathbf{0},1)}} h_{\mathbf{y}}(\mathbf{v},\boldsymbol{\lambda}) = \max_{\boldsymbol{\lambda}\in\overline{B^{|C|}_{\ell^1}(\mathbf{0},1)}} \min_{\mathbf{v}\in\overline{B^{2m}_{\ell^2}(\mathbf{0},\|\mathbf{y}\|_2)}} h_{\mathbf{y}}(\mathbf{v},\boldsymbol{\lambda})$$

holds. Thus, we will in fact be finished if we can show that $\min_{\mathbf{v}\in\overline{B^{2m}_{\ell^2}(\mathbf{0},\|\mathbf{y}\|_2)}} h_{\mathbf{y}}(\mathbf{v},\boldsymbol{\lambda}) \leq 0$ holds for each $\boldsymbol{\lambda} \in \overline{B^{|C|}_{\ell^1}(\mathbf{0},1)}$. By rescaling this in turn is implied by showing that $\forall \mathbf{u} \in \text{conv}(C \cup -C)$ $\exists \mathbf{v} \in \overline{B^{2m}_{\ell^2}(\mathbf{0},\|\mathbf{y}\|_2)}$ such that

$$(\langle \mathbf{y},\mathbf{u}\rangle - \Re\left(\langle \mathbf{v},\Phi\mathbf{u}\rangle\right) - \epsilon\|\mathbf{y}\|_2) \leq 0 \tag{4.15}$$

holds.

To prove (4.15) for a fixed $\mathbf{u} \in \text{conv}(C \cup -C) \subseteq \text{conv}(S_{\mathcal{M}} \cup -S_{\mathcal{M}}) = \text{conv}(S_{\mathcal{M}})$ and thereby establish the stated theorem, one may set $\mathbf{v} = \|\mathbf{y}\|_2 \frac{\Phi\mathbf{u}}{\|\Phi\mathbf{u}\|_2}$. Doing so we see that the left side of (4.15) simplifies to $\langle \mathbf{y},\mathbf{u}\rangle - \|\mathbf{y}\|_2\|\Phi\mathbf{u}\|_2 - \epsilon\|\mathbf{y}\|_2$. To finish, we note that indeed

$$\begin{aligned}\langle \mathbf{y},\mathbf{u}\rangle - \|\mathbf{y}\|_2\|\Phi\mathbf{u}\|_2 - \epsilon\|\mathbf{y}\|_2 &\leq \|\mathbf{y}\|_2\|\mathbf{u}\|_2 - \|\mathbf{y}\|_2\|\Phi\mathbf{u}\|_2 - \epsilon\|\mathbf{y}\|_2 \\ &\leq \|\mathbf{y}\|_2(\|\mathbf{u}\|_2 - \|\Phi\mathbf{u}\|_2 - \epsilon) \leq 0\end{aligned}$$

will then hold since Φ provides ϵ-convex hull distortion for $S_{\mathcal{M}}$. □

Lemma 4.3.4. *Let $\mathcal{M} \subset \mathbb{R}^N$ be non-empty, $\epsilon \in (0,1)$, and suppose that $\Phi \in \mathbb{C}^{m\times N}$ provides ϵ-convex hull distortion for $S_{\mathcal{M}}$. Then, there exists an outer bi-Lipschitz extension of Φ, $f : \mathbb{R}^N \to$*

$\mathbb{C}^{m+1}$, *with the property that*

$$\left| \|f(\mathbf{x}) - f(\mathbf{y})\|_2^2 - \|\mathbf{x} - \mathbf{y}\|_2^2 \right| \leq 24\epsilon \|\mathbf{x} - \mathbf{y}\|_2^2 \tag{4.16}$$

holds for all $\mathbf{x} \in \mathcal{M}$ *and* $\mathbf{y} \in \mathbb{R}^N$.

Proof. Given $\mathbf{y} \in \mathbb{R}^N$ let $\mathbf{y}_{\mathcal{M}} \in \overline{\mathcal{M}}$ satisfy $\|\mathbf{y} - \mathbf{y}_{\mathcal{M}}\|_2 = \inf_{\mathbf{x} \in \overline{\mathcal{M}}} \|\mathbf{y} - \mathbf{x}\|_2$.[2] We define

$$f(\mathbf{y}) := \begin{cases} (\Phi \mathbf{y}, 0) & \text{if } \mathbf{y} \in \overline{\mathcal{M}} \\ \left(\Phi \mathbf{y}_{\mathcal{M}} + g(\mathbf{y} - \mathbf{y}_{\mathcal{M}}), \sqrt{\|\mathbf{y} - \mathbf{y}_{\mathcal{M}}\|_2^2 - \|g\left(\mathbf{y} - \mathbf{y}_{\mathcal{M}}\right)\|_2^2} \right) & \text{if } \mathbf{y} \notin \overline{\mathcal{M}} \end{cases}$$

where g is defined as in Lemma 4.3.3. Fix $\mathbf{x} \in \mathcal{M}$. If $\mathbf{y} \in \overline{\mathcal{M}}$ then $\|f(\mathbf{x}) - f(\mathbf{y})\|_2^2 = \|\Phi(\mathbf{x} - \mathbf{y})\|_2^2$, and so we can see that $\left| \|f(\mathbf{x}) - f(\mathbf{y})\|_2^2 - \|\mathbf{x} - \mathbf{y}\|_2^2 \right| \leq 3\epsilon \|\mathbf{x} - \mathbf{y}\|_2^2$ will hold since Φ will be 3ϵ-JL embedding of $\overline{\mathcal{M}} - \overline{\mathcal{M}}$ (recall the proof of Corollary 4.3.1 and note the linearity of Φ). Thus, it suffices to consider a fixed $\mathbf{y} \notin \overline{\mathcal{M}}$. In that case we have

$$\begin{aligned} \|f(\mathbf{x}) - f(\mathbf{y})\|_2^2 &= \|\Phi(\mathbf{x} - \mathbf{y}_{\mathcal{M}}) - g\left(\mathbf{y} - \mathbf{y}_{\mathcal{M}}\right)\|_2^2 + \|\mathbf{y} - \mathbf{y}_{\mathcal{M}}\|_2^2 - \|g\left(\mathbf{y} - \mathbf{y}_{\mathcal{M}}\right)\|_2^2 \\ &= \|\mathbf{y} - \mathbf{y}_{\mathcal{M}}\|_2^2 + \|\Phi(\mathbf{x} - \mathbf{y}_{\mathcal{M}})\|_2^2 - 2\Re\left(\langle g\left(\mathbf{y} - \mathbf{y}_{\mathcal{M}}\right), \Phi(\mathbf{x} - \mathbf{y}_{\mathcal{M}})\rangle\right) \end{aligned} \tag{4.17}$$

by the polarization identity and parallelogram law.

Similarly we have that

$$\|\mathbf{x} - \mathbf{y}\|_2^2 = \|(\mathbf{x} - \mathbf{y}_{\mathcal{M}}) - (\mathbf{y} - \mathbf{y}_{\mathcal{M}})\|_2^2 = \|\mathbf{y} - \mathbf{y}_{\mathcal{M}}\|_2^2 + \|\mathbf{x} - \mathbf{y}_{\mathcal{M}}\|_2^2 - 2\langle \mathbf{y} - \mathbf{y}_{\mathcal{M}}, \mathbf{x} - \mathbf{y}_{\mathcal{M}} \rangle. \tag{4.18}$$

[2] One can see that it suffices to approximately compute $\mathbf{y}_{\mathcal{M}}$ in order to achieve (4.16) up to a fixed precision.

Subtracting (4.18) from (4.17) we can now see that

$$\begin{aligned}\left|\|f(\mathbf{x})-f(\mathbf{y})\|_2^2-\|\mathbf{x}-\mathbf{y}\|_2^2\right| \leq\ & \left|\|\Phi(\mathbf{x}-\mathbf{y}_{\mathcal{M}})\|_2^2-\|\mathbf{x}-\mathbf{y}_{\mathcal{M}}\|_2^2\right| + \\ & 2\left|\Re\left(\left\langle g\left(\mathbf{y}-\mathbf{y}_{\mathcal{M}}\right), \Phi(\mathbf{x}-\mathbf{y}_{\mathcal{M}})\right\rangle\right)-\left\langle \mathbf{y}-\mathbf{y}_{\mathcal{M}}, \mathbf{x}-\mathbf{y}_{\mathcal{M}}\right\rangle\right| \\ \leq\ & 3\epsilon\|\mathbf{x}-\mathbf{y}_{\mathcal{M}}\|_2^2 + 4\epsilon\|\mathbf{y}-\mathbf{y}_{\mathcal{M}}\|_2\|\mathbf{x}-\mathbf{y}_{\mathcal{M}}\|_2 \\ \leq\ & 3\epsilon\|\mathbf{x}-\mathbf{y}_{\mathcal{M}}\|_2^2 + 2\epsilon\left(\|\mathbf{y}-\mathbf{y}_{\mathcal{M}}\|_2^2+\|\mathbf{x}-\mathbf{y}_{\mathcal{M}}\|_2^2\right) \end{aligned} \tag{4.19}$$

where the second inequality again appeals to Φ being a 3ϵ-JL embedding of $\overline{\mathcal{M}}-\overline{\mathcal{M}}$, and to Lemma 4.3.3. Considering (4.19) we can see that

- $\|\mathbf{y}-\mathbf{y}_{\mathcal{M}}\|_2 \leq \|\mathbf{y}-\mathbf{x}\|_2$ by the definition of $\mathbf{y}_{\mathcal{M}}$, and so
- $\|\mathbf{x}-\mathbf{y}_{\mathcal{M}}\|_2 \leq \|\mathbf{x}-\mathbf{y}\|_2+\|\mathbf{y}-\mathbf{y}_{\mathcal{M}}\|_2 \leq 2\|\mathbf{x}-\mathbf{y}\|_2$, and thus
- $\|\mathbf{y}-\mathbf{y}_{\mathcal{M}}\|_2^2+\|\mathbf{x}-\mathbf{y}_{\mathcal{M}}\|_2^2 \leq (\|\mathbf{y}-\mathbf{y}_{\mathcal{M}}\|_2+\|\mathbf{x}-\mathbf{y}_{\mathcal{M}}\|_2)^2 \leq 9\|\mathbf{x}-\mathbf{y}\|_2^2$.

Using the last two inequalities above in (4.19) now yields the stated result. □

We are now prepared to prove the two main results of this section.

4.3.4 Proof of Theorem 4.3.1

Apply Theorem 4.3.4 with $\epsilon \leftarrow \epsilon/24$ in order obtain $\epsilon/24$-convex hull distortion for $S_{\mathcal{M}}$ via Φ. Then, apply Lemma 4.3.4.

4.3.5 Proof of Theorem 4.3.2

To begin we apply Corollary 4.3.1 with, e.g., $p = 1/2$ to demonstrate that an $\left\lceil \frac{c''}{\epsilon^2}\left(w(S_{\mathcal{M}})+\sqrt{\ln(4)}\right)^2\right\rceil \times N$ matrix with i.i.d. standard normal random entries can provide $(\epsilon/24)$-convex hull distortion for $S_{\mathcal{M}}$, where c'' is an absolute constant. Hence, such a matrix Φ exists. An application of Lemma 4.3.4 now finishes the proof.

4.4 The Proof of Theorem 4.1.1

We apply Theorem 4.3.2 together with Theorem 4.2.1 to bound the Gaussian width of $S_{\mathcal{M}}$.

4.5 A Numerical Evaluation of Terminal Embeddings

In this section we consider several variants of the optimization approach mentioned in Section 3.3 of [80] for implementing a terminal embedding $f : \mathbb{R}^N \to \mathbb{R}^{m+1}$ of a finite set $X \subset \mathbb{R}^N$. In effect, this requires us to implement a function satisfying two sets of constraints from [80, Section 3.3] that are analogous to the two properties of $g : \mathbb{R}^N \to \mathbb{C}^m$ listed at the beginning of the proof of Lemma 4.3.3. See Lines 1 and 2 of Algorithm 4.1 for a concrete example of one type of constrained minimization problem solved herein to accomplish this task.

Algorithm 4.1 Terminal Embedding of a Finite Set

Input: $\epsilon \in (0,1)$, $X \subset \mathbb{R}^N$, $|X| =: n$, $S \subset \mathbb{R}^N$, $|S| =: n'$, $m \in \mathbb{N}$ with $m < N$, a random matrix with i.i.d. standard Gaussian entries, $\Phi \in \mathbb{R}^{m \times N}$, rescaled to perform as a JL embedding matrix $\Pi := \frac{1}{\sqrt{m}}\Phi$

Output: A terminal embedding of X, $f \in \mathbb{R}^N \to \mathbb{R}^{m+1}$, evaluated on S

for $\mathbf{u} \in S$ **do**

1) Compute $\mathbf{x}_{NN} := \operatorname{argmin}_{\mathbf{x} \in X} \|\mathbf{u} - \mathbf{x}\|_2$

2) Solve the following constrained minimization problem to compute a minimizer $\mathbf{u}' \in \mathbb{R}^m$

$$\begin{aligned} \text{Minimize} \quad & h_{\mathbf{u},\mathbf{x}_{NN}}(\mathbf{z}) := \|\mathbf{z}\|_2^2 + 2\langle \Pi(\mathbf{u} - \mathbf{x}_{NN}), \mathbf{z} \rangle \\ \text{subject to} \quad & \|\mathbf{z}\|_2 \leq \|\mathbf{u} - \mathbf{x}_{NN}\|_2 \\ & |\langle \mathbf{z}, \Pi(\mathbf{x} - \mathbf{x}_{NN}) \rangle - \langle \mathbf{u} - \mathbf{x}_{NN}, \mathbf{x} - \mathbf{x}_{NN} \rangle| \leq \epsilon \|\mathbf{u} - \mathbf{x}_{NN}\|_2 \|\mathbf{x} - \mathbf{x}_{NN}\|_2 \ , \ \forall \mathbf{x} \in X \end{aligned}$$

3) Compute $f : \mathbb{R}^N \to \mathbb{R}^{m+1}$ at $\mathbf{u}$ via

$$f(\mathbf{u}) := \begin{cases} (\Pi\mathbf{u}, 0), & \mathbf{u} \in X \\ (\Pi\mathbf{x}_{NN} + \mathbf{u}', \sqrt{\|\mathbf{u} - \mathbf{x}_{NN}\|_2^2 - \|\mathbf{u}'\|_2^2}), & \mathbf{u} \notin X \end{cases}$$

end for

Crucially, we note that any choice $\mathbf{u}' \in \mathbb{R}^m$ *of a* $\mathbf{z}$ *satisfying the two sets of constraints in Line 2 of Algorithm 4.1 for a given* $\mathbf{u} \in \mathbb{R}^N$ *is guaranteed to correspond to an evaluation of a valid terminal embedding of* X *at* $\mathbf{u}$ *in Line 3.* This leaves the choice of the objective function, $h_{\mathbf{u},\mathbf{x}_{NN}}$, minimized in Line 2 of Algorithm 4.1 open to change without effecting its theoretical performance guarantees. Given this setup, several heretofore unexplored practical questions about terminal embeddings immediately present themselves. These include:

1. Repeatedly solving the optimization problem in Line 2 of Algorithm 4.1 to evaluate a terminal embedding of X on S is certainly more computationally expensive than simply evaluating a

standard linear Johnson-Lindenstrauss (JL) embedding of X on S instead. How do terminal embeddings empirically compare to standard linear JL embedding matrices on real-world data in the context of, e.g., compressive classification? When, if ever, is their additional computational expense actually justified in practice?

2. Though any choice of objective function $h_{\mathbf{u},\mathbf{x}_{NN}}$ in Line 2 of Algorithm 4.1 must result in a terminal embedding f of X based on the available theory, some choices probably lead to better empirical performance than others. What's a good default choice?

3. How much dimensionality reduction are terminal embeddings capable of in the context of, e.g., accurate compressive classification using real-world data?

In keeping with the motivating application discussed in Section 4.1.1 above, we will explore some preliminary answers to these three questions in the context of compressive classification based on real-world data below.

4.5.1 A Comparison Criteria: Compressive Nearest Neighbor Classification

Given a labelled data set $\mathcal{D} \subset \mathbb{R}^N$ with label set $\mathcal{L}$, we let $Label : \mathcal{D} \to \mathcal{L}$ denote the function which assigns the correct label to each element of the data set. To address the three questions above we will use compressive nearest neighbor classification accuracy as a primary measure of an embedding strategy's quality. See Algorithm 4.2 for a detailed description of how this accuracy can be computed for a given data set $\mathcal{D}$.

Algorithm 4.2 Measuring Compressive Nearest Neighbor Classification Accuracy

Input: $\epsilon \in (0,1)$, A labeled data set $\mathcal{D} \subset \mathbb{R}^N$ split into two disjoint subsets: A training set $X \subset \mathcal{D}$ with $|X| =: n$, and a test set $S \subset \mathcal{D}$ with $|S| =: n'$, such that $S \cap X = \emptyset$. A compressive dimension $m < N$.

Output: Successful Nearest Neighbor Classification Percentage for Data Embedded in $\mathbb{R}^{m+1}$

Fix $f : \mathbb{R}^N \to \mathbb{R}^{m+1}$, an embedding of the training data $X \subset \mathbb{R}^N$ into $\mathbb{R}^{m+1}$ satisfying

$$(1-\epsilon)\|\mathbf{x}-\mathbf{y}\|_2 \leq \|f(\mathbf{x})-f(\mathbf{y})\|_2 \leq (1+\epsilon)\|\mathbf{x}-\mathbf{y}\|_2$$

for all $\mathbf{x}, \mathbf{y} \in X$. [Note: this can either be a JL-embedding of X, or a stronger terminal embedding of X.]

% *Embed the training data into* $\mathbb{R}^{m+1}$.
for $\mathbf{x} \in X$ **do**
    Compute $f(\mathbf{x})$ using, e.g., Algorithm 4.1.
end for

% *Classify the test data using its embedded distance in* $\mathbb{R}^{m+1}$.
$p = 0$
for $\mathbf{u} \in S$ **do**
    Compute $f(\mathbf{u})$ using, e.g., Algorithm 4.1
    Compute $\mathbf{x} = \text{argmin}_{\mathbf{y} \in X} \|f(\mathbf{u}) - f(\mathbf{y})\|_2$
    if $Label(\mathbf{u}) = Label(\mathbf{x})$ **then**
        $p = p + 1$
    end if
end for

Output the Successful Classification Percentage = $\frac{p}{n'} \times 100\%$

Note that Algorithm 4.2 can be used to help us compare the quality of different embedding strategies. For example, one can use Algorithm 4.2 to compare different choices of objective functions $h_{\mathbf{u},\mathbf{x}_{NN}}$ in Line 2 of Algorithm 4.1 against one another by running Algorithm 4.2 multiple times on the same training and test data sets while only varying the implementation of Algorithm 4.1 each time. This is exactly the type of approach we will use below. Of course, before we can begin we must first decide on some labelled data sets $\mathcal{D}$ to use in our classification experiments.

4.5.2 Our Choice of Training and Testing Data Sets

Herein we consider two standard benchmark image data sets which allow for accurate uncompressed Nearest Neighbor (NN) classification. The images in each data set can then be vectorized

and embedded using, e.g., Algorithm 4.1 in order to test the accuracies of compressed NN classification variants against both one another, as well as against standard uncompressed NN classification. These benchmark data sets are as follows.

Figure 4.1 Example images from the MNIST data set (left), and the COIL-100 data set (right).

The MNIST data set [67, 22] consists of 60,000 training images of 28×28-pixel grayscale hand-written images of the digits 0 through 9. Thus, MNIST has 10 labels to correctly classify between, and $N = 28^2 = 784$. For all experiments involving the MNIST dataset, $n/10$ digits of each type are selected uniformly at random to form the training set X, for a total of n vectorized training images in $\mathbb{R}^{784}$. Then, 100 digits of each type are randomly selected from those not used for training in order to form the test set S, leading to a total of $n' = 1000$ vectorized test images in $\mathbb{R}^{784}$. See the left side of Figure 4.1 for example MNIST images.

The COIL-100 data set [82] is a collection of 128×128-pixel color images of 100 objects, each photographed 72 times where the object has been rotated by 5 degrees each time to get a complete rotation. However, only the green color channel of each image is used herein for simplicity. Thus, herein COIL-100 consists of $7,200$ total vectorized images in $\mathbb{R}^N$ with $N = 128^2 = 16,384$, where each image has one of 100 different labels (72 images per label). For all experiments involving this COIL-100 data set, $n/100$ training images are down sampled from each of the 100 objects' rotational image sequences. Thus, the training sets each contain $n/100$ vectorized images of each object, each photographed at rotations of $\approx 36000/n$ degrees (rounded to multiples of 5). The resulting training data sets therefore all consist of n vectorized images in $\mathbb{R}^{16,384}$. After forming

each training set, 10 images of each type are then randomly selected from those not used for training in order to form the test set S, leading to a total of $n' = 1000$ vectorized test images in $\mathbb{R}^{16,384}$ per experiment. See the right side of Figure 4.1 for example COIL-100 images.

4.5.3 A Comparison of Four Embedding Strategies via NN Classification

In this section we seek to better understand (*i*) when terminal embeddings outperform standard JL-embedding matrices in practice with respect to accurate compressive NN classification, (*ii*) what type of objective functions $h_{\mathbf{u},\mathbf{x}_{NN}}$ in Line 2 of Algorithm 4.1 perform best in practice when computing a terminal embedding, and (*iii*) how much dimensionality reduction one can achieve with a terminal embedding without appreciably degrading standard NN classification results in practice. To gain insight on these three questions we will compare the following four embedding strategies in the context of NN classification. These strategies begin with the most trivial linear embeddings (i.e., the identity map) and slowly progress toward extremely non-linear terminal embeddings.

(a) **Identity:** We use the data in its original uncompressed form (i.e., we use the trivial embedding $f : \mathbb{R}^N \to \mathbb{R}^N$ defined by $f(\mathbf{u}) = \mathbf{u}$ in Algorithm 4.2). Here the embedding dimension $m + 1$ is always fixed to be N.

(b) **Linear:** We compressively embed our training data X using a JL embedding. More specifically, we generate an $m \times N$ random matrix Φ with i.i.d. standard Gaussian entries and then set $f : \mathbb{R}^N \to \mathbb{R}^{m+1}$ to be $f(\mathbf{u}) := \left(\frac{1}{\sqrt{m}}\Phi\mathbf{u},\ 0\right)$ in Algorithm 4.2 for various choices of m. It is then hoped that f will embed the test data S well in addition to the training data X. Note that this embedding choice for f is consistent with Algorithm 4.1 where one lets $X = X \cup S$ when evaluating Line 3, thereby rendering the minimization problem in Line 2 irrelevant.

(c) **A Valid Terminal Embedding That's as Linear as Possible:** To minimize the pointwise difference between the terminal embedding f computed by Algorithm 4.1 and the linear map defined above in (b), we may choose the objective function in Line 2 of Algorithm 4.1 to be $h_{\mathbf{u},\mathbf{x}_{NN}}(\mathbf{z}) := \langle \Pi(\mathbf{x}_{NN} - \mathbf{u}), \mathbf{z} \rangle$. To see why solving this minimizes the pointwise difference between f and the linear map in (b), let $\mathbf{u}'$ be such that $\langle \Pi(\mathbf{x}_{NN} - \mathbf{u}), \mathbf{z} \rangle$ is minimal subject to

the constraints in Line 2 of Algorithm 4.1 when $\mathbf{z} = \mathbf{u}'$. Since $\mathbf{u}$ and $\mathbf{x}_{NN}$ are fixed here, we note that $\mathbf{z} = \mathbf{u}'$ will then also minimize

$$\begin{aligned}
&\|\Pi(\mathbf{x}_{NN} - \mathbf{u})\|_2^2 + 2\langle \Pi(\mathbf{x}_{NN} - \mathbf{u}), \mathbf{z}\rangle + \|\mathbf{u} - \mathbf{x}_{NN}\|_2^2 \\
= \; &\|\Pi(\mathbf{x}_{NN} - \mathbf{u})\|_2^2 + \|\mathbf{z}\|_2^2 + 2\langle \Pi(\mathbf{x}_{NN} - \mathbf{u}), \mathbf{z}\rangle + \|\mathbf{u} - \mathbf{x}_{NN}\|_2^2 - \|\mathbf{z}\|_2^2 \\
= \; &\|\Pi(\mathbf{x}_{NN} - \mathbf{u}) + \mathbf{z}\|_2^2 + \|\mathbf{u} - \mathbf{x}_{NN}\|_2^2 - \|\mathbf{z}\|_2^2 \\
= \; &\left\| \left(\Pi\mathbf{x}_{NN} + \mathbf{z}, \sqrt{\|\mathbf{u} - \mathbf{x}_{NN}\|_2^2 - \|\mathbf{z}\|_2^2} \right) - (\Pi\mathbf{u}, 0) \right\|_2^2
\end{aligned}$$

subject to the desired constraints. Hence, we can see that choosing $\mathbf{z} = \mathbf{u}'$ as above is equivalent to minimizing $\|f(\mathbf{u}) - (\Pi\mathbf{u}, 0)\|_2^2$ over all valid choices of terminal embeddings f that satisfy the existing theory.

(d) **A Terminal Embedding Computed by Algorithm 4.1 as Presented:** This terminal embedding is computed using Algorithm 4.1 exactly as it is formulated above (i.e., with the objective function in Line 2 chosen to be $h_{\mathbf{u},\mathbf{x}_{NN}}(\mathbf{z}) := \|\mathbf{z}\|_2^2 + 2\langle \Pi(\mathbf{u} - \mathbf{x}_{NN}), \mathbf{z}\rangle$). Note that this choice of objective function was made to encourage non-linearity in the resulting terminal embedding f computed by Algorithm 4.1. To understand our intuition for making this choice of objective function in order to encourage non-linearity in f, suppose that $\|\mathbf{z}\|_2^2 + 2\langle \Pi(\mathbf{u} - \mathbf{x}_{NN}), \mathbf{z}\rangle$ is minimal subject to the constraints in Line 2 of Algorithm 4.1 when $\mathbf{z} = \mathbf{u}'$. Since $\mathbf{u}$ and $\mathbf{x}_{NN}$ are fixed independently of $\mathbf{z}$ this means that $\mathbf{z} = \mathbf{u}'$ then also minimize

$$\|\mathbf{z}\|_2^2 + 2\langle \Pi(\mathbf{u} - \mathbf{x}_{NN}), \mathbf{z}\rangle + \|\Pi(\mathbf{u} - \mathbf{x}_{NN})\|_2^2 = \|\mathbf{z} + \Pi(\mathbf{u} - \mathbf{x}_{NN})\|_2^2.$$

Hence, this objection function is encouraging $\mathbf{u}'$ to be as close to $-\Pi(\mathbf{u} - \mathbf{x}_{NN}) = \Pi(\mathbf{x}_{NN} - \mathbf{u})$ as possible subject to satisfying the constraints in Line 2 of Algorithm 4.1. Recalling (c) just above, we can now see that this is exactly encouraging $\mathbf{u}'$ to be a value for which the objective function we seek to minimize in (c) is relatively large.

We are now prepared to empirically compare the four types of embeddings (a) – (d) on the data

sets discussed above in Section 4.5.2. To do so, we run Algorithm 4.2 four times for several different choices of embedding dimension m on each data set below, varying the choice of embedding f between (a), (b), (c), and (d) for each value of m. The successful classification percentage is then plotted as a function of m for each different data set and choice of embedding. See Figures 4.2(a) and 4.2(c) for the results. In addition, to quantify the extent to which the embedding strategies (b) – (d) above are increasingly nonlinear, we also measure the relative distance between where each training-set embedding f maps points in the test sets versus where its associated linear training-set embedding would map them. More specifically, for each embedding f and test point $\mathbf{u} \in S$ we let

$$\text{Nonlinearity}_f(\mathbf{u}) = \frac{\|f(\mathbf{u}) - (\Pi\mathbf{u}, 0)\|2}{\|(\Pi\mathbf{u}, 0)\|2} \times 100\%$$

See Figures 4.2(b) and 4.2(d) for plots of

$$\text{Mean}_{\mathbf{u}\in S}\ \text{Nonlinearity}_f(\mathbf{u})$$

for each of the embedding strategies (b) – (d) on the data sets discussed in Section 4.5.2.

To compute solutions to the minimization problem in Line 2 of Algorithm 4.1 below we used the MATLAB package CVX [41, 40] with the initialization $\mathbf{z}_0 = \Pi(\mathbf{u} - \mathbf{x}_{NN})$ and $\epsilon = 0.1$ in the constraints. All simulations were performed using MATLAB R2021b on an Intel desktop with a 2.60GHz i7-10750H CPU and 16GB DDR4 2933MHz memory. All code used to generate the figures below is publicly available at https://github.com/MarkPhilipRoach/TerminalEmbedding.

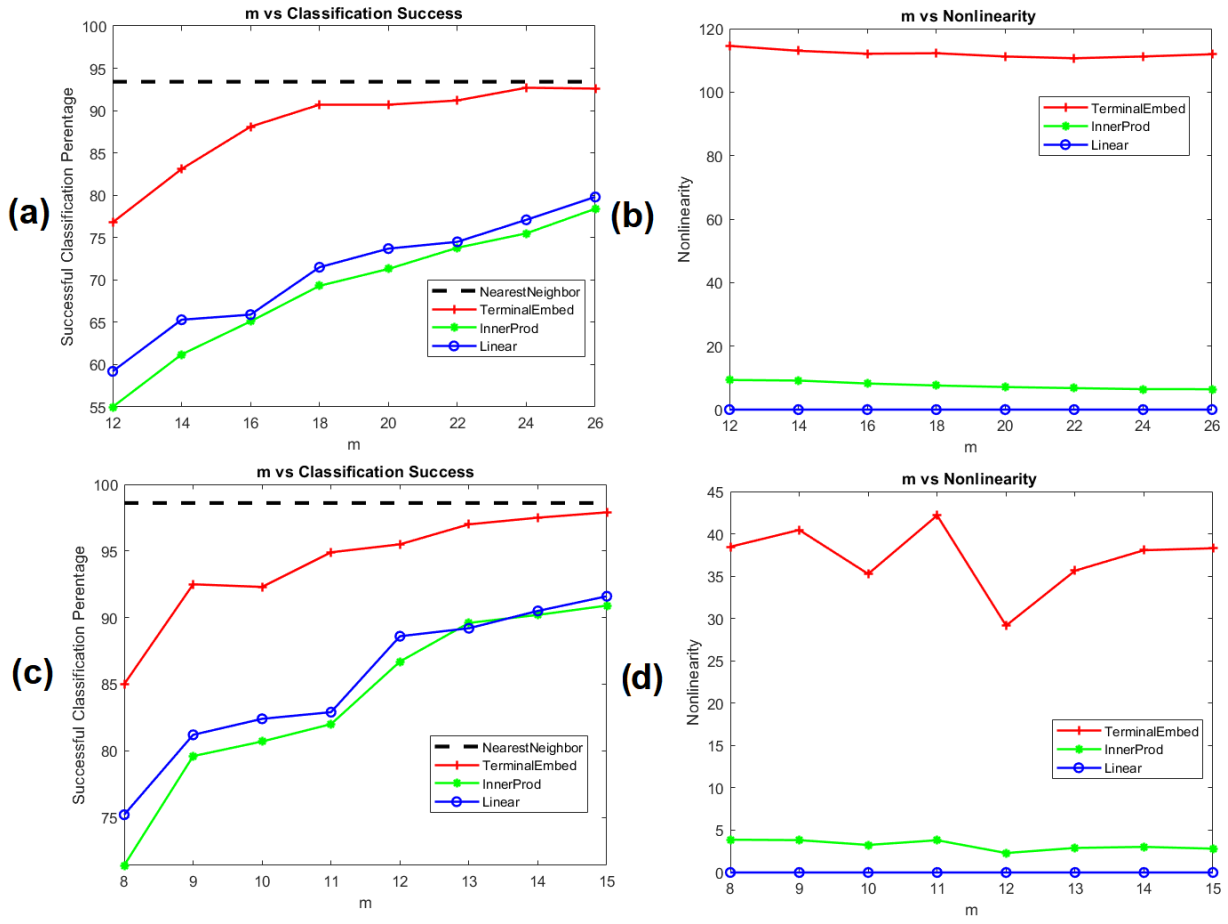

Figure 4.2 Figures 4.2(a) and 4.2(b) concern the MNIST data set with training set size $n = 4000$ and test set size $n' = 1000$ in all experiments. Similarly, Figures 4.2(c) and 4.2(d) concern the COIL-100 data set with training set size $n = 3600$ and test set size $n' = 1000$ in all experiments. In both Figures 4.2(a) and 4.2(c) the dashed black "NearestNeighbor" line plots the classification accuracy when the **Identity** map (a) is used in Algorithm 4.2. Note that the "NearestNeighbor" line is independent of m because the identity map involves no compression. Similarly, in all of the Figures 4.2(a) – 4.2(d) the red "TerminalEmbed" curves correspond to the use of Algorithm 4.1 as it's presented to compute highly non-linear terminal embeddings (embedding strategy (d) above), the green "InnerProd" curves correspond to the use of nearly linear terminal embeddings (embedding strategy (c) above), and the blue "Linear" curves correspond to the use of **Linear** JL embedding matrices (embedding strategy (b) above).

Looking at Figure 4.2 one can see that the most non-linear embedding strategy (d) – i.e., Algorithm 4.1 – allows for the best compressed NN classification performance, outperforming standard linear JL embeddings for all choices of m. Perhaps most interestingly, it also quickly

converges to the uncompressed NN classification performance, matching it to within 1 percent at the values of $m = 24$ for MNIST and $m = 15$ for COIL-100. This corresponds to relative dimensionality reductions of

$$100(1 - 24/784)\% \approx 96.9\%$$

and

$$100(1 - 15/16384)\% \approx 99.9\%,$$

respectively, with negligible loss of NN classification accuracy. As a result, it does indeed appear as if nonlinear terminal embeddings have the potential to allow for improvements in dimensionality reduction in the context of classification beyond what standard linear JL embeddings can achieve.

Of course, challenges remain in the practical application of such nonlinear terminal embeddings. Principally, their computation by, e.g., Algorithm 4.1 is orders of magnitude slower than simply applying a JL embedding matrix to the data one wishes to compressively classify. Nonetheless, if dimension reduction at all costs is one's goal, terminal embeddings appear capable of providing better results than their linear brethren. And, recent theoretical work [18] aimed at lessening their computational deficiencies looks promising.

4.5.4 Additional Experiments on Effective Distortions and Run Times

In this section we further investigate the best performing terminal embedding strategy from the previous section (i.e., Algorithm 4.1) on the MNIST and COIL-100 data sets. In particular, we provide illustrative experiments concerning the improvement of (*i*) compressive classification accuracy with training set size, and (*ii*) the effective distortion of the terminal embedding with embedding dimension $m + 1$. Furthermore, we also investigate (*iii*) the run time scaling of Algorithm 4.1.

To compute the effective distortions of a given (terminal) embedding of training data X, $f : \mathbb{R}^N \to \mathbb{R}^{m+1}$, over all available test and train data $X \cup S$ we use

$$\text{MaxDist}_f = \max_{x \in X} \max_{u \in S \cup X \setminus \{x\}} \frac{\|f(\mathbf{u}) - f(\mathbf{x})\|_2}{\|\mathbf{u} - \mathbf{x}\|_2}, \quad \text{MinDist}_f = \min_{x \in X} \min_{u \in S \cup X \setminus \{x\}} \frac{\|f(\mathbf{u}) - f(\mathbf{x})\|_2}{\|\mathbf{u} - \mathbf{x}\|_2}.$$

Note that these correspond to estimates of the upper and lower multiplicative distortions, respectively, of a given terminal embedding in (4.2). In order to better understand the effect of the minimizer $\mathbf{u}'$ of the minimization problem in Line 2 of Algorithm 4.1 on the final embedding f, we will also separately consider the effective distortions of its component linear JL embedding $\mathbf{u} \mapsto (\Pi\mathbf{u}, 0)$ below. See Figures 4.3 and 4.4 for such plots using the MNIST and COIL-100 data sets, respectively.

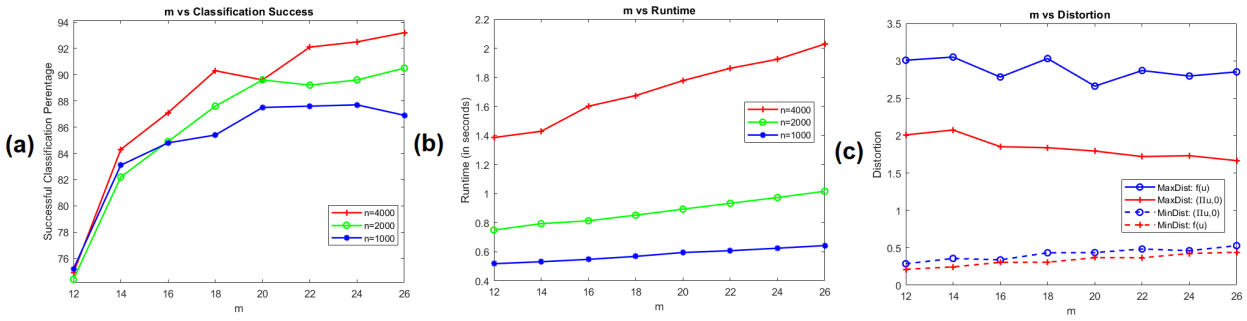

Figure 4.3 This figure compares (a) compressive NN classification accuracies, and (b) the classification run times of Algorithm 4.2 averaged over all $\mathbf{u} \in S$, on the MNIST data set. Three different training data set sizes $n = |X| \in \{1000,\ 2000,\ 4000\}$ were fixed as the embedding dimension $m + 1$ varied for each of the first two subfigures. Recall that the test set size is always fixed to $n' = 1000$. In addition, Figure (c) compares MaxDist_f and MinDist_f for the nonlinear f computed by Algorithm 4.1 versus its component linear embedding $\mathbf{u} \mapsto (\Pi\mathbf{u}, 0)$ as m varies for a fixed embedded training set size of $n = 4000$.

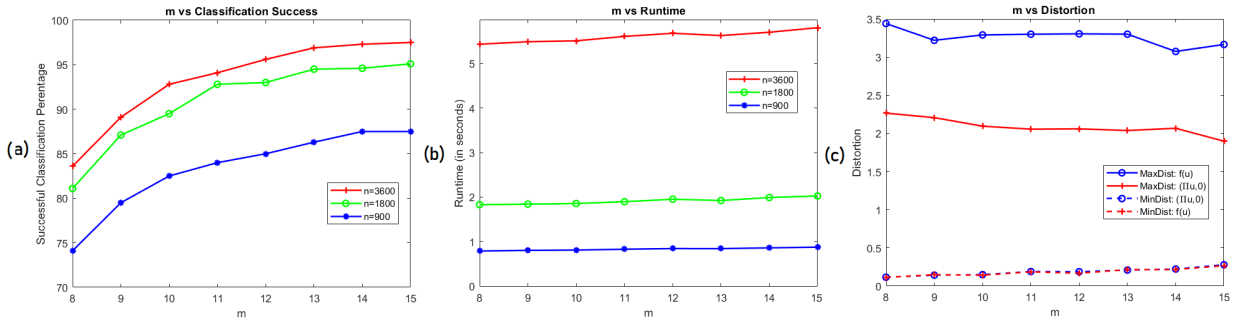

Figure 4.4 Figures (a) and (b) here are run with identical parameters as for their corresponding subfigures in Figure 4.3, except using the COIL-100 data set. Similarly, Figure (c) compares MaxDist_f and MinDist_f for the nonlinear f computed by Algorithm 4.1 versus its component linear embedding $\mathbf{u} \mapsto (\Pi\mathbf{u}, 0)$ as m varies for a fixed embedded training set size of $n = 3600$.

Looking at Figures 4.3 and 4.4 one notes several consistent trends. First, compressive classification accuracy increases with both training set size n and embedding dimension m, as generally expected. Second, compressive classification run times also increase with training set size n (as well as more mildly with embedding dimension m). This is mainly due to the increase in the number of constraints in Line 2 of Algorithm 4.1 with the training set size n. Finally, the distortion plots indicate that the nonlinear terminal embeddings f computed by Algorithm 4.1 tend to preserve the lower distortions of their component linear JL embeddings while simultaneously increasing their upper distortions. As a result, the nonlinear terminal embeddings considered here appear to spread the initially JL embedded data out, perhaps pushing different classes away from one another in the process. If so, it would help explained the increased compressive NN classification accuracy observed for Algorithm 4.1 in Figure 4.2.

4.5.5 Additional Simulation

Figure 4.5 Example images from the Fashion-MNIST data set

The Fashion-MNIST data set [67, 22] consists of 60,0000 training images of 28×28-pixel grayscale images of clothing items, with 10 labels to correctly classify between, and $N = 28^2 = 784$. For all experiments involving the Fashion-MNIST dataset $n/10$ images of each clothing item are selected uniformly at random to form the training set X, for a total of n vectorized training images in $\mathbb{R}^{784}$. Then, 100 images of each type are randomly selected from those not used for training in order to form the test set S, leading to a total of $n' = 1000$ vectorized test images in $\mathbb{R}^{784}$. See Figure 4.5 for example Fashion-MNIST images.

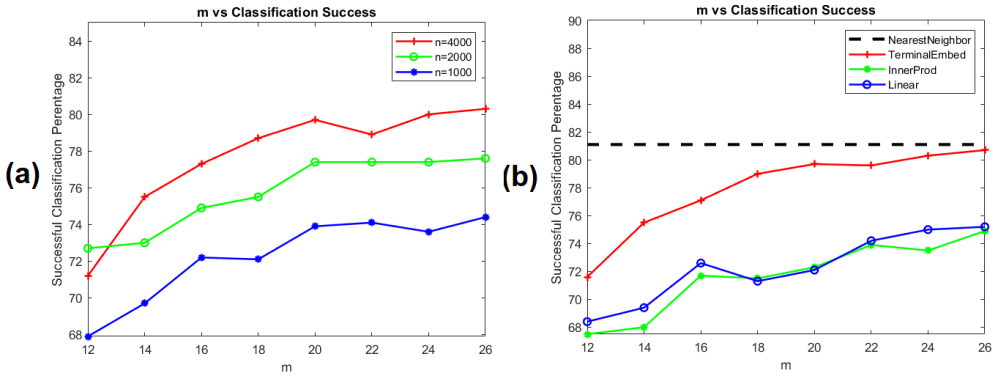

Figure 4.6 Figure 4.6(a) compares compressive NN classification accuracies. Three different training data set sizes $n = |X| \in \{1000,\ 2000,\ 4000\}$ were fixed as the embedding dimension $m + 1$ varied. The test set size is again fixed to $n' = 1000$. Figure 4.6(b) concerns the algorithmic comparison, as discussed in Figure 4.2, with training set size $n = 4000$.

4.6 Compressed Classification from Phaseless Measurements

We now consider applying compressed classification to the measurements generated from chapters 2 and 3. For near-field ptychographic measurements, we use measurements of the form given in 2.2, using $\mathbf{p}$ and $\mathbf{m}$ as given in Lemma 2.3.2. For the far-field ptychographic measurements, we use measurements of the form given in 3.1, using masks given in 2.8. For both of these measurements, we will then vectorize so that we can apply our classification algorithm. For the addition of noise, we apply a Gaussian noise vector to each $\mathbf{u} \in S$ as given in Algorithm 4.2, that is, we replace $\mathbf{u}$ with $\mathbf{u} + \mathbf{n}$, $\mathbf{n} \in \mathbb{R}^d$, before applying the minimization problem from Algorithm 4.1.

For our $\mathbf{x}$, we will use the grayscaled vectorizations of the MNIST and COIL-100 images as performed in the previous section. This however, causes issues with the original space will be for which we embed. For instance, consider a grayscale image that is $P \times P$-pixels. Once vectorized, this is a P^2-vector and thus the matrix of measurements will be $P^2 \times P^2$. Once this has been vectorized, we will finally result in a P^4-vector for our training/testing data. For the MNIST data, this would result in vectors of size $28^4 \approx 600$ thousand, whereas for the COIL-100 data, this would result in vectors of size $128^4 \approx 268$ million. As a consequence, this results in long running times for the algorithm. To counteract this, rather than taking the full measurement matrix, we instead sub-sample based on the frequencies. In particular, we will simply take the first column

for our training/testing data. We will demonstrate that this approach not only allows for successful classification, but it does not impact the result in a significant manner.

Firstly we demonstrate our results on classifying NFP measurements of the MNIST dataset. Since the vectorized images are of length $d = 28^2 = 784$, we choose $delta = 25$ such that d is divisible by $2\delta - 1$.

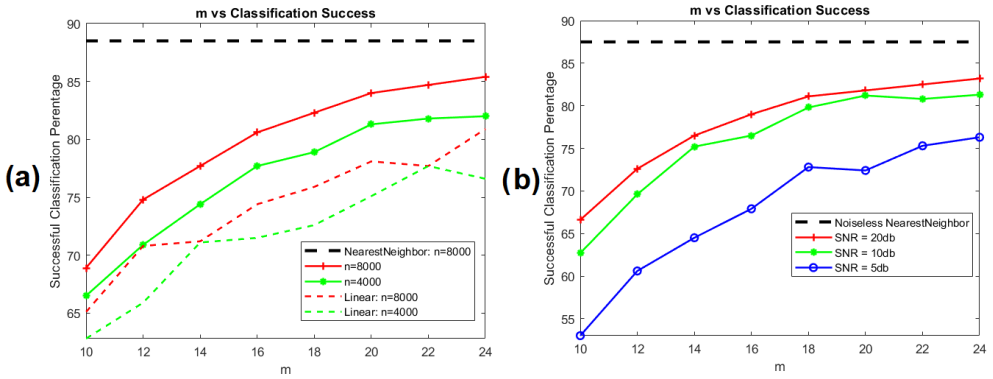

Figure 4.7 Figure 4.7(a) compares compressive NN classification accuracies for the MNIST-NFP measurements ($\delta = 25$). Three different training data set sizes $n = |X| \in \{4000, 8000\}$ were fixed as the embedding dimension $m + 1$. NearestNeighbor refers to the nearest neighbor classification in the original space. Linear refers to the linear embedding described in Section 4.5.3. Figure 4.7(b) concerns the classification in which varying levels of noise are applied, with the training data set size fixed to $n = 8000$. Noiseless NearestNeighbor refers to the noiseless nearest neighbor classification in the original space. For both figures, the test set size is fixed to $n' = 1000$.

Secondly, we demonstrate our results on classifying NFP measurements of the COIL-100 dataset. Here, we encounter an issue in choosing a suitable δ since the vectorized COIL-100 images are of length $d = 128^2 = 16,384 = 2^{14}$, as such no δ exists wherein d is divisible by $2\delta - 1$. Instead, we choose a δ that approximates the ratio of d/δ that we used the MNIST-NFP simulations above. We then artificially extend d using the same process discussed in Section 2.7 to ensure divisibility.

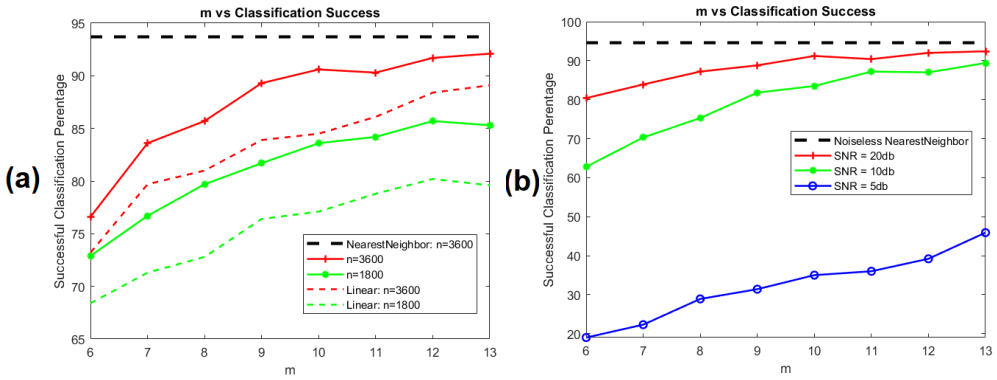

Figure 4.8 Figure 4.8(a) compares compressive NN classification accuracies for the COIL-NFP measurements ($\delta = 525$). Two different training data set sizes $n = |X| \in \{1800, 3600\}$ were fixed as the embedding dimension $m + 1$. NearestNeighbor refers to the nearest neighbor classification in the original space. Linear refers to the linear embedding described in Section 4.5.3. Figure 4.8(b) concerns the classification in which varying levels of noise are applied, with the training data set size fixed to $n = 3600$. Noiseless NearestNeighbor refers to the noiseless nearest neighbor classification in the original space. For both figures, the test set size is fixed to $n' = 1000$.

Finally, we look at the effect of sub-sampling the frequency index, using simply the nearest neighbor classification in the original space.

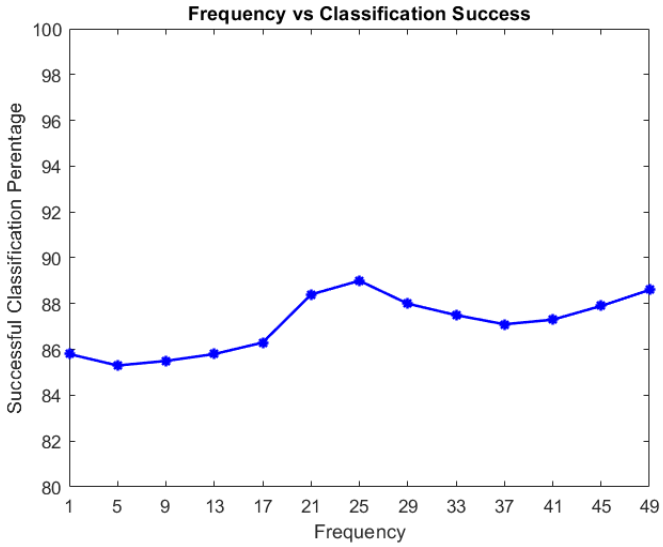

Figure 4.9 Figure representing the non-compressed NN classification of the MNIST-NFP measurements ($\delta = 25$), with varying levels of sub-sampling, that is,
$Y_{k,\ell} = |(\mathbf{p} * (S_k \mathbf{m} \circ \mathbf{x}))_\ell|^2$, $(k, \ell) \in [d]_0 \times [K]_0$, with varying levels for K, up to $2\delta - 1$.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Ali Ahmed, Alireza Aghasi, and Paul Hand. Blind deconvolutional phase retrieval via convex programming. *Advances in Neural Information Processing Systems*, 31, 2018.

[2] Ali Ahmed, Benjamin Recht, and Justin Romberg. Blind deconvolution using convex programming. *IEEE Transactions on Information Theory*, 60(3):1711–1732, 2013.

[3] Jacopo Antonello and Michel Verhaegen. Modal-based phase retrieval for adaptive optics. *JOSA A*, 32(6):1160–1170, 2015.

[4] GR Ayers and J Christopher Dainty. Iterative blind deconvolution method and its applications. *Optics letters*, 13(7):547–549, 1988.

[5] Richard G Baraniuk and Michael B Wakin. Random projections of smooth manifolds. *Foundations of computational mathematics*, 9(1):51–77, 2009.

[6] Robert Beinert and Gerlind Plonka. Ambiguities in one-dimensional discrete phase retrieval from fourier magnitudes. *Journal of Fourier Analysis and Applications*, 21(6):1169–1198, 2015.

[7] Tamir Bendory, Robert Beinert, and Yonina C Eldar. Fourier phase retrieval: Uniqueness and algorithms. In *Compressed Sensing and its Applications*, pages 55–91. Springer, 2017.

[8] JR Bond and G Efstathiou. The statistics of cosmic background radiation fluctuations. *Monthly Notices of the Royal Astronomical Society*, 226(3):655–687, 1987.

[9] Joe Buhler and Zinovy Reichstein. Symmetric functions and the phase problem in crystallography. *Transactions of the American Mathematical Society*, 357(6):2353–2377, 2005.

[10] Oliver Bunk, Martin Dierolf, Søren Kynde, Ian Johnson, Othmar Marti, and Franz Pfeiffer. Influence of the overlap parameter on the convergence of the ptychographical iterative engine. *Ultramicroscopy*, 108(5):481–487, 2008.

[11] Imre Bárány. A generalization of carathéodory's theorem. *Discrete Mathematics*, 40(2):141–152, 1982.

[12] Emmanuel J Candès, Yonina C Eldar, Thomas Strohmer, and Vladislav Voroninski. Phase retrieval via matrix completion. *SIAM review*, 57(2):225–251, 2015.

[13] Emmanuel J. Candès, Xiaodong Li, and Mahdi Soltanolkotabi. Phase retrieval from coded diffraction patterns. *Applied and Computational Harmonic Analysis*, 39(2):277 – 299, 2015.

[14] Emmanuel J Candes, Xiaodong Li, and Mahdi Soltanolkotabi. Phase retrieval via wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, 61(4):1985–2007, 2015.

[15] Alfred S Carasso. Direct blind deconvolution. *SIAM Journal on Applied Mathematics*, 61(6):1980–2007, 2001.

[16] Huibin Chang, Pablo Enfedaque, and Stefano Marchesini. Blind ptychographic phase retrieval via convergent alternating direction method of multipliers. *SIAM Journal on Imaging Sciences*, 12(1):153–185, 2019.

[17] Huibin Chang, Li Yang, and Stefano Marchesini. Fast iterative algorithms for blind phase retrieval: A survey. *arXiv preprint arXiv:2211.06619*, 2022.

[18] Yeshwanth Cherapanamjeri and Jelani Nelson. Terminal embeddings in sublinear time. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1209–1216. IEEE, 2022.

[19] Jesse N Clark, Xiaojing Huang, Ross J Harder, and Ian K Robinson. Continuous scanning mode for ptychography. *Optics letters*, 39(20):6066–6069, 2014.

[20] Albert Cohen, Wolfgang Dahmen, and Ronald DeVore. Compressed sensing and best k-term approximation. *Journal of the American mathematical society*, 22(1):211–231, 2009.

[21] Mark A Davenport, Marco F Duarte, Michael B Wakin, Jason N Laska, Dharmpal Takhar, Kevin F Kelly, and Richard G Baraniuk. The smashed filter for compressive classification and target recognition. In *Computational Imaging V*, volume 6498, page 64980H. International Society for Optics and Photonics, 2007.

[22] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[23] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE international conference on computer vision*, pages 576–584, 2015.

[24] Jan Drenth. *Principles of protein X-ray crystallography*. Springer Science & Business Media, 2007.

[25] George H Dunteman. *Principal components analysis*. Number 69. Sage, 1989.

[26] TB Edo, DJ Batey, AM Maiden, C Rau, U Wagner, ZD Pevsić, TA Waigh, and JM Rodenburg. Sampling in x-ray ptychography. *Physical Review A*, 87(5):053850, 2013.

[27] Armin Eftekhari and Michael B Wakin. New analysis of manifold embeddings and signal

recovery from compressive measurements. *Applied and Computational Harmonic Analysis*, 39(1):67–109, 2015.

[28] Yonina C Eldar, Pavel Sidorenko, Dustin G Mixon, Shaby Barel, and Oren Cohen. Sparse phase retrieval from short-time fourier measurements. *IEEE Signal Processing Letters*, 22(5):638–642, 2014.

[29] Michael Elkin, Arnold Filtser, and Ofer Neiman. Terminal embeddings. *arXiv preprint arXiv:1603.02321*, 2016.

[30] Albert Fannjiang and Pengwen Chen. Blind ptychography: uniqueness and ambiguities. *Inverse Problems*, 36(4):045005, 2020.

[31] Albert Fannjiang and Zheqing Zhang. Fixed point analysis of douglas–rachford splitting for ptychography and phase retrieval. *SIAM Journal on Imaging Sciences*, 13(2):609–650, 2020.

[32] Herbert Federer. Curvature measures. *Transactions of the American Mathematical Society*, 93(3):418–418, March 1959.

[33] Frank Filbir, Felix Krahmer, and Oleh Melnyk. On recovery guarantees for angular synchronization. *Journal of Fourier Analysis and Applications*, 27(2):1–26, 2021.

[34] DA Fish, AM Brinicombe, ER Pike, and JG Walker. Blind deconvolution by means of the richardson–lucy algorithm. *JOSA A*, 12(1):58–65, 1995.

[35] Horacio E Fortunato and Manuel M Oliveira. Fast high-quality non-blind deconvolution using sparse adaptive priors. *The Visual Computer*, 30(6):661–671, 2014.

[36] Grant R Fowles. *Introduction to modern optics*. Courier Corporation, 1989.

[37] Si Gao, Peng Wang, Fucai Zhang, Gerardo T Martinez, Peter D Nellist, Xiaoqing Pan, and Angus I Kirkland. Electron ptychographic microscopy for three-dimensional imaging. *Nature communications*, 8(1):1–8, 2017.

[38] Pierre Godard, Marc Allain, and Virginie Chamard. Imaging of highly inhomogeneous strain field in nanocrystals using x-ray bragg ptychography: A numerical study. *Physical Review B*, 84(14):144109, 2011.

[39] JW Goodman. Film-grain noise in wavefront-reconstruction imaging. *JOSA*, 57(4):493–502, 1967.

[40] Michael Grant and Stephen Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited,

2008. http://stanford.edu/~boyd/graph_dcp.html.

[41] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. http://cvxr.com/cvx, March 2014.

[42] D Griffin, D Deadrick, and Jae Lim. Speech synthesis from short-time fourier transform magnitude and its application to speech processing. In *ICASSP'84. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 9, pages 61–64. IEEE, 1984.

[43] R Hegerl and W Hoppe. Phase evaluation in generalized diffraction (ptychography). *Proc. Fifth Eur. Cong. Electron Microscopy*, pages 628–629, 1972.

[44] Reiner Hegerl and W Hoppe. Dynamische theorie der kristallstrukturanalyse durch elektronenbeugung im inhomogenen primärstrahlwellenfeld. *Berichte der Bunsengesellschaft für physikalische Chemie*, 74(11):1148–1154, 1970.

[45] W Hoppe. Diffraction in inhomogeneous primary wave fields. *Acta Crystallogr. A 25*, pages 495–501,508–515, 1969.

[46] SO Hruszkewycz, Marc Allain, MV Holt, CE Murray, JR Holt, PH Fuoss, and Virginie Chamard. High-resolution three-dimensional structural microscopy by single-angle bragg ptychography. *Nature materials*, 16(2):244–251, 2017.

[47] Xiaojing Huang, Kenneth Lauer, Jesse N Clark, Weihe Xu, Evgeny Nazaretski, Ross Harder, Ian K Robinson, and Yong S Chu. Fly-scan ptychography. *Scientific reports*, 5(1):1–5, 2015.

[48] IBM. https://www.ibm.com/topics/knn.

[49] M. A. Iwen, B. Preskitt, R. Saab, and A. Viswanathan. Phase retrieval from local measurements: improved robustness via eigenvector-based angular synchronization. *Applied and Computational Harmonic Analysis*, 48:415 – 444, 2020.

[50] Mark Iwen, Arman Tavakoli, and Benjamin Schmidt. Lower bounds on the low-distortion embedding dimension of submanifolds of $\mathbb{R}^n$. *arXiv preprint arXiv:2105.13512*, 2021.

[51] Mark A Iwen, Deanna Needell, Elizaveta Rebrova, and Ali Zare. Lower memory oblivious (tensor) subspace embeddings with fewer random bits: modewise methods for least squares. *SIAM Journal on Matrix Analysis and Applications*, 42(1):376–416, 2021.

[52] Mark A. Iwen, Benjamin Schmidt, and Arman Tavakoli. On fast johnson-lindenstrauss embeddings of compact submanifolds of R^N with boundary. *Arxiv*, 2110.04193, 2021.

[53] Mark A. Iwen, Aditya Viswanathan, and Yang Wang. Fast Phase Retrieval from Local Correlation Measurements. *SIAM Journal on Imaging Sciences*, 9(4):1655–1688, 2016.

[54] Kishore Jaganathan, Yonina Eldar, and Babak Hassibi. Phase retrieval with masks using convex optimization. In *2015 IEEE International Symposium on Information Theory (ISIT)*, pages 1655–1659. IEEE, 2015.

[55] Kishore Jaganathan, Yonina C Eldar, and Babak Hassibi. Stft phase retrieval: Uniqueness guarantees and recovery algorithms. *IEEE Journal of selected topics in signal processing*, 10(4):770–781, 2016.

[56] Francis Arthur Jenkins and Harvey Elliott White. Fundamentals of optics. *Indian Journal of Physics*, 25:265–266, 1957.

[57] Yi Jiang, Zhen Chen, Yimo Han, Pratiti Deb, Hui Gao, Saien Xie, Prafull Purohit, Mark W Tate, Jiwoong Park, Sol M Gruner, et al. Electron ptychography of 2d materials to deep sub-ångström resolution. *Nature*, 559(7714):343–349, 2018.

[58] GH John, R Kohavi, and K Pfleger. Machine learning: proceedings of the eleventh international conference. *Irrelevant features and the subset selection problem*, 1994.

[59] William B Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. *Contemporary mathematics*, 26:189–206, 1984.

[60] Aggelos K Katsaggelos and Kuen-Tsair Lay. Maximum likelihood blur identification and image restoration using the em algorithm. *IEEE Transactions on Signal Processing*, 39(3):729–733, 1991.

[61] Samina Khalid, Tehmina Khalil, and Shamila Nasreen. A survey of feature selection and feature extraction techniques in machine learning. In *2014 science and information conference*, pages 372–378. IEEE, 2014.

[62] Mojzesz Kirszbraun. Über die zusammenziehende und lipschitzsche transformationen. *Fundamenta Mathematicae*, 22(1):77–108, 1934.

[63] Deepa Kundur and Dimitrios Hatzinakos. Blind image restoration via recursive filtering using deterministic constraints. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 4, pages 2283–2286. IEEE, 1996.

[64] Deepa Kundur and Dimitrios Hatzinakos. A novel blind deconvolution scheme for image restoration using recursive filtering. *IEEE transactions on signal processing*, 46(2):375–390, 1998.

[65] Marcus Frederick Charles Ladd, Rex Alfred Palmer, and Rex Alfred Palmer. *Structure determination by X-ray crystallography*, volume 233. Springer, 1977.

[66] Kasper Green Larsen and Jelani Nelson. Optimality of the johnson-lindenstrauss lemma. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages

633–638. IEEE, 2017.

[67] Y. LeCun, C. Cortes, and C.J.C. Burges. The mnist database of handwritten digits. 1998.

[68] Anat Levin, Yair Weiss, Fredo Durand, and William T Freeman. Understanding blind deconvolution algorithms. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2354–2367, 2011.

[69] Peng Li, Nicholas W Phillips, Steven Leake, Marc Allain, Felix Hofmann, and Virginie Chamard. Revealing nano-scale lattice distortions in implanted material with 3d bragg ptychography. *Nature communications*, 12(1):1–13, 2021.

[70] Xiaodong Li, Shuyang Ling, Thomas Strohmer, and Ke Wei. Rapid, robust, and reliable blind deconvolution via nonconvex optimization. *Applied and computational harmonic analysis*, 47(3):893–934, 2019.

[71] Aristidis C Likas and Nikolas P Galatsanos. A variational approach for bayesian blind image deconvolution. *IEEE transactions on signal processing*, 52(8):2222–2233, 2004.

[72] Z-C Liu, Rui Xu, and Y-H Dong. Phase retrieval in protein crystallography. *Acta Crystallographica Section A: Foundations of Crystallography*, 68(2):256–265, 2012.

[73] Deepali Lodhia, Daniel Brown, Frank Brueckner, Ludovico Carbone, Paul Fulda, Keiko Kokeyama, and Andreas Freise. Phase effects due to beam misalignment on diffraction gratings. *arXiv preprint arXiv:1303.7016*, 2013.

[74] Sepideh Mahabadi, Konstantin Makarychev, Yury Makarychev, and Ilya Razenshteyn. Nonlinear dimension reduction via outer bi-lipschitz extensions. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1088–1101, 2018.

[75] Andrew Maiden, Daniel Johnson, and Peng Li. Further improvements to the ptychographical iterative engine. *Optica*, 4(7):736–745, 2017.

[76] Andrew M Maiden and John M Rodenburg. An improved ptychographical phase retrieval algorithm for diffractive imaging. *Ultramicroscopy*, 109(10):1256–1262, 2009.

[77] Sami Eid Merhi. *Phase Retrieval from Continuous and Discrete Ptychographic Measurements*. Michigan State University, 2019.

[78] Rafael Molina, Aggelos K Katsaggelos, Javier Abad, and Javier Mateos. A bayesian approach to blind deconvolution based on dirichlet distributions. In *1997 IEEE international conference on acoustics, speech, and signal processing*, volume 4, pages 2809–2812. IEEE, 1997.

[79] Shyam Narayanan and Jelani Nelson. Optimal terminal dimensionality reduction in euclidean

space. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1064–1069, 2019.

[80] Shyam Narayanan and Jelani Nelson. Optimal terminal dimensionality reduction in euclidean space. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1064–1069, 2019.

[81] S Nawab, T Quatieri, and Jae Lim. Signal reconstruction from short-time fourier transform magnitude. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 31(4):986–998, 1983.

[82] Sameer A Nene, Shree K Nayar, and Hiroshi Murase. Columbia object image library (coil-100). 1996.

[83] J v Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.

[84] Michal Odstrvcil, Mirko Holler, and Manuel Guizar-Sicairos. Arbitrary-path fly-scan ptychography. *Optics express*, 26(10):12585–12593, 2018.

[85] Byung Tae Oh, Shaw-min Lei, and C-C Jay Kuo. Advanced film grain noise extraction and synthesis for high-definition video coding. *IEEE transactions on circuits and systems for video technology*, 19(12):1717–1729, 2009.

[86] Olympus. https://www.olympus-lifescience.com/en/microscope-resource/primer/techniques/oblique/obliqueintro/.

[87] Xiaoze Ou, Roarke Horstmeyer, Guoan Zheng, and Changhuei Yang. High numerical aperture fourier ptychography: principle, implementation and characterization. *Optics express*, 23(3):3472–3491, 2015.

[88] Michael Perlmutter, Sami Merhi, Aditya Viswanathan, and Mark Iwen. Inverting spectrogram measurements via aliased wigner distribution deconvolution and angular synchronization. *Information and Inference: A Journal of the IMA*, 2020.

[89] Franz Pfeiffer. X-ray ptychography. *Nature Photonics*, 12(1):9–17, 2018.

[90] M Portnoff. Short-time fourier analysis of sampled speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(3):364–373, 1981.

[91] Brian Preskitt and Rayan Saab. Admissible measurements and robust algorithms for ptychography. *Journal of Fourier Analysis and Applications*, 27(2):1–39, 2021.

[92] Brian P Preskitt. *Phase retrieval from locally supported measurements*. University of California, San Diego, 2018.

[93] Klaus G Puschmann and Franz Kneer. On super-resolution in astronomical imaging. *Astronomy & Astrophysics*, 436(1):373–378, 2005.

[94] Jianliang Qian, Chao Yang, A Schirotzek, F Maia, and S Marchesini. Efficient algorithms for ptychographic phase retrieval. *Inverse Problems and Applications, Contemp. Math*, 615:261–280, 2014.

[95] JM Rodenburg. Ptychography and related diffractive imaging methods. *Advances in Imaging and Electron Physics*, 150:87–184, 2008.

[96] P Rosero-Montalvo, P Diaz, Jose Alejandro Salazar-Castro, DF Pena-Unigarro, Andres J Anaya-Isaza, Juan C Alvarado-Pérez, Roberto Therón, and Diego Hernán Peluffo-Ordóñez. Interactive data visualization using dimensionality reduction and similarity-based representations. In *IberoAmerican Congress on Pattern Recognition*, pages 334–342. Springer, 2017.

[97] P Yu Rotha and David M Paganin. Blind phase retrieval for aberrated linear shift-invariant imaging systems. *New Journal of Physics*, 12(7):073040, 2010.

[98] John W Sammon. A nonlinear mapping for data structure analysis. *IEEE Transactions on computers*, 100(5):401–409, 1969.

[99] Pavel Sidorenko and Oren Cohen. Single-shot ptychography. *Optica*, 3(1):9–14, 2016.

[100] VI Slyusar. A family of face products of matrices and its properties. *Cybernetics and systems analysis*, 35(3):379–384, 1999.

[101] George F Smoot and Douglas Scott. Cosmic background radiation. *The European Physical Journal C-Particles and Fields*, 15:145–149, 2000.

[102] MS Smyth and JHJ Martin. x ray crystallography. *Molecular Pathology*, 53(1):8, 2000.

[103] D.A. Spielman. Spectral and algebraic graph theory. Incomplete Draft, Yale University, 2019.

[104] Filip Sroubek and Jan Flusser. Multichannel blind iterative image restoration. *IEEE Transactions on Image Processing*, 12(9):1094–1106, 2003.

[105] J-L Starck and Fionn Murtagh. Astronomical image and data analysis. 2007.

[106] Marco Stockmar, Peter Cloetens, Irene Zanette, Bjoern Enders, Martin Dierolf, Franz Pfeiffer, and Pierre Thibault. Near-field ptychography: phase retrieval for inline holography using a structured illumination. *Scientific reports*, 3(1):1–6, 2013.

[107] Marco Stockmar, Irene Zanette, Martin Dierolf, Bjoern Enders, Richard Clare, Franz Pfeiffer,

Peter Cloetens, Anne Bonnin, and Pierre Thibault. X-ray near-field ptychography for optically thick specimens. *Physical Review Applied*, 3(1):014005, 2015.

[108] Yukio Takahashi, Akihiro Suzuki, Shin Furutaku, Kazuto Yamauchi, Yoshiki Kohmura, and Tetsuya Ishikawa. Bragg x-ray ptychography of a silicon crystal: Visualization of the dislocation strain field and the production of a vortex beam. *Physical Review B*, 87(12):121201, 2013.

[109] Pierre Thibault, Martin Dierolf, Oliver Bunk, Andreas Menzel, and Franz Pfeiffer. Probe retrieval in ptychographic coherent diffractive imaging. *Ultramicroscopy*, 109(4):338–343, 2009.

[110] Pierre Thibault, Martin Dierolf, Andreas Menzel, Oliver Bunk, Christian David, and Franz Pfeiffer. High-resolution scanning x-ray diffraction microscopy. *Science*, 321(5887):379–382, 2008.

[111] Eric Thiébaut and J-M Conan. Strict a priori constraints for maximum-likelihood blind deconvolution. *JOSA A*, 12(3):485–492, 1995.

[112] Christoph Thäle. 50 years sets with positive reach–a survey. *Surveys in Mathematics and its Applications*, 3:123–165, 2008. Publisher: University Constantin Brancusi.

[113] Lei Tian, Xiao Li, Kannan Ramchandran, and Laura Waller. Multiplexed coded illumination for fourier ptychography with an led array microscope. *Biomedical optics express*, 5(7):2376–2389, 2014.

[114] Esther HR Tsai, Ivan Usov, Ana Diaz, Andreas Menzel, and Manuel Guizar-Sicairos. X-ray ptychography with extended depth of field. *Optics express*, 24(25):29089–29108, 2016.

[115] Robert N Tubbs. Lucky exposures: Diffraction limited astronomical imaging through the atmosphere. *arXiv preprint astro-ph/0311481*, 2003.

[116] Roman Vershynin. *High-dimensional probability: an introduction with applications in data science*. Number 47 in Cambridge series in statistical and probabilistic mathematics. Cambridge University Press, New York, NY, 2018.

[117] Aditya Viswanathan and Mark Iwen. Fast angular synchronization for phase retrieval via incomplete information. In *Wavelets and Sparsity XVI*, volume 9597, page 959718. International Society for Optics and Photonics, 2015.

[118] Adriaan Walther. The question of phase retrieval in optics. *Optica Acta: International Journal of Optics*, 10(1):41–49, 1963.

[119] DP Woody and PL Richards. Spectrum of the cosmic background radiation. *Physical Review Letters*, 42(14):925, 1979.

[120] Michael M Woolfson and Michael Mark Woolfson. *An introduction to X-ray crystallography*. Cambridge University Press, 1997.

[121] Shixiang Wu, Chao Dong, and Yu Qiao. Blind image restoration based on cycle-consistent network. *IEEE Transactions on Multimedia*, 2022.

[122] H Yang, RN Rutte, L Jones, M Simson, R Sagawa, H Ryll, M Huth, TJ Pennycook, MLH Green, H Soltau, et al. Simultaneous atomic-resolution electron ptychography and z-contrast imaging of light and heavy elements in complex nanostructures. *Nature Communications*, 7(1):1–8, 2016.

[123] Yu-Li You and Mostafa Kaveh. Blind image restoration by anisotropic regularization. *IEEE Transactions on Image Processing*, 8(3):396–407, 1999.

[124] He Zhang, Shaowei Jiang, Jun Liao, Junjing Deng, Jian Liu, Yongbing Zhang, and Guoan Zheng. Near-field fourier ptychography: super-resolution phase retrieval via speckle illumination. *Optics express*, 27(5):7498–7512, 2019.

[125] Guoan Zheng. *Fourier ptychographic imaging: a MATLAB tutorial*. Morgan & Claypool Publishers, 2016.

[126] Guoan Zheng, Roarke Horstmeyer, and Changhuei Yang. Wide-field, high-resolution fourier ptychographic microscopy. *Nature photonics*, 7(9):739–745, 2013.

[127] Guoan Zheng, Cheng Shen, Shaowei Jiang, Pengming Song, and Changhuei Yang. Concept, implementations and applications of fourier ptychography. *Nature Reviews Physics*, 3(3):207–223, 2021.

APPENDIX A

NEAR-FIELD PTYCHOGRAPHY

A.1 Technical Lemmas

We state the following lemmas for the sake of completeness. Note that we index all vectors modulo d.

Lemma A.1.1. *Let* $\mathbf{x}, \mathbf{y} \in \mathbb{C}^d$. *We have that* $|\langle \mathbf{x}, \overline{\mathbf{y}} \rangle|^2 = |\langle \overline{\mathbf{x}}, \mathbf{y} \rangle|^2$.

Lemma A.1.2. *Let* $\mathbf{x}, \mathbf{y} \in \mathbb{C}^d$. *We have that* $\langle \mathbf{x}, \mathbf{y} \circ \mathbf{z} \rangle = \langle \mathbf{x} \circ \overline{\mathbf{y}}, \mathbf{z} \rangle$.

Proof. By the definition of the inner product and the Hadamard product

$$\langle \mathbf{x}, \mathbf{y} \circ \mathbf{z} \rangle = \sum_{n=0}^{d-1} x_n \overline{(\mathbf{y} \circ \mathbf{z})_n} = \sum_{n=0}^{d-1} x_n \overline{y}_n \overline{z}_n = \sum_{n=0}^{d-1} (\mathbf{x} \circ \overline{\mathbf{y}})_n \overline{z}_n = \langle \mathbf{x} \circ \overline{\mathbf{y}}, \mathbf{z} \rangle.$$

□

Lemma A.1.3. *Let* $\mathbf{x}, \mathbf{y} \in \mathbb{C}^d, k \in \mathbb{Z}$. *We have that*

1. $(\mathbf{x} * \mathbf{y})_k = \langle S_{-k} \widetilde{\mathbf{x}}, \overline{\mathbf{y}} \rangle$;
2. $\mathbf{x} \circ S_k \mathbf{y} = S_k (S_{-k} \mathbf{x} \circ \mathbf{y})$;
3. $\langle S_k \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, S_{-k} \mathbf{y} \rangle$.

Proof. Proof of 1: Let $\mathbf{x}, \mathbf{y} \in \mathbb{C}^d$. By the definition of the circular convolution

$$(\mathbf{x} * \mathbf{y})_k = \sum_{n=0}^{d-1} x_{k-n} y_n = \sum_{n=0}^{d-1} x_{k-n} \overline{\overline{y_n}} = \sum_{n=0}^{d-1} (S_{-k} \widetilde{\mathbf{x}})_n \overline{\overline{y_n}} = \langle S_{-k} \widetilde{\mathbf{x}}, \overline{\mathbf{y}} \rangle.$$

Proof of 2: Let $\mathbf{x}, \mathbf{y} \in \mathbb{C}^d, k \in \mathbb{Z}$. Let $n \in [d]_0$ be arbitrary. Then we have that

$$(\mathbf{x} \circ S_k \mathbf{y})_n = \mathbf{x}_n (S_k \mathbf{y})_n = (S_{-k} \mathbf{x})_{n+k} \mathbf{y}_{n+k} = (S_k (S_{-k} \mathbf{x} \circ \mathbf{y}))_n.$$

Proof of 3: Noting that we index modulo d, we have that

$$\langle S_k \mathbf{x}, \mathbf{y} \rangle = \sum_{n=0}^{d-1} (S_k \mathbf{x})_n \overline{y_n} = \sum_{n=0}^{d-1} x_{n+k} \overline{y_n} = \sum_{n=k}^{d+k-1} x_n \overline{y_{n-k}} = \sum_{n=0}^{d-1} \mathbf{x}_n \overline{y_{n-k}} = \langle \mathbf{x}, S_{-k} \mathbf{y} \rangle.$$

□

We now give our proof of Lemma 2.4.1.

Proof of Lemma 2.4.1. Fix $\phi \in [0, 2\pi)$. By the triangle inequality we have

$$\begin{aligned} \|\mathbf{x} - e^{\mathbb{i}\phi} \mathbf{x}_{\text{est}}\|_2 &= \|\mathbf{x}^{(\text{mag})} \circ \mathbf{x}^{(\theta)} - \mathbf{x}_{\text{est}}^{(\text{mag})} \circ e^{i\phi} \mathbf{x}_{\text{est}}^{(\theta)}\|_2 \\ &\leq \|\mathbf{x}^{(\text{mag})} \circ \mathbf{x}^{(\theta)} - \mathbf{x}^{(\text{mag})} \circ e^{\mathbb{i}\phi} \mathbf{x}_{\text{est}}^{(\theta)}\|_2 \\ &\quad + \|\mathbf{x}^{(\text{mag})} \circ e^{\mathbb{i}\phi} \mathbf{x}_{\text{est}}^{(\theta)} - \mathbf{x}_{\text{est}}^{(\text{mag})} \circ e^{i\phi} \mathbf{x}_{\text{est}}^{(\theta)}\|_2. \end{aligned} \tag{A.1}$$

For the first term, we may use the inequality $\|\mathbf{u} \circ \mathbf{v}\|_2 \leq \|\mathbf{u}\|_\infty \|\mathbf{v}\|_2$ to see that

$$\begin{aligned} \|\mathbf{x}^{(\text{mag})} \circ \mathbf{x}^{(\theta)} - \mathbf{x}^{(\text{mag})} \circ e^{\mathbb{i}\phi} \mathbf{x}_{\text{est}}^{(\theta)}\|_2 &\leq \|\mathbf{x}^{(\text{mag})}\|_\infty \|\mathbf{x}_{\text{est}}^{(\theta)} - e^{-\mathbb{i}\phi} \mathbf{x}^{(\theta)}\|_2 \\ &= \|\mathbf{x}\|_\infty \|\mathbf{x}_{\text{est}}^{(\theta)} - e^{-\mathbb{i}\phi} \mathbf{x}^{(\theta)}\|_2. \end{aligned} \tag{A.2}$$

For the second term, we see that

$$\begin{aligned} \|\mathbf{x}^{(\text{mag})} \circ e^{\mathbb{i}\phi} \mathbf{x}_{\text{est}}^{(\theta)} - \mathbf{x}_{\text{est}}^{(\text{mag})} \circ e^{i\phi} \mathbf{x}_{\text{est}}^{(\theta)}\|_2 &\leq \|e^{\mathbb{i}\phi} \mathbf{x}_{\text{est}}^{(\theta)}\|_\infty \cdot \|\mathbf{x}^{(\text{mag})} - \mathbf{x}_{\text{est}}^{(\text{mag})}\|_2 \\ &= \|\mathbf{x}^{(\text{mag})} - \mathbf{x}_{\text{est}}^{(\text{mag})}\|_2. \end{aligned} \tag{A.3}$$

Combining (A.2) and (A.3) with (A.1) and minimizing over ϕ completes the proof. □

A.2 Auxilliary Results from Spectral Graph Theory

In this section, we will prove several lemmas related to the graph Laplacian and its eigenvalues. The following definition defines a partial ordering on the set of weighted graphs induced by the

spectrum of their graph Laplacians.

Definition A.2.1. *We say that a symmetric matrix* $\mathbf{A}$ *is positive semi-definite and write* $\mathbf{A} \succeq \mathbf{0}$ *if* $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0, \forall \mathbf{x} \in \mathbb{R}^n$ *(or equivalently if all the eigenvalues of* $\mathbf{A}$ *are non-negative). We define the Loewner order*[1] $\succeq$ *by the rule that* $\mathbf{A} \succeq \mathbf{B}$ *if* $\mathbf{A} - \mathbf{B}$ *is positive semi-definite (or equivalently if* $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq \mathbf{x}^T\mathbf{B}\mathbf{x}, \forall \mathbf{x} \in \mathbb{R}^n$*). For two graphs* G *and* H *with the same number of vertices, we will define* $G \succeq H$ *if* $\mathbf{L}_G \succeq \mathbf{L}_H$*. We will also write* $G \succeq \sum_{i=0}^{n-1} H_i$ *if* $\mathbf{L}_G \succeq \sum_{i=0}^{n-1} \mathbf{L}_{H_i}$*, and for a scalar* c *we will write* $G \succeq cH$ *if* $\mathbf{L}_G \succeq c\mathbf{L}_H$*.*

Remark A.2.1. *If* $G \succeq H$ *and* τ_G *and* τ_H *are the smallest non-zero eigenvalues* $\mathbf{L}_G$ *and* $\mathbf{L}_H$*, then one can use the fact that* $\tau_G = \min_{\substack{x \in R^n \\ x \perp 1}} \frac{\mathbf{x}^T\mathbf{L}_G\mathbf{x}}{\mathbf{x}^T\mathbf{x}}$ *(see [103]) to verify that* $\tau_G \geq \tau_H$*.*

We now define some basic terminology for weighted graphs. (We note that these definitions may also be applied to unweighted graphs by interpreting each edge as having weight one.)

Definition A.2.2. ***(Weighted Distance Definitions)*** *Let* $G = (V, E, \mathbf{W})$ *be a weighted graph.*

(i) For any subgraph $H = (V', E')$ *of* G*, we define the* ***weight*** *of* H*, denoted* $w(H)$*, as*

$$w(H) \coloneqq \sum_{(i,j) \in E'} W_{i,j},$$

(ii) If P *is a path inside* G*, we will let* $len(P) \coloneqq w(P)$ *denote the* ***weighted length*** *of* P*.*

(iii) We define the ***weighted distance*** *between two vertices* u *and* v*,* $dist_G(u, v)$*, to be the minimal weighted length of any path from* u *to* v

(iv) The ***weighted diameter*** *of* G*, denoted by* $diam(G)$*, is the maximum distance between any two vertices in* G*, that is,*

$$diam(G) \coloneqq \max\{dist_G(u, v) \mid (u, v) \in V \times V\}.$$

In some contexts, it will be useful to consider the pointwise inverses of the weights $W_{i,j}$.

[1] *The Loewner order is actually a partial ordering since there exist* $\mathbf{A}$ *and* $\mathbf{B}$ *such that* $\mathbf{A} \not\succeq \mathbf{B}$ *and* $\mathbf{B} \not\succeq \mathbf{A}$*.*

Definition A.2.3. ***(Inverse Weighted Distance Definitions)*** *Let $G = (V, E, \mathbf{W})$ be a weighted graph.*
(i) For any subgraph $H = (V', E')$ of G, the ***inverse weight*** *of H, is defined by*

$$w^{-1}(H) := \sum_{(i,j)\in E'} \frac{1}{W_{i,j}},$$

(ii) For a path P inside G, we refer to $len^{-1}(P) := w^{-1}(P)$ as the ***inverted weighted length*** *of P.*
(iii) For two vertices u and v we will refer to the minimal value of $w^{-1}(P)$ over all paths from u to v as the ***inverted weighted distance****, denoted by $dist_G^{-1}(u, v)$.*
(iv) The ***inverse weighted diameter*** *of G, denoted by $diam^{-1}(G)$, is the maximum distance between any two vertices in G, that is,*

$$diam^{-1}(G) := \max\{dist_G^{-1}(u, v) \mid (u, v) \in V \times V\}.$$

The proof of Lemma 2.4.4 (and thus Theorem 2.1.1), relies on the following lemma to provide a lower bound for the spectral gap τ_G.

Lemma A.2.1. ***(Weighted Spectral Bound)*** *Let $G = (V, E, \mathbf{W})$ be a weighted, connected graph with $|V| = n$, and let $W_{\min}$ and W_{max} denote the minimum and maximum value of any of the (nonzero) weights of G. Then*

$$\tau_G \geq \frac{2 \cdot W_{\min}}{W_{max}(n-1) \cdot diam^{-1}(G)}.$$

To prove Lemma A.2.1, we recall the following lemma from [103].

Lemma A.2.2. ***(Weighted Path Inequality)*** *(Lemma 5.6.1 [103]) Let $P_n = (v_0, v_1, \ldots, v_{n-1})$ be a path of length n and assume that, for all $0 \leq i < n-2$, w_i, the weight of (v_i, v_{i+1}) is strictly positive. For $0 \leq i < n-2$, let $G_{i,i+1} = (V, (v_i, v_{i+1}))$ be the graph whose vertex set V is that same as the vertex set of G but only has a single edge (v_i, v_{i+1}). Similarly, let $G_{0,n-1} = (V, (v_0, v_{n-1}))$ be the*

graph with only a single edge (v_0, v_{n-1}). *Then*

$$G_{0,n-1} \preccurlyeq \left(\sum_{i=0}^{n-2} \frac{1}{w_i}\right) \sum_{i=0}^{n-2} w_i G_{i,i+1} \text{ [2]} = len^{-1}(P_n) \cdot P_n,$$

where the final equality is interpreted is the sense of $A \preccurlyeq B$ *and* $B \preccurlyeq A$.

The Proof of Lemma A.2.1. For $u, v \in V$, let $G_{u,v} = (V, (u, v))$ denote the graph with only a single edge from u to v and let $P_{u,v}$ denote a path from u to v with minimal weighted inverse length. Then, by Lemma A.2.2 we have

$$G_{u,v} \preccurlyeq \text{len}^{-1}(P_{u,v}(G)) \cdot P_{u,v}(G) \preccurlyeq \text{diam}^{-1}(G) \cdot P_{u,v}(G) \preccurlyeq \text{diam}^{-1}(G) \cdot G,$$

where the last inequality holds since for all subgraphs H of a graph G, $H \preccurlyeq G$ (Section 5.2 [103])

Let $\widetilde{K}_n$ be the extended weighted, complete graph on n vertices with weighted matrix $\widetilde{\mathbf{W}}$, where $\widetilde{W}_{i,j} = \begin{cases} W_{i,j}, & (i,j) \in E \\ W_{\min}, & (i,j) \notin E \end{cases}$. Then by summing over all vertices, we have that

$$\mathbf{L}_{\widetilde{K}_n} = \sum_{0 \le i < j \le n-1} \widetilde{W}_{i,j} \mathbf{L}_{G_{i,j}} \preccurlyeq W_{\max} \sum_{0 \le i < j \le n-1} \text{diam}^{-1}(G) \cdot \mathbf{L}_G,$$

Since $\sum_{0 \le i < j \le n-1} 1 = n(n-1)$, we then have that

$$\widetilde{K}_n \preccurlyeq \frac{W_{\max} n(n-1)}{2} \text{diam}^{-1}(G) \cdot G,$$

which, by Remark (A.2.1) implies

$$\tau_{\widetilde{K}_n} \le \frac{W_{\max} n(n-1)}{2} \text{diam}^{-1}(G) \tau_G,$$

[2] *Under this construction, we see that if we have a weight which is much larger than all of the others, it effectively gets nullified by taking the inverse.*

and therefore

$$\tau_G \geq \frac{2\tau_{\widetilde{K}_n}}{W_{\max} n(n-1) \cdot \operatorname{diam}^{-1}(G)}.$$

Letting K_n be the unweighted graph on n vertices we see that

$$\mathbf{x}^T \mathbf{L}_{\widetilde{K}_n} \mathbf{x} = \sum_{(a,b)\in[n]_0} \widetilde{W}_{a,b}(x(a)-x(b))^2 \geq W_{\min} \sum_{(a,b)\in[n]_0} (x(a)-x(b))^2 = W_{\min} \mathbf{x}^T \mathbf{L}_{K_n} \mathbf{x}.$$

Therefore, $\tau_{\widetilde{K}_n} = \underset{\substack{x\in R^n \\ x\perp 1}}{min} \frac{\mathbf{x}^T \mathbf{L}_{\widetilde{K}_n} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \geq W_{\min} \underset{\substack{x\in R^n \\ x\perp 1}}{min} \frac{\mathbf{x}^T \mathbf{L}_{K_n} \mathbf{x}}{\mathbf{x}^T \mathbf{x}} \geq W_{\min} \cdot \tau_{K_n}$. Thus, since $\tau_{K_N} = n$ (5.4.1, [103]), we have that $\tau_G \geq \frac{2W_{\min}}{W_{\max}(n-1) \cdot \operatorname{diam}^{-1}(G)}$. □

Our next result uses Lemma A.2.1 to produce a bound for τ_G in terms of the diameter of the underlying unweighted graph.

Theorem A.2.1. *Let $G = (V, E, \mathbf{W})$ be a weighted graph and let $W_{\min}$ and W_{max} be the minimum and maximum value of any its (nonzero) weights. Then*

$$\tau_G \geq \frac{2 \cdot (W_{min})^2}{W_{max}(n-1) diam(G_{unw})},$$

where $G_{unw} = (V, E)$ is the unweighted counterpart of G.

Proof. Let $G' = (V, E, \mathbf{W}')$, where $W'_{i,j} = 1/W_{i,j}$ if $W_{i,j} \neq 0$ and $W'_{i,j} = 0$ otherwise. Let $W'_{\max}$ be the maximum element of $\mathbf{W}'$. Observe that by construction, we have $W'_{\max} = \frac{1}{W_{\min}}$. Moreover, it follows immediately from Definition A.2.2 that we have $\operatorname{diam}^{-1}(G) = \operatorname{diam}(G')$. Therefore,

$$\operatorname{diam}^{-1}(G) = \operatorname{diam}(G') \leq W'_{\max} \cdot \operatorname{diam}(G_{\text{unw}}) = \frac{1}{W_{\min}} \operatorname{diam}(G_{\text{unw}}).$$

So by Lemma A.2.1, we have that

$$\tau_G \geq \frac{2 \cdot W_{\min}}{W_{\max}(n-1) \cdot \operatorname{diam}^{-1}(G)} \geq \frac{2 \cdot (W_{\min})^2}{W_{\max}(n-1) \operatorname{diam}(G_{\text{unw}})}.$$

□

APPENDIX B

FAR-FIELD PTYCHOGRAPHY

B.1 Sub-Sampling

In this section, we discuss sub-sampling lemmas that can be used in conjunction with Algorithm 3.1. In many cases, an illumination of the sample can cause damage to the sample, and applying the illumination beam (which can be highly irradiative) repeatedly at a single point can destroy it. Considering the risks to the sample and the costs of operating the measurement equipment, there are strong incentives to reduce the number of illuminations applied to any object.

Definition B.1.1. *Let $s \in \mathbb{N}$ such that $s \mid d$. We define the **sub-sampling operator** $Z_s : \mathbb{C}^d \longrightarrow \mathbb{C}^{\frac{d}{s}}$ defined component-wise via*

$$(Z_s \mathbf{x})_n := x_{n \cdot s}, \quad \forall n \in [d/s]_0 \tag{B.1}$$

We now have an aliasing lemma which allows us to see the impact of performing the Fourier transform on a sub-sampled specimen.

Lemma B.1.1. ***(Aliasing)*** *([77], Lemma 2.0.1.) Let $s \in \mathbb{N}$ such that $s \mid d$, $\mathbf{x} \in \mathbb{C}^d, \omega \in [\frac{d}{s}]_0$. Then we have that*

$$\left(F_{\frac{d}{s}}(Z_s \mathbf{x})\right)_\omega = \frac{1}{s} \sum_{r=0}^{s-1} \hat{\mathbf{x}}_{\omega - r\frac{d}{s}} \tag{B.2}$$

Proof. Let $d \in \mathbb{N}$ and suppose $s \in \mathbb{N}$ divides d. Let $\mathbf{x} \in \mathbb{C}^d$ and $\omega \in [\dfrac{d}{s}]_0$ be arbitrary. By the definition of the discrete Fourier transform and sub-sampling operator, we have that

$$\left(F_{\frac{d}{s}}(Z_s \mathbf{x})\right)_\omega = \sum_{n=0}^{\frac{d}{s}-1} (Z_s \mathbf{x})_n e^{-\frac{2\pi i n \omega}{d/s}} = \sum_{n=0}^{\frac{d}{s}-1} x_{ns} e^{-\frac{2\pi i n \omega s}{d}} \tag{B.3}$$

By the inverse DFT and by collecting terms, we have that

$$\sum_{n=0}^{\frac{d}{s}-1} x_{ns} e^{-\frac{2\pi i n\omega s}{d}} = \frac{1}{d} \sum_{n=0}^{\frac{d}{s}-1} \left(\sum_{r=0}^{d-1} \hat{x}_r e^{\frac{2\pi i r n s}{d}} \right) e^{-\frac{2\pi i n (r-\omega) s}{d}} \tag{B.4}$$

By treating this as a sum of DFTs, we then have that

$$\frac{1}{d} \sum_{n=0}^{\frac{d}{s}-1} \left(\sum_{r=0}^{d-1} \hat{x}_r e^{\frac{2\pi i r n s}{d}} \right) e^{-\frac{2\pi i n (r-\omega) s}{d}} = \frac{1}{s} \sum_{r=0}^{s-1} \hat{x}_{\omega + r\frac{d}{s}} = \frac{1}{s} \sum_{r=0}^{s-1} \hat{x}_{\omega - r\frac{d}{s}} \tag{B.5}$$

□

Before we start looking at aliased WDD, we need to introduce a lemma which will show the effect of taking a Fourier transform of an autocorrelation.

Lemma B.1.2. ***(Fourier Transform Of Autocorrelation)*** *([77], Lemma 2.0.2.) Let* $\mathbf{x} \in \mathbb{C}^d$ *and* $\alpha, \omega \in [d]_0$. *Then*

$$\left(\mathbf{F}_d(\mathbf{x} \circ S_\omega \bar{\mathbf{x}}) \right)_\alpha = \frac{1}{d} e^{2\pi i \omega \alpha / d} \left(\mathbf{F}_d(\hat{\mathbf{x}} \circ S_{-\alpha} \bar{\hat{\mathbf{x}}}) \right)_\omega \tag{B.6}$$

Proof. Let $\mathbf{x} \in \mathbb{C}^d$ and let $\alpha, \omega \in [d]_0$ be arbitrary. By the convolution theorem, we have that

$$\left(\mathbf{F}_d(\mathbf{x} \circ S_\omega \bar{\mathbf{x}}) \right)_\alpha = \frac{1}{d} (\hat{\mathbf{x}} *_d \mathbf{F}_d(S_\omega \bar{\mathbf{x}}))_\alpha \tag{B.7}$$

By technical equality (iii), we can revert the Fourier transform of the shift operator to the modulation operator of the Fourier transform

$$\frac{1}{d} (\hat{\mathbf{x}} *_d \mathbf{F}_d(S_\omega \bar{\mathbf{x}}))_\alpha = \frac{1}{d} (\hat{\mathbf{x}} *_d (W_\omega \hat{\bar{\mathbf{x}}})_\alpha \quad = \frac{1}{d} \sum_{n=0}^{d-1} \hat{x}_n (W_\omega \hat{\bar{\mathbf{x}}})_{\alpha - n} = \frac{1}{d} \sum_{n=0}^{d-1} \hat{x}_n \hat{\bar{x}}_{\alpha - n} e^{\frac{2\pi i \omega (\alpha - n)}{d}} \tag{B.8}$$

with the latter equalities being the definition of the convolution and modulation. By applying

reversals and using that $\tilde{\tilde{\mathbf{x}}} = \mathbf{x}$, we have that

$$\frac{1}{d}\sum_{n=0}^{d-1}\hat{x}_n\hat{\tilde{x}}_{\alpha-n}e^{\frac{2\pi i\omega(\alpha-n)}{d}} = \frac{1}{d}e^{\frac{2\pi i\omega\alpha}{d}}\sum_{n=0}^{d-1}\hat{x}_n\tilde{\hat{x}}_{n-\alpha}e^{-\frac{2\pi i\omega\alpha}{d}} \tag{B.9}$$

Finally by applying technical equality (vi) and using the definition of the shift operator and Hadamard product, we have that

$$\frac{1}{d}e^{\frac{2\pi i\omega\alpha}{d}}\sum_{n=0}^{d-1}\hat{x}_n\tilde{\hat{x}}_{n-\alpha}e^{-\frac{2\pi i\omega\alpha}{d}} = \frac{1}{d}e^{\frac{2\pi i\omega\alpha}{d}}\sum_{n=0}^{d-1}\hat{x}_n\bar{\hat{x}}_{n-\alpha}e^{-\frac{2\pi i\omega\alpha}{d}} = \frac{1}{d}e^{2\pi i\omega\alpha/d}\left(\mathbf{F}_d(\hat{\mathbf{x}}\circ S_{-\alpha}\bar{\hat{\mathbf{x}}})\right)_\omega \tag{B.10}$$

□

B.1.1 Sub-Sampling In Frequency

We will first look at sub-sampling in frequency.

Definition B.1.2. *Let K be a positive factor of d, and assume that the data is measured at K equally spaced Fourier modes. We denote the set of Fourier modes of step-size $\frac{d}{k}$ by*

$$\mathcal{K} = \frac{d}{K}[K]_0 = \left\{0, \frac{d}{K}, \frac{2d}{K}, \ldots, d - \frac{d}{K}\right\} \tag{B.11}$$

Definition B.1.3. *Let $\mathbf{A} \in \mathbb{C}^{d\times d}$ with columns $\mathbf{a}_j$, $K \mid d$. We denote by $\mathbf{A}_{K,d} \in \mathbb{C}^{K\times d}$ the sub-matrix of $\mathbf{A}$ whose ℓ^{th} column is equal to $Z_{\frac{d}{K}}(\mathbf{a}_\ell)$.*

With these definitions, we will now convert the sub-sampled measurements into a more solvable form.

Lemma B.1.3. *([77], Lemma 2.1.1.) Suppose that the noisy spectrogram measurements are collected on a subset $\mathcal{K} \subseteq [d]_0$ of K equally space Fourier modes. Then for any $\omega \in [K]_0$*

$$\left((\mathbf{F}_K\mathbf{Y}_{K,d})^T\right)_\omega = K\sum_{r=0}^{\frac{d}{K}-1}(\mathbf{x}\circ S_{\omega-rK}\bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}}\circ S_{rK-\omega}\bar{\tilde{\mathbf{m}}}) + \left((\mathbf{F}_K\mathbf{N}_{K,d})^T\right)_\omega$$

where $\mathbf{Y}_{K,d} \in \mathbb{C}^{K\times d} - \mathbf{N}_{K,d} \in \mathbb{C}^{K\times d}$ is the matrix of sub-sampled noiseless $K \cdot d$ measurements.

Proof. For $\ell \in [d]_0$, the ℓ^{th} column of the matrix $\mathbf{Y}$ is

$$\mathbf{y}_\ell = \mathbf{F}_d\Big((\mathbf{x} \circ S_{-\ell}\mathbf{m}) *_d (\bar{\tilde{\mathbf{x}}} \circ S_\ell \bar{\tilde{\mathbf{m}}})\Big) + \eta_\ell \tag{B.12}$$

and thus for any $\alpha \in [K]_0$

$$\Big(Z_{\frac{d}{K}}(\mathbf{y}_\ell)\Big)_\alpha = \Big(\mathbf{F}_d\Big((\mathbf{x} \circ S_{-\ell}\mathbf{m}) *_d (\bar{\tilde{\mathbf{x}}} \circ S_\ell \bar{\tilde{\mathbf{m}}})\Big)\Big)_{\alpha\frac{d}{K}} + \Big(Z_{\frac{d}{K}}(\mathbf{j}_\ell)\Big)_\alpha \tag{B.13}$$

and by aliasing lemma (with $s = \frac{d}{K}$)

$$\Big(\mathbf{F}_K\Big(Z_{\frac{d}{K}}(\mathbf{y}_\ell)\Big)_\alpha\Big)_\omega = \frac{K}{d}\sum_{r=0}^{\frac{d}{K}-1}(\hat{\mathbf{y}}_\ell)_{\omega-rK} = d \cdot \frac{K}{d}\sum_{r=0}^{\frac{d}{K}-1}\Big((\mathbf{x} \circ S_{\omega-rK}\bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}} \circ S_{rK-\omega}\bar{\tilde{\mathbf{m}}})\Big)_\ell + \Big(\mathbf{F}_K\Big(Z_{\frac{d}{K}}(\mathbf{j}_\ell)\Big)_\alpha\Big)_\omega \tag{B.14}$$

The ℓ^{th} column of $\mathbf{Y}_{K,d} \in \mathbb{C}^{K\times d}$ is equal to $Z_{\frac{d}{K}}(\mathbf{y}_\ell)$. Then for any $\omega \in [K]_0$, the ω^{th} column of $(\mathbf{F}_K Y_{K,d})^T \in \mathbb{C}^{d\times K}$ may be computed as

$$\Big((\mathbf{F}_K\mathbf{Y}_{K,d})^T\Big)_\omega = K\sum_{r=0}^{\frac{d}{K}-1}(\mathbf{x} \circ S_{\omega-rK}\bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}} \circ S_{rK-\omega}\bar{\tilde{\mathbf{m}}}) + \Big((\mathbf{F}_K\mathbf{N}_{K,d})^T\Big)_\omega \in \mathbb{C}^d \tag{B.15}$$

□

B.1.2 Sub-Sampling In Frequency And Space

We will now look at sub-sampling in both frequency and space.

Definition B.1.4. *Let L be a positive factor of d. Suppose measurements are collected at L equally spaced physical shifts of step-size $\frac{d}{L}$. We denote the set of shifts by $\mathcal{L}$, that is*

$$\mathcal{L} = \frac{d}{L}[L]_0 = \Big\{0, \frac{d}{L}, \frac{2d}{L}, \ldots, d - \frac{d}{L}\Big\} \tag{B.16}$$

Definition B.1.5. *Let $\mathbf{A} \in \mathbb{C}^{d\times d}$, $L \mid d$. We denote by $\mathbf{A}_{d,L} \in \mathbb{C}^{d\times L}$ the sub-matrix of $\mathbf{A}$ whose rows are those of $\mathbf{A}$, sub-sampled in step-size $\frac{d}{L}$.*

We will now prove a similar lemma as before, but now we will sub-sample in both frequency and space.

Lemma B.1.4. *([77], Lemma 2.1.2.) Suppose we have noisy spectrogram measurements collected on a subset $\mathcal{K} \subseteq [d]_0$ of K equally spaced frequencies and a subset $\mathcal{L} \subseteq [d]_0$ of L equally spaced physical shifts. Then for any $\omega \in [K]_0, \alpha \in [L]_0$*

$$\left(\mathbf{F}_L\left(\mathbf{Y}_{K,L}^T(\mathbf{F}_K^T)_\omega\right)\right)_\alpha = \frac{KL}{d}\sum_{r=0}^{\frac{d}{K}-1}\sum_{\ell=0}^{\frac{d}{L}-1}\left(\mathbf{F}_d((\mathbf{x}\circ S_{\omega-rK}\bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}}\circ S_{rk-\omega}\bar{\tilde{\mathbf{m}}}))\right)_{\alpha-\ell L} + \left(\mathbf{F}_L\left(\mathbf{N}_{K,L}^T(\mathbf{F}_K^T)_\omega\right)\right)_\alpha$$

where $\mathbf{Y}_{K,L} - \mathbf{N}_{K,L} \in \mathbb{C}^{K\times L}$ is the matrix of sub-sampled noiseless $K \cdot L$ measurements.

Proof. For fixed $\ell \in [d]_0, \omega \in [K]_0$, we have computed

$$\left(\mathbf{F}_K(Z_{\frac{d}{K}}(\mathbf{y}_\ell))\right)_\omega = K\sum_{r=0}^{\frac{d}{K}-1}\left((\mathbf{x}\circ S_{\omega-rK}\bar{\mathbf{x}}) *_d \mathbf{F}_d(\tilde{\mathbf{m}}\circ S_{rk_\omega}\bar{\tilde{\mathbf{m}}})\right)_\ell$$

Fix $\omega \in [K]_0$, and define the vector $\mathbf{p}_\omega \in \mathbb{C}^L$ by

$$(\mathbf{p}_\omega)_\ell := \left(\mathbf{F}_K(Z_{\frac{d}{K}}(\mathbf{y}_{\ell\frac{d}{L}}))\right)_\omega + \left(\mathbf{F}_K(Z_{\frac{d}{K}}(\mathbf{J}_{\ell\frac{d}{L}}))\right)_\omega, \quad \forall \ell \in [L]_0$$

Note that the rows of $\mathbf{Y}_{K,L}, \mathbf{N}_{K,L} \in \mathbb{C}^{K\times L}$ are those of $\mathbf{Y}_{K,L}, \mathbf{N}_{K,L} \in \mathbb{C}^{K\times d}$, sub-sampled in step-size of $\frac{d}{L}$. Thus

$$(\mathbf{p}_\omega)_\ell = \left((\mathbf{Y}_{K,L})^T(\mathbf{F}_K^T)_\omega\right)_\ell + \left((\mathbf{Y}_{K,L})^T(\mathbf{F}_K^T)_\omega\right)_\ell$$

where $(\mathbf{F}_K^T)_\omega \in \mathbb{C}^K$ is the ω^{th} column of $\mathbf{F}_K^T$. Therefore

$$\mathbf{p}_\omega = \mathbf{Y}_{K,L}^T(\mathbf{F}_K^T)_\omega + \mathbf{N}_{K,L}^T(\mathbf{F}_K^T)_\omega \in \mathbb{C}^L, \quad \forall \omega \in [K]_0$$

For any $\ell \in [L]_0$, we have

$$
\begin{aligned}
(\mathbf{p}_\omega)_\ell &= K \sum_{r=0}^{\frac{d}{K}-1} \left((\mathbf{x} \circ S_{\omega - rK} \bar{\mathbf{x}}) *_d \mathbf{F}_d (\tilde{\mathbf{m}} \circ S_{rk_\omega} \bar{\tilde{\mathbf{m}}}) \right)_{\ell \frac{d}{L}} + \mathbf{N}_{K,L}^T (\mathbf{F}_K^T)_\omega \\
&= K \cdot \left(Z_{\frac{d}{L}} \left(\sum_{r=0}^{\frac{d}{K}-1} (\mathbf{x} \circ S_{\omega - rK} \bar{\mathbf{x}}) *_d \mathbf{F}_d (\tilde{\mathbf{m}} \circ S_{rk_\omega} \bar{\tilde{\mathbf{m}}}) \right) \right)_\ell + \mathbf{N}_{K,L}^T (\mathbf{F}_K^T)_\omega
\end{aligned}
$$

and for any $\alpha \in [L]_0$, by aliasing, we have that

$$
(\mathbf{F}_L \mathbf{p}_\omega)_\alpha = \frac{KL}{d} \sum_{r=0}^{\frac{d}{K}-1} \sum_{\ell=0}^{\frac{d}{L}-1} \left(\mathbf{F}_d ((\mathbf{x} \circ S_{\omega - rK} \bar{\mathbf{x}}) *_d (\tilde{\mathbf{m}} \circ S_{rk-\omega} \bar{\tilde{\mathbf{m}}})) \right)_{\alpha - \ell L} + \left(\mathbf{F}_L \left(\mathbf{N}_{K,L}^T (\mathbf{F}_K^T)_\omega \right) \right)_\alpha
$$

□

APPENDIX C

BLIND DECONVOLUTION

C.1 Alternative Approach

In this section we discuss the convex relaxation approach studied in [2].

C.1.1 Convex Relaxation

In [2], the approach is to solve a convex version of the problem. Given $\mathbf{y} \in \mathbb{C}^L$, their goal is to find $\mathbf{h} \in \mathbb{R}^k$, $\mathbf{x} \in \mathbb{R}^N$ that are consistent with the observations. Making no additional assumptions other than the dimensions, the way to choose between multiple feasible points is by solving using least-squares. That is,

$$\text{minimize}_{\mathbf{u},\mathbf{v}} \ \|\mathbf{u}\|_2^2 + \|\mathbf{v}\|_2^2 \qquad \text{subject to} \ \ \mathbf{y}(\ell) = \langle \mathbf{c}_\ell, \mathbf{u} \rangle \langle \mathbf{v}, \mathbf{b}_\ell \rangle, \quad 1 \leq \ell \leq L$$

This is a non-convex quadratic optimization problem. The cost function is convex, but the quadratic equality constraints mean that the feasible set is non-convex. The dual of this minimization problem is the SDP and taking the dual again will give us a convex program

$$\min_{\mathbf{W}_1, \mathbf{W}_2, \mathbf{X}} \frac{1}{2} tr(\mathbf{W}_1) + \frac{1}{2} tr(\mathbf{W}_2) \qquad \text{subject to} \begin{bmatrix} \mathbf{W}_1 & \mathbf{X} \\ \mathbf{X}^* & \mathbf{W}_2 \end{bmatrix} \geq 0, \mathbf{y} = \mathcal{A}(\mathbf{X})$$

which is equivalent to

$$\min \|\mathbf{X}\|_* \qquad \text{subject to} \ \ \mathbf{y} = \mathcal{A}(\mathbf{X})$$

where $\|\mathbf{X}\|^* = tr(\sqrt{\mathbf{X}^*\mathbf{X}})$ denotes the nuclear norm.

In [2], they achieved guarantees for relatively large K and N, when B is incoherent in the Fourier domain, and when C is generic. We can now outline the algorithm from [2].

Algorithm C.1 Convex Relaxed Blind Deconvolution Algorithm

Input: Normalized Fourier measurement $\mathbf{y}$,

Output: Estimate underlying signal and blurring function

1) Compute $\mathcal{A}^*(\mathbf{y})$

2) Find the leading singular value, left and right singular vectors of $\mathcal{A}^*(\mathbf{y})$, denoted by d, $\tilde{\mathbf{h_0}}$, and $\tilde{\mathbf{x_0}}$ respectively

3) Let $\mathbf{X}_0 = \tilde{\mathbf{h}}_0\tilde{\mathbf{x}}_0^*$ denote the initial estimate and solve the following optimization problem

$$\begin{aligned} &\min && \|\mathbf{X}\|_* \\ &\text{subject to} && \|\mathbf{y} - \mathcal{A}(\mathbf{X})\| \leq \delta \end{aligned} \tag{C.1}$$

where $\|\cdot\|_*$ denotes the nuclear norm and $\|\mathbf{e}\|_2 \leq \delta$

Return $(\mathbf{h}, \mathbf{x})$ for $\mathbf{X} = \mathbf{h}\mathbf{x}^*$