

Optimal Standoff Imaging using Structured Laser Illumination and Graphical Models

by

Nicholas David Hardy

Submitted to the Department of Electrical Engineering and Computer
Science

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2014

© Massachusetts Institute of Technology 2014. All rights reserved.

Author
Department of Electrical Engineering and Computer Science
August 29, 2014

Certified by
Jeffrey H. Shapiro
Julius A. Stratton Prof. of Elec. Eng.
Thesis Supervisor

Accepted by
Leslie A. Kolodziejcki
Chairman, Department Committee on Graduate Theses

Optimal Standoff Imaging using Structured Laser Illumination and Graphical Models

by

Nicholas David Hardy

Submitted to the Department of Electrical Engineering and Computer Science
on August 29, 2014, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Structured illumination can be used to form images without using a lens or a detector array. A series of spatially-structured laser pulses is cast on the scene of interest, and a single-detector power measurement is made on the light each pulse returns from the scene. There has been significant interest in the “ghost imaging” configuration, in which the spatial patterns are randomly generated—e.g., by driving the pixels of a spatial light modulator with independent, identically-distributed pseudorandom inputs—and the sequence of measurements is correlated with reference versions of those patterns to image the scene. This naive reconstruction, however, is far from optimal for standoff imaging, for which rough-surfaced objects create laser speckle in the measurements. We develop a graphical model that encompasses the probabilistic relationships in structured-illumination standoff imaging along with an approximate message-passing algorithm for belief propagation to perform optimal scene reconstruction. This approach lets us accurately model the statistics of speckled images, photon detection, and atmospheric turbulence, as well as incorporate intelligent priors for the scene that capture the inherent structure of real-world objects. The result is state-of-the-art scene reconstructions.

Thesis Supervisor: Jeffrey H. Shapiro
Title: Julius A. Stratton Prof. of Elec. Eng.

Acknowledgments

This thesis would not have been completed without the support of many people. First and foremost is my advisor, Professor Jeffrey Shapiro. He has been a constant source of guidance and knowledge during my graduate studies, and tirelessly edited this thesis over the last few months. I also need to thank Professor Vivek Goyal, who started my foray into wavelet-basis signal processing, and also served as a reader on my thesis committee.

My parents, David and Catherine Hardy, have been incredibly supportive and encouraging throughout my studies. Sometimes going home is the best medicine, and I can't thank them enough for always being there for me. I would also like to thank my soon-to-be-wife Beverly Chu. She has been incredibly supportive and patient over the last year, and provided invaluable editing and feedback for this thesis. I cannot imagine how I would have gotten here without her.

Of course, no graduate studies are complete without peers to help push you along. While everyone in the Optical and Quantum Communications group was a joy to work with, I would especially like to thank Nivedita Chandrasekaran, Christopher Chudzicki, and Justin Dove for their insightful conversation and collaboration.

This research has been supported by the DARPA Information in a Photon Program through U.S. Army Research Office grant number W911NF-10-1-0404 and the National Science Foundation under grant number 1161413.

Contents

1	Introduction	13
1.1	Active Standoff Imaging	13
1.2	Thesis Outline	17
1.3	Notation	18
1.3.1	Integration	18
1.3.2	Functions of Random Variables	19
2	Physical Framework	23
2.1	Optical Source	23
2.2	Propagation Through Turbulence	25
2.3	Target Reflection	28
2.4	Detection	31
2.5	Summary	34
3	Ghost Imaging	37
3.1	Imaging Framework	40
3.2	Spatial Resolution	42
3.3	Signal-to-Noise Ratio	45
3.4	Comparison to a Floodlight Laser Radar	51
3.4.1	Framework	52
3.4.2	Spatial Resolution	55
3.4.3	Signal-to-Noise Ratio	56
3.5	Performance Comparison	58

3.5.1	Spatial-Resolution Comparison	59
3.5.2	Signal-to-Noise Ratio Comparison	60
3.5.3	Conclusions	63
3.6	Advances in Ghost Imaging	65
4	Discretization of the On-Target Irradiance and the Target's Field-Reflection Coefficient	69
4.1	Reference-Pattern Discretization	71
4.2	Target Field-Reflection Coefficient	73
4.3	Target-to-Measurements Transformation	74
4.3.1	LMMSE Estimation of the Coupling Coefficient	75
4.3.2	Model of Received Photon Number	80
4.4	Discrete Simulator	84
4.4.1	Statistics of \mathbf{z}_n and \mathbf{v}_n via Simulation	85
5	Cramér-Rao Bound	89
5.1	A Bound on Mean-Squared Error	89
5.2	Non-Bayesian Cramér-Rao Bound	92
5.2.1	Covariance Matrix for no Discretization Error	95
5.2.2	Fisher Information when Oversampling	97
5.2.3	Fisher Information when Undersampling	98
5.2.4	Comparison to Ghost Imaging Theory	99
5.3	Bayesian Cramér-Rao Bound	102
5.3.1	Expected Fisher Information when Oversampling	106
5.3.2	Expected Fisher Information when Undersampling	106
6	Linear Minimum Mean-Squared Error Estimator	111
6.1	Unconditional Mean Values	112
6.2	Target and Measurement Cross-Covariance	113
6.3	Measurement Autocovariance	114
6.4	Theoretical and Simulated Performance	116

6.4.1	Comparison of Theory and Simulation	116
6.4.2	Comparison to the Cramér-Rao Bound	117
7	Approximate Inference for Optimal Estimation	121
7.1	Optimal Estimation	123
7.2	Graphical Modeling	126
7.3	Loopy Belief Propagation	127
7.3.1	Sum-Product Algorithm	128
7.3.2	Max-Sum Algorithm	130
8	Adapting GAMP for Standoff Optical Imaging	133
8.1	Developing GAMP	135
8.1.1	Approximations	137
8.1.2	Factor-Node Computations	138
8.1.3	Variable-Node Computations	140
8.1.4	GAMP Algorithm	144
8.2	Fitting GAMP to the Physics-of-Imaging Framework	144
8.2.1	GAMP for Structured-Intensity Imaging	146
8.2.2	Additive-Noise Term	147
8.2.3	Matrix Uncertainty	148
8.2.4	Non-Centered Matrix	150
8.2.5	Reconciling with the GAMP Algorithm	154
8.3	Comparison to LMMSE Estimate and CRB	160
9	Extending GAMP: Parameter Estimation, Wavelet-Basis Prior, and Speckle Mitigation	167
9.1	Adaptive GAMP	168
9.2	Estimating the Target in a Wavelet Basis	170
9.2.1	Wavelet-Basis Prior	173
9.2.2	Measurement-Matrix Centering for Wavelet Basis	174
9.2.3	Simulation Results	175

9.3	Exposing Speckle in the Model	179
9.3.1	Model of Integrated Speckle	179
9.3.2	Connection of Speckle to Measurements	182
9.3.3	Perturbative Expansion	182
9.4	Joint Target and Speckle Estimation	185
9.4.1	Maximum Likelihood Approximation to Expectation Maximiza- tion Update	186
9.4.2	Speckle Estimation	189
9.4.3	Final Algorithm	190
9.4.4	Simulation Results	192
10	Conclusions	197
A	Physical System	205
A.1	Turbulence Square-Law Approximation	205
A.2	Computing the Reference Irradiance Pattern for SLM-Generated Field	207
A.2.1	Efficient Computation of Reference Field	208
A.2.2	Average Reference Irradiance	211
A.3	Rough-Surface Statistics	212
A.3.1	Mean	213
A.3.2	Covariance	213
A.4	Rough-Surface Induced Speckle	214
A.4.1	Precise Noise Calculations	215
A.4.2	Approximate Noise Calculations	217
B	Useful Mathematics	219
B.1	Exponential Families	219
B.2	Gaussian Moment-Factoring Theorem	220
B.3	Jensen’s Inequality	221
B.4	Cauchy-Schwarz Inequality	221
B.5	Generalized Expectation-Maximization Algorithm	222

B.5.1	Derivation of Standard Lower Bound	223
B.6	Rank-One Updates	224
C	Additional Derivations for GAMP	227
C.1	Correctness of Non-Centered GAMP Variant	227
C.1.1	Simplify Message Components	228
C.1.2	Simplify Log-Likelihood Components	229
C.2	Input Estimators	232
C.2.1	Gaussian Prior for MMSE Estimation	233
C.2.2	Gamma Prior for MAP Estimate	235
C.3	Output Estimators	236
C.3.1	Gaussian Measurement Channel	238
	Acronyms	247
	Glossary	248

Chapter 1

Introduction

Standoff imaging, in which one acquires the spatial reflectivity pattern of an object that may be kilometers away, has a wide range of applications. These uses range from military intelligence gathering to surveying, astronomy, and hobbyist photography. A prime example of a standoff-imaging device is the camera, which only requires a lens and a photodetector array to take pictures, and can be miniaturized to fit on a cell phone. The camera is a passive imaging device, however, so it can only image naturally illuminated or self-luminous objects. Flash-enabled photography is an active imaging technique,¹ but its effectiveness is limited to distances of a few meters, because light from the flash spreads too fast to be of use at kilometer ranges.

1.1 Active Standoff Imaging

Laser radar, or LADAR, is an active imaging technique at optical wavelengths that replaces flash photography's spatially-incoherent light source with a laser illuminator, making it capable of standoff imaging.

LADAR has primarily been used in two modalities: (1) a repetitively-pulsed laser beam is raster-scanned over the target, and the strength and time-of-flight for the light returned from each pulse is detected to produce reflectivity and range information

¹In passive sensing one measures an existing signal, while in active sensing one directly stimulates the target to produce the signal that is measured.

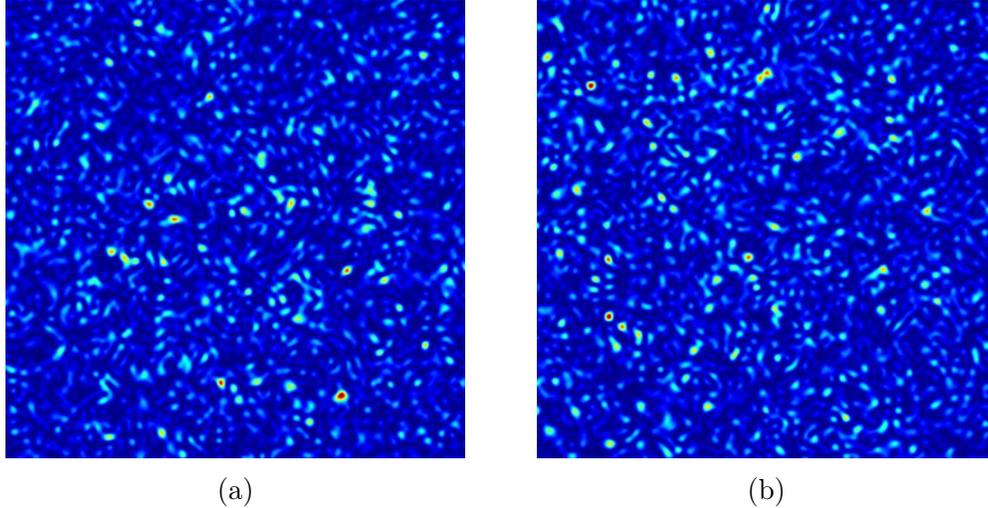


Figure 1-1: Two examples of the intensity patterns of pseudo-randomly generated structured fields in planes orthogonal to the direction of propagation.

about the area that was illuminated by that pulse; and (2) a repetitively-pulsed laser beam illuminates a larger region on the target and the light returned from that region is imaged onto a photodetector array to obtain spatially-resolved reflectivity and range information about the illuminated portion of the target. Modern LADARs tend to use the floodlight/array configuration, because it can image large scenes more rapidly than raster-scanning systems [1].

In recent years an alternative to traditional LADAR has arisen: imaging with structured laser illumination. In structured-illumination imaging a series of spatially-structured pulses (examples of pseudo-randomly generated structured fields, which is our main focus, are shown in Fig. 1-1) are transmitted towards a target. Each pulse reflects off the target and produces a unique field at the detector plane. Similarly to the first LADAR modality, a single measurement is made per pulse. However, because each measurement is a mixture of reflections from different parts of the target, these measurements alone are not enough to infer the target's reflectivity pattern; knowledge of the on-target structured fields is also required. This basic setup for structured-illumination laser imaging is shown in Fig. 1-2.

The best way to use this information (the combination of measurements and on-target field patterns) to infer the reflectivity pattern of the target is an open question,

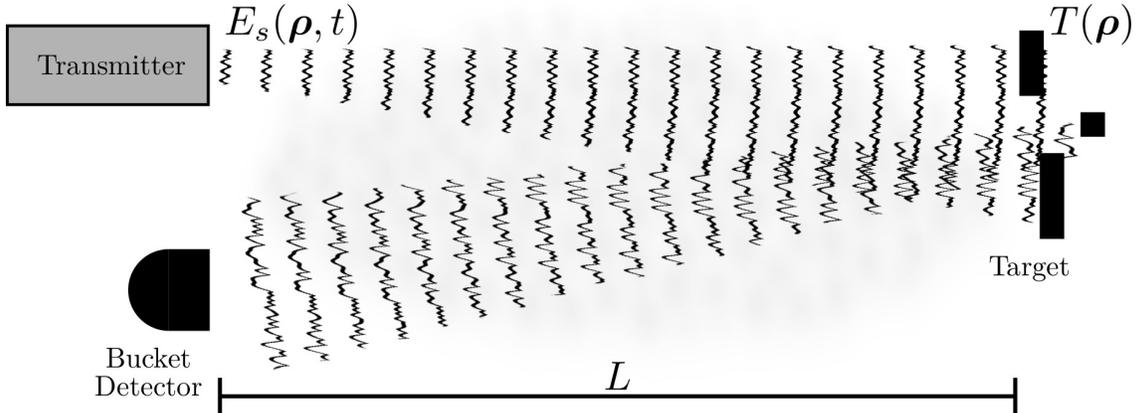


Figure 1-2: The setup for our structured-illumination laser imaging framework. The transmitter creates a series of structured fields $E_s(\cdot)$ that propagate to—and reflect off—the target. The reflected field propagates back to a bucket detector, where a portion is collected and detected for subsequent processing to form an image of the target.

and the primary focus of this thesis. Our goal is to advance the state-of-the-art for scene reconstruction in remote imaging when using structured laser illumination by first developing a rigorous physical framework for the problem, and then adapting modern signal processing methods to construct optimized inference algorithms.

Motivation for using Structured Illumination for Standoff Imaging

Our interest in advancing the use of structured illumination for standoff imaging arose from our work on ghost imaging for standoff imaging through atmospheric turbulence [2–5]. While structured illumination has many uses and modalities—perhaps its most publicized use is to create depth maps in the XBOX Kinect sensor [6]—our approach begins by generalizing the ghost imaging framework developed in [5]. In this scenario a sequence of laser pulses with randomly generated spatial patterns are transmitted toward the target. Light reflected by the target propagates back to the bucket detector. For a time-of-flight delay—associated with the range to the target—the returned photon-flux is measured for each pulse, creating a vector of measurements. These measurements, in conjunction with knowledge of the on-target spatial pattern produced by each pulse, permit formation of an image for the target

at that range delay.

Neither knowledge of the on-target spatial patterns nor the bucket detector measurements alone are sufficient to form a target image. The “ghost image” name originates from how the image emerges from knowledge of both the on-target patterns and the bucket measurements. We can therefore determine the reflectivity pattern of the entire target by correlating the vector of bucket detector measurements with the vector of photon-flux densities (found from the known on-target spatial patterns) at each transverse location.

This is a simplistic view of ghost imaging; see Ch. 3 for a full history and analysis of standard correlation-based ghost imaging. For now it is sufficient to note that this general framework—in which pseudo-randomly generated fields are cast on the target and the reflected photons are measured by a bucket detector—does not have to be limited to the ghost-imaging method of target estimation. Indeed, as explained in Ch. 3, compressed sensing techniques have been applied to achieve a substantial reduction in the number of spatial patterns needed to form an image from the Fig. 1-2 setup. However, the methods employed in standard compressive sensing algorithms are better suited to speckle-free transmissive imaging, and are not optimized for obtaining a target image from structured-illumination standoff measurements. Using modern graphical model inference techniques to develop such an optimal structured-illumination imager is the goal of this thesis.

As a final note to these introductory remarks we point out that for our work to be applicable to real-world scenarios, we will need to account for target-induced speckle and atmospheric turbulence. The early experiments and accompanying theory for ghost imaging observed objects in transmission [7–12], whereas standoff sensing requires them to be measured in reflection. When seen in reflection, however, surface-roughness on the order of the laser wavelength gives rise to quasi-Lambertian reflections that, after propagation to the bucket detector, produce strong, spatially-varying fluctuations in the photon flux known as laser speckle. Our work [3–5] was the first to address the impact of speckle on conventional ghost imaging.

Standoff sensing must also cope with the effects of propagation through the at-

mosphere. The earth’s atmosphere is comprised of a variety of gases and entrained particulates, causing wavelength-dependent absorption and Rayleigh scattering [13]. These effects will cause attenuation, but that can be minimized with an appropriate choice of the operating wavelength. The most deleterious atmospheric effects arise from propagating in bad weather, i.e., though fog or clouds [13]. Here the scattering is so severe that we do not expect to be able to perform optical imaging of any type. Even in clear weather, however, the atmosphere is in constant flux. This random mixing of the air parcels with $\sim 1K$ temperature fluctuations creates random spatiotemporal variations in the refractive index known as atmospheric turbulence [13]. Ghost imaging has been analyzed for propagation through atmospheric turbulence [5, 14], and the effects of turbulence will be included in our structured illumination framework.

1.2 Thesis Outline

We start in Ch. 2 by formalizing the physical framework for our imaging scenario upon which we will build the rest of the thesis. This framework includes descriptions of the optical transmitter that generates structured illumination, and the subsequent propagation, target reflection, and photodetection processes. In Ch. 3 we use our framework to develop a pulsed ghost-imaging system that operates through time-varying turbulence, and compare its performance to that of a more traditional pulsed LADAR system. This analysis is carried out for a continuous-space target (no artificial pixelation), but to further our analysis and develop more advanced algorithms we need to discretize the framework from Ch. 2. This discretization, which is carried out in Ch. 4, is non-trivial.

We complete this new discretized framework in Ch. 5, by establishing theoretical limits for the possible performance of *any* inference algorithm. We do this by deriving the Cramér-Rao bound (CRB) [15], which is a lower bound on the achievable mean-squared estimation error across all unbiased estimators, that will serve as our performance goal when developing new estimators.

The second part of this thesis begins in Ch. 6, where we develop the linear minimum mean-squared error (LMMSE) estimator for the Ch. 4 framework. This is the estimator with the least expected squared-error among all estimators that are formed from a linear combination of the measurements. It is thus computationally efficient, but does not take full advantage of the richness of the probabilistic framework established in Ch. 4.

To approach truly optimal estimators we turn to modern approximate inference techniques based on graphical modeling and belief propagation. These concepts are presented in Ch. 7, as they are fundamental to understanding the Generalized Approximate Message Passing (GAMP) algorithm [16, 17], which we adapt to our imaging scenario in Ch. 8 in our first attempt at optimal structured-illumination target estimation. We extend this new algorithm in Ch. 9 to use a more complex prior model on both the target and the speckle inherent in the imaging system, resulting in a state-of-the-art algorithm that is the culmination of this thesis.

1.3 Notation

To facilitate the reader’s comprehension of the mathematical formulas in this thesis, we will now lay out some of the notation that will be used. Working at the juncture of optical physics and probabilistic inference algorithms results in a significant amount of notation, some of which clashes between the two fields. We have attempted to unify the notation used throughout, keeping it as faithful as possible to what is standard in each field. A glossary has been provided at the end of the thesis with short descriptions of each term used, with a reference to its first use.

1.3.1 Integration

Evaluating paraxial propagation of electromagnetic fields (e.g., laser beams) via diffraction theory involves a two-dimensional integral over a plane. Our analysis of the imaging scenario will involve many such propagations. Indeed, when analyzing the signal-to-noise ratio of ghost imaging there will be concatenated integrations over 18

different variables. We thus adopt a compact notation, representing the two Cartesian coordinates (ρ_x, ρ_y) for each plane with the coordinate vector $\boldsymbol{\rho}$. Moreover, to maintain clarity about which differential elements go with which integrals, the two will be kept together in a pseudo-operator notation. The integration of some function $f(\rho_x, \rho_y)$ over the infinite plane then becomes

$$\iint f(\rho_x, \rho_y) d\rho_x d\rho_y \longrightarrow \int d\boldsymbol{\rho} f(\boldsymbol{\rho}). \quad (1.1)$$

This notation becomes even more useful when the integrations are over subsets of the planes. Optical propagation often involves performing the integration over a pupil function. This could be the propagation through a transmitter's exit pupil, a receiver's entrance pupil, or over the photosensitive region of a photodetector. We represent the *field aperture* function as $\mathcal{A}(\boldsymbol{\rho})$, which we take to satisfy $0 \leq \mathcal{A}(\boldsymbol{\rho}) \leq 1$, and adopt the interchangeable notation

$$\int d\boldsymbol{\rho} \mathcal{A}(\boldsymbol{\rho}) f(\boldsymbol{\rho}) = \int_{\mathcal{A}} d\boldsymbol{\rho} f(\boldsymbol{\rho}). \quad (1.2)$$

For photodetection we integrate over the square of the field aperture, \mathcal{A}^2 , and use the same notational convenience by writing

$$\int d\boldsymbol{\rho} \mathcal{A}^2(\boldsymbol{\rho}) f(\boldsymbol{\rho}) = \int_{\mathcal{A}^2} d\boldsymbol{\rho} f(\boldsymbol{\rho}). \quad (1.3)$$

Most often \mathcal{A} will simply be an indicator function, equaling unity within an area, and zero outside; in such cases $\mathcal{A}^2 = \mathcal{A}$. However, it will sometimes be useful to approximate the \mathcal{A} region with a Gaussian function for which $\mathcal{A}^2 \neq \mathcal{A}$, so we will maintain the notational difference even when there is no functional difference.

1.3.2 Functions of Random Variables

In the standoff active imaging scenarios we shall consider, many random (stochastic) processes play significant roles. These include atmospheric turbulence, rough-surface scattering effects, and photodetection shot noise. For much of our analysis these quan-

tities are treated as random variables and processed without confusion. However, for some quantities we need to distinguish between the random variable and a particular *realization* of that variable². We will make note of these cases as they occur, but to simplify the distinction for the reader we will denote the random variable with a sans-serif font, and the realization with the standard serif or math font, i.e., \mathbf{x} is a random variable, while x is a deterministic realization of that variable.

Distributions

For some random variables we explicitly define distributions. For instance, if \mathbf{x} is a Gaussian random variable with mean μ and variance σ^2 , then its probability density function $p_{\mathbf{x}}(x)$ is

$$p_{\mathbf{x}}(x) = \frac{e^{-(x-\mu)^2/2\sigma^2}}{\sqrt{2\pi\sigma^2}}. \quad (1.4)$$

This distribution is *parameterized* by μ , and σ^2 , and in some cases we may wish to make that explicit as $p_{\mathbf{x}}(x) \rightarrow p_{\mathbf{x}}(x; \mu, \sigma^2)$. We also define joint and conditional distributions for multiple random variables, e.g.,

$$p_{y,\mathbf{x}}(y, x), \quad p_{\mathbf{x}|y}(x|y). \quad (1.5)$$

for the two-dimensional case.

Expectations

The expected value of a function $f(\cdot)$ of a random variable \mathbf{x} is usually denoted as $\mathbf{E}_{\mathbf{x}}[f(\mathbf{x})]$ or $\langle f(\mathbf{x}) \rangle_{\mathbf{x}}$. In this thesis we will use the latter notation; for a random variable \mathbf{x} with a continuous probability distribution $p_{\mathbf{x}}(x)$, the expected value of the function is

$$\langle f(\mathbf{x}) \rangle_{\mathbf{x}} = \int dx f(x) p_{\mathbf{x}}(x), \quad (1.6)$$

²For instance, we develop a statistical model for the bucket detector measurements, but in our inference algorithms we process actual measurements.

When it is obvious which random variables are being averaged over (or if the expectation is over all random variables in the function) we will drop the subscript, i.e. $\langle \mathbf{x} \rangle = \langle \mathbf{x} \rangle_{\mathbf{x}}$. This same convention will be used when dealing with variances and covariances,

$$\mathbf{var}_{\mathbf{x}}[f(\mathbf{x})] = \langle (f(\mathbf{x}) - \langle f(\mathbf{x}) \rangle_{\mathbf{x}})^2 \rangle_{\mathbf{x}} \quad (1.7)$$

$$= \langle f^2(\mathbf{x}) \rangle_{\mathbf{x}} - \langle f(\mathbf{x}) \rangle_{\mathbf{x}}^2 \quad (1.8)$$

$$\mathbf{cov}_{\mathbf{x}}[f(\mathbf{x}), g(\mathbf{x})] = \langle (f(\mathbf{x}) - \langle f(\mathbf{x}) \rangle_{\mathbf{x}})(g(\mathbf{x}) - \langle g(\mathbf{x}) \rangle_{\mathbf{x}}) \rangle_{\mathbf{x}} \quad (1.9)$$

$$= \langle f(\mathbf{x})g(\mathbf{x}) \rangle_{\mathbf{x}} - \langle f(\mathbf{x}) \rangle_{\mathbf{x}} \langle g(\mathbf{x}) \rangle_{\mathbf{x}}. \quad (1.10)$$

The expectation of a function of two random variables, $f(\mathbf{y}, \mathbf{x})$, conditioned on knowledge of \mathbf{y} , is written as

$$\langle f(\mathbf{y}, \mathbf{x}) | \mathbf{y} \rangle_{\mathbf{x}} = \int dx f(\mathbf{y}, x) p_{\mathbf{x}|\mathbf{y}}(x|\mathbf{y}). \quad (1.11)$$

When $\mathbf{y} = y$ is given, Eq. (1.11) becomes a number,

$$\langle f(\mathbf{y}, \mathbf{x}) | \mathbf{y} \rangle_{\mathbf{x}} = \int dx f(y, x) p_{\mathbf{x}|y}(x|y). \quad (1.12)$$

Otherwise $\langle f(\mathbf{y}, \mathbf{x}) | \mathbf{y} \rangle_{\mathbf{x}}$ is a random variable because it is a function of the random variable \mathbf{y} .

We also need to differentiate statistical measures arising from prior distributions, versus those that are calculated from measurements. In general, we use an overbar to denote a prior mean, while a caret is used for an estimate that incorporates extra data, i.e., \mathbf{x} has prior mean $\bar{x} = \langle \mathbf{x} \rangle$, while we denote the posterior estimate of \mathbf{x} given the observation $\mathbf{y} = y$ as $\hat{x} = \langle \mathbf{x} | y \rangle$. In this case \hat{x} is a posterior mean, but carets will be used for other types of estimates as well.

Chapter 2

Physical Framework

This thesis is concerned with the development and analysis of structured-illumination imaging systems for standoff applications. To compare the relative performance of different imaging methods we need a common physical framework, which is the focus of this chapter. The general setup, shown in Fig. 1-2, is straightforward: a transmitter generates a series of optical pulses that radiate out to the target; upon hitting the target some of that light reflects back toward the receiver where a portion of the reflected light is detected. Subsequent signal processing then yields an image (estimate) of the target.

We will walk through this framework in that order, first characterizing our transmitter, then setting up the propagation through atmospheric turbulence to the target, and finally describing the reflection from the target and subsequent photodetection. At the end of the chapter we summarize and expand on the assumptions and approximations made for this framework. Some of this chapter has been published [5].

2.1 Optical Source

In this thesis we limit our consideration to standoff imaging in reflection at wavelengths in the infrared to ultraviolet region. In other words, we are interested in imaging targets at kilometer or longer ranges using laser sources and photodetection receivers. We model our transmitter as emitting quasimonochromatic, classical scalar

waves with a center frequency ω_0 (with corresponding wavelength $\lambda_0 = 2\pi c/\omega_0$ and wavenumber $k_0 = \omega_0/c$, where c is the speed of light). The field has a complex envelope $E_s(\boldsymbol{\rho}, t)$ as a function of its transverse coordinates $\boldsymbol{\rho} = (\rho_x, \rho_y)$ and time t , and is normalized so that $|E_s(\boldsymbol{\rho}, t)|^2$ is the photon flux density, with units of photons/m²-s.

Initial theory and experiments for reflective ghost imaging used continuous-wave sources [4]. However, in order to resolve returns from multiple depths (and because discrete pulses will be needed for more advanced reconstruction methods), we will assume a pulsed-mode transmitter that radiates a sequence of spatial patterns riding on a periodic train of identical temporal pulses,

$$E_s(\boldsymbol{\rho}, t) = \sqrt{\mathbb{N}_s} \sum_{n=0}^{N-1} \xi(\boldsymbol{\rho}, n) p(t - n\tau_s). \quad (2.1)$$

where: N is the number of pulses, \mathbb{N}_s is number of transmitted photons per pulse, τ_s is the pulse repetition period, $\xi(\boldsymbol{\rho}, n)$ is the normalized spatial mode ($\int d\boldsymbol{\rho} |\xi(\boldsymbol{\rho}, n)|^2 = 1$) for pulse n , and $p(t)$ is the normalized pulse shape ($\int dt |p(t)|^2 = 1$). The pulse shape is time-limited to $|t| \leq \tau_p/2$, where τ_p is the pulse width, which we expect to be picoseconds to nanoseconds in duration. To avoid inter-pulse interference we require $\tau_s > \tau_p$. If we attempt to image a three dimensional target with significant depth, the pulse repetition period will have to be extended to avoid overlap from pulses returning from different parts of the target, i.e., for an extended target of depth ΔL we assume $\tau_s > 2\Delta L/c + \tau_p$.

Structured-illumination imagers may use random, pseudo-random, or deterministic spatial patterns. For our analysis purposes, we will take the $\{\xi(\boldsymbol{\rho}, n)\}$ to be independent, identically distributed, zero-mean Gaussian random fields, which are completely characterized by the following Gaussian-Schell model correlation functions,

$$\langle \xi(\boldsymbol{\rho}_1, n_1) \xi(\boldsymbol{\rho}_2, n_2) \rangle = 0 \quad (2.2)$$

$$\langle \xi^*(\boldsymbol{\rho}_1, n_1) \xi(\boldsymbol{\rho}_2, n_2) \rangle = \delta_{n_1, n_2} \frac{2}{\pi a_0^2} e^{-(|\boldsymbol{\rho}_1|^2 + |\boldsymbol{\rho}_2|^2)/a_0^2} e^{-|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2|^2/2\rho_0^2}. \quad (2.3)$$

In this model a_0 is the intensity radius of the field, while $\rho_0 \ll a_0$ is its coherence length.

This partially-coherent field can be generated in two principal ways. The first is to pass a stream of laser pulses through a rotating ground-glass diffuser. The second method is to shine that same laser pulse stream on a spatial light modulator (SLM) whose array of pixels impose independent pseudo-random phases pixel-to-pixel and pulse-to-pulse [18]. It should be noted that in both methods Gaussian field statistics are not accurate near the transmitter. However, targets of interest will be in the transmitter's far-field, and each location on the target will be illuminated by the superposition of many coherence areas (for the ground-glass transmitter) or many pixels (for the SLM transmitter), so that the Central Limit Theorem (CLT) can be applied to the resulting field [18]. This on-target field (L -m away from the source) will have a new coherence length ρ_L and intensity radius a_L that are computed from the source statistics as

$$\rho_L = 2L/k_0a_0, \quad a_L = 2L/k_0\rho_0. \quad (2.4)$$

2.2 Propagation Through Turbulence

For structured-illumination imaging in a standoff scenario, light will propagate through kilometers of turbulent atmosphere to the target and back to the detector. In bad weather, e.g., fog, there will be so much absorption and scattering that imaging will be impossible. We therefore assume operation in good weather, where we only contend with clear-air turbulence effects. In such scenarios, atmospheric turbulence causes refractive index changes on the order of 10^{-6} [13]. While these changes might seem small, the accumulated effect as light propagates can be profound. For instance, light fields at wavelength $\lambda_0 = 1.5 \mu\text{m}$ that propagate through media with a refractive index difference 10^{-6} suffer a π -rad phase offset in just 0.75 m [2]. Over several kilometers of propagation the phases can become completely uncorrelated. Even worse, spatially-varying turbulence causes constructive and destructive interference

during the propagation, yielding scintillation (intensity fluctuations) in the on-target light [19].

In this thesis we use the Kolmogorov model for turbulence, which attributes clear-air turbulence index of refraction variations—for a spatial scale between a maximum of $\ell_{\max} \simeq 10 - 100$ m and a minimum of $\ell_{\min} \simeq 10^{-3}$ m—to temperature fluctuations. The temperature fluctuation's spatial-structure function follows a two-thirds power law so that the mean-square temperature difference between two points is proportional to $|\boldsymbol{\rho}|^{2/3}$ where $\boldsymbol{\rho}$ is the coordinate difference between two points. The temperature fluctuations also have a time dependence, and these temporal fluctuations come from two sources: the changing of the shape of the refractive-index eddies as they mix and evolve, and the drifting of these eddies in the wind [20]. In normal conditions, the time evolution is dominated by the latter mechanism, and the refractive-index structure has a typical coherence time of $\tau_\psi = 10^{-3} - 10^{-2}$ second. This implies that on shorter time scales the atmosphere is essentially frozen, and the turbulence effects are constant.

To model propagation of an input-plane field $E_m(\boldsymbol{\rho}, t)$ to an output-plane field, $E_m(\boldsymbol{\rho}, t, L)$ that is L -m away on path m , we use the extended Huygens-Fresnel principle [13],

$$E_m(\boldsymbol{\rho}, t, L) = \int d\boldsymbol{\rho}' E_m(\boldsymbol{\rho}', t - L/c) h_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t) \quad (2.5)$$

where $h_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ is the propagation kernel

$$h_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t) = \frac{k_0 e^{ik_0(L + |\boldsymbol{\rho} - \boldsymbol{\rho}'|^2/2L)} e^{\psi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)}}{i2\pi L} \quad (2.6)$$

at time t . It is the product of the Fresnel-diffraction Green's function and the atmospheric fluctuation term. The effect of atmospheric turbulence on path m is contained in the complex-valued fluctuation term $\psi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$, which arises from Kolmogorov-spectrum turbulence that is, in general, distributed non-uniformly along the L -m-long path [13]. The real and imaginary parts of this fluctuation, $\chi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ and

$\phi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$, can be physically interpreted as the log-amplitude and phase fluctuations along path m at time t , from point $\boldsymbol{\rho}'$ in the input plane to point $\boldsymbol{\rho}$ in the output plane [20].

In our system shown in Fig. 1-2, we have two paths of interest: the signal path, $m = s$, from the transmitter plane to the target, and the return path, $m = r$, from the target back to the bucket detector (receiver plane). Both of these paths are taken to have independent turbulence of possibly different distributions. The independence assumption requires a bistatic configuration in which the transmitter and the detector have non-overlapping pupils, although they still may be located close to one another.

In calculating the statistics of the detected signal we primarily need the mutual coherence function of the turbulence kernel on each path [13],

$$\langle e^{\psi_m(\boldsymbol{\rho}_1, \boldsymbol{\rho}'_1, t)} e^{\psi_m^*(\boldsymbol{\rho}_2, \boldsymbol{\rho}'_2, t)} \rangle = e^{-D_m(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2, \boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)/2}, \quad (2.7)$$

where

$$D_m(\boldsymbol{\rho}, \boldsymbol{\rho}') = 2.91k_0^2L \int_0^1 ds C_{n,m}^2(sL) |\boldsymbol{\rho}s + \boldsymbol{\rho}'(1-s)|^{\frac{5}{3}} \quad (2.8)$$

is the turbulence structure function, and $\{C_{n,m}^2(z) : 0 \leq z \leq L\}$ is the turbulence-strength along path m . At each depth z , $C_{n,m}^2(z)$ is assumed to be constant across the perpendicular plane. Although this is a rigorous description of the effects of the turbulence on the propagation, it is not computationally tractable for analysis. To obtain closed-form results that clarify our analysis we use the square-law approximation—which we derive in Appendix A.1—so that Eq. (2.8) becomes

$$D_m(\boldsymbol{\rho}, \boldsymbol{\rho}') = \frac{|\boldsymbol{\rho}|^2\zeta_m + \boldsymbol{\rho} \cdot \boldsymbol{\rho}'(8/3 - \zeta'_m - \zeta_m) + |\boldsymbol{\rho}'|^2\zeta'_m}{\varrho_m^2}, \quad (2.9)$$

where ζ'_m and ζ_m are weighting terms for the input and output planes, given by

$$\zeta'_m = \frac{8}{3} \int_0^1 ds (1-s)^2 \mathcal{C}_{n,m}^2(sL) \quad (2.10)$$

$$\zeta_m = \frac{8}{3} \int_0^1 ds s^2 \mathcal{C}_{n,m}^2(sL). \quad (2.11)$$

These in turn are functions of the normalized turbulence-strength profile

$$\mathcal{C}_{n,m}^2(z) = \frac{C_{n,m}^2(z)}{\int_0^1 ds C_{n,m}^2(sL)} \quad (2.12)$$

for path m .

The denominator in Eq. (2.9), ϱ_m , is the spherical-wave coherence length for path m when the turbulence over that path is distributed uniformly with the same integrated strength as the actual distribution. It is given by

$$\varrho_m = \left(1.09 k_0^2 L \int_0^1 ds C_{n,m}^2(sL) \right)^{-3/5}. \quad (2.13)$$

In what follows we assume that the weighted turbulence coherence length at the transmitter is much larger than the transmitter's coherence length ($\varrho_s/\sqrt{\zeta'_s} \gg \rho_0$), but will allow $\varrho_s/\sqrt{\zeta'_s}$ to be larger or smaller than the transmitter's intensity radius a_0 .

Unfortunately, higher-order moments of the turbulence are difficult to calculate. The fourth-order moment appears in our analysis in several places, but to find a closed-form expression that we can evaluate requires several additional assumptions, making the result less general. These assumptions and their consequences will be presented when needed.

2.3 Target Reflection

The surfaces of most real-world objects that we are interested in imaging are rough when considered on the scale of the illuminating field's wavelength [21]. When these

targets are illuminated with quasimonochromatic fields, the reflected light exhibits speckles in the far-field irradiance profile that is present at the bucket detector. These speckles arise from the rough-surface disrupting the phase front of light it reflects due to its microscopic height variations, as shown in Fig. 2-1. Propagation of the disrupted wave-front away from the rough surface then leads to the constructive and destructive interference that constitutes the speckle pattern seen at the detector. Following standard laser-radar theory [22], we will approximate the rough surface's field reflection coefficient by a complex-Gaussian random process¹ whose coherence length is approximately the wavelength of the illuminating field, i.e., $\varrho_T \sim \lambda_0$. For more details on this derivation see Appendix A.4.

The field that reaches the detector will have a coherence length $\sim \lambda_0/w_T$ and intensity radius $\sim \lambda_0/\varrho_T$, where w_T is the extent of the illuminated reflective surface. Because $\varrho_T \sim \lambda_0$ the reflected light will spread throughout the reflective hemisphere, i.e., the rough surface is a quasi-Lambertian scatterer. The strength of the reflected light is driven by the average reflectivity profile, which is what we want to image; this corresponds to what the human eye would see when looking at the target.

As suggested in Fig. 1-2, we consider a collection of K quasi-planar, rough surfaced targets arranged so that they do not occlude each other when viewed from either the transmitter or the detector, and are located at discrete ranges $L_0 < L_1 < \dots < L_{K-1}$ that are resolved by the pulse width τ_p . We also require that the targets' spatial features of interest are all larger than the on-target coherence length $\rho_{L_k} = 2L_k/k_0a_0$, so that they can each be spatially resolved as well.

We model each target at depth k by an underlying average intensity reflectivity profile $\mathcal{T}_k(\boldsymbol{\rho})$ so that the field reflectivity is

$$T_k(\boldsymbol{\rho}) = \sqrt{\mathcal{T}_k(\boldsymbol{\rho})}\vartheta_k(\boldsymbol{\rho}), \quad (2.14)$$

where $\vartheta_k(\boldsymbol{\rho})$ encapsulates the phase fluctuations induced by the rough surface, and as

¹As was the case for our transmitter, these Gaussian statistics are not valid if we are concerned with the field immediately after reflection. However, once we have reached the far field, this approximation yields results that are similar to those obtained with a more accurate field model.

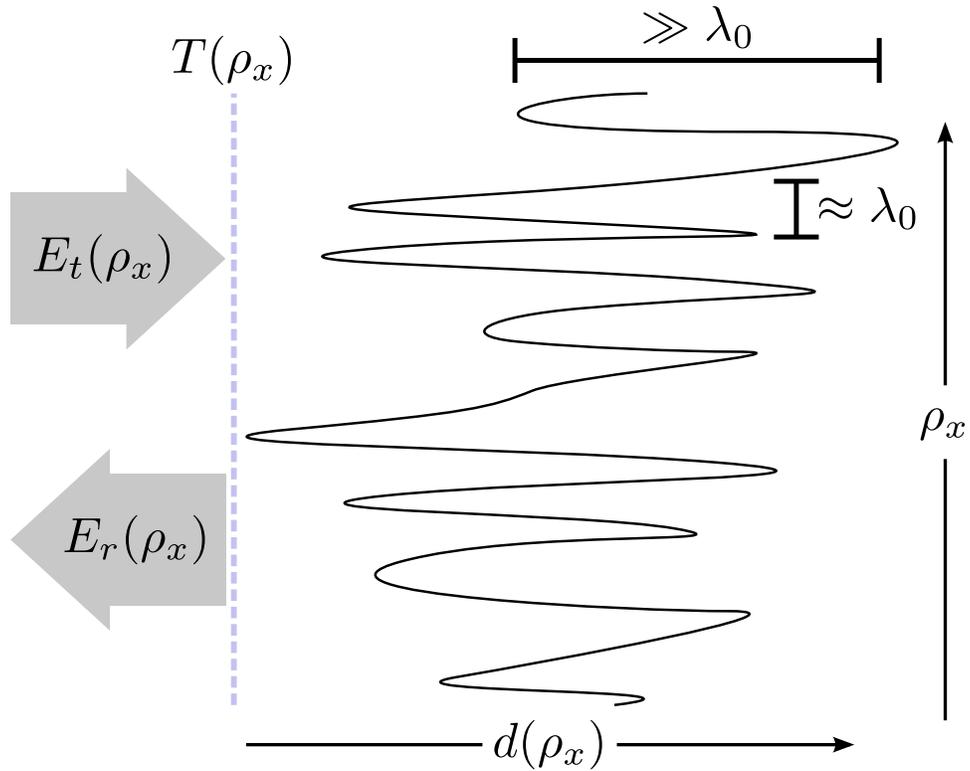


Figure 2-1: An example of a slice through a rough surface. Viewed in one transverse direction (ρ_x), the depth variations $d(\rho_x)$ from the reference plane (dotted gray line) are modeled as a Gaussian random process whose standard deviation greatly exceeds a wavelength of the incoming light ($\gg \lambda_0$), while its correlation length is on the order of λ_0 . The field reflectivity coefficient $T(\rho_x)$ can be regarded as occurring at the reference plane, where the reflected field $E_r(\rho_x)$ is transversely modulated by $\vartheta(\rho_x) = e^{i2k_0d(\rho_x)}$ relative to the incoming field $E_t(\rho_x)$. When we only need statistics about the reflected field after it has propagated into the far-field, we can approximate the effects of the phase modulation with a complex-Gaussian random process.

discussed, will be approximated by a zero-mean complex-Gaussian random process with coherence length ϱ_T . Since this coherence length is much smaller than any other feature of the target or illuminating field, for the purposes of our analysis the autocorrelation function of $\vartheta_k(\boldsymbol{\rho})$ becomes [22]

$$\langle \vartheta_k^*(\boldsymbol{\rho}_1) \vartheta_k(\boldsymbol{\rho}_2) \rangle = \pi \varrho_T^2 \delta(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2). \quad (2.15)$$

The reflected field leaving the rough surface at range L_k is then $E_s(\boldsymbol{\rho}, t, L_k) T_k(\boldsymbol{\rho})$.

2.4 Detection

There are two main types of photodetection systems used in LADAR applications: heterodyne-detection systems and direct-detection systems. Heterodyne detection beats the incoming target-return light on the surface of the photodetector with a strong local-oscillator field that is offset by a radio frequency (called the intermediate frequency, ω_{IF}) from ω_0 . The resulting intermediate-frequency photocurrent provides information about the complex envelope of the target-return component whose polarization and spatial mode match that of the local oscillator.

In contrast, direct detection yields a photocurrent that has a component proportional to the target-return power that is incident on the photodetector². Thus, because practical standoff structured illumination imaging may require its bucket detector to collect many target-return speckles, heterodyne detection systems are not a preferred option because they are fundamentally single-mode detectors unless multiple-element, multiple local oscillator configurations are employed. On the other hand, heterodyne detectors permit moving target indication via Doppler measurements that direct detection systems cannot. In going forward, we will limit our consideration to direct detection.

To make optimal use of the target-return field we need to isolate the portion of the photocurrent associated with range L_k . To do this we employ a matched filter,

²If a photon counting detector is used, then its count rate is proportional to the target-return power in the absence of extraneous (background) light.

with impulse response $g(t) = |p(-t)|^2$, on the detector's output, resulting in a filtered photocurrent from the single pixel bucket detector that is given by

$$i_b(t) = \int d\tau g(t - \tau) [q\eta P_b(\tau) + \Delta i_b(\tau)], \quad (2.16)$$

where

$$P_b(\tau) = \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \left| \sum_{k=0}^{K-1} E_r(\boldsymbol{\rho}, \tau, L_k) \right|^2 \quad (2.17)$$

is the received photon flux at time τ , η is the detector's quantum efficiency, q is the electron charge, \mathcal{A}_b is the field transmission of the bucket detector's active area, $\Delta i_b(\tau)$ is the zero-mean additive shot noise, and $E_r(\boldsymbol{\rho}, \tau, L_k)$ is the reflected field from range k , propagated back to the detector with Eq. (2.5). The shot noise arises from the Poisson process of semi-classical photodetection [23]; its covariance function is a signal-dependent white noise, given knowledge of the received photon flux, i.e.,

$$\langle \Delta i_b(\tau_1) \Delta i_b(\tau_2) | P_b \rangle = q^2 \eta \delta(\tau_1 - \tau_2) P_b(\tau_1). \quad (2.18)$$

We now discretize the filtered photocurrent by sampling at time delays that correspond to returns from range L for pulse n . The round trip time-of-flight for the photons returning from range L is $2L/c$. Adding the $n\tau_s$ delay for the generation of pulse n , the light returned from range L_k that is due to pulse n has delay

$$\tau_{n,k} = n\tau_s + 2L_k/c. \quad (2.19)$$

Thus our filtered photocurrent, after sampling, is reduced to a matrix of measurements, given by

$$i_b(n, k) = \int d\tau g(\tau_{n,k} - \tau) \left[q\eta \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \left| \sum_{k=0}^{K-1} E_r(\boldsymbol{\rho}, \tau, L_k) \right|^2 + \Delta i_b(\tau) \right]. \quad (2.20)$$

To simplify the discretization of the measurements, we note that for each pulse the

turbulence is essentially frozen over the duration of propagation, i.e. $\tau_\psi \gg 2L/c$ where $\tau_\psi \sim 1$ ms is the turbulence coherence time and $2L/c < 0.1$ ms for $L < 10$ km. This allows us to use the characteristic time $\tau_{n,k}$ to evaluate the turbulence for measurement (n, k) . Furthermore, since we are assuming that the pulses are both short enough and sufficiently spaced to avoid any overlap from different target ranges or different pulses during detection, viz.,

$$p(t - \tau_{n,k})p(t - \tau_{n',k'}) = \delta_{nn'}\delta_{kk'}|p(t - \tau_{n,k})|^2, \quad (2.21)$$

the photon flux density in Eq. (2.20) simplifies as follows

$$\begin{aligned} & \left| \sum_{k=0}^{K-1} E_r(\boldsymbol{\rho}, \tau, L_k) \right|^2 \\ &= \left| \int d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau) \sum_{k=0}^{K-1} T_k(\boldsymbol{\rho}') \int d\boldsymbol{\rho}'' h_s(\boldsymbol{\rho}', \boldsymbol{\rho}'', \tau - L_k/c) E_s(\boldsymbol{\rho}'', \tau - 2L_k/c) \right|^2 \\ &= \mathbb{N}_s \left| \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} \int d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau) T_k(\boldsymbol{\rho}') \int d\boldsymbol{\rho}'' h_s(\boldsymbol{\rho}', \boldsymbol{\rho}'', \tau - L_k/c) \xi(\boldsymbol{\rho}'', n) p(\tau - \tau_{n,k}) \right|^2 \\ &= \mathbb{N}_s \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} |p(\tau - \tau_{n,k})|^2 \left| \int d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_{n,k}) T_k(\boldsymbol{\rho}') \xi_t(\boldsymbol{\rho}', n, k) \right|^2 \\ &= \mathbb{N}_s \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} |p(\tau - \tau_{n,k})|^2 |\xi_b(\boldsymbol{\rho}, n, k)|^2. \end{aligned} \quad (2.22)$$

Here $\xi_t(\boldsymbol{\rho}', n, k)$ and $\xi_b(\boldsymbol{\rho}, n, k)$ are the transverse field profiles at the target and detector associated with the return from target k and pulse n ,

$$\xi_t(\boldsymbol{\rho}', n, k) = \int d\boldsymbol{\rho}'' h_s(\boldsymbol{\rho}', \boldsymbol{\rho}'', \tau_{n,k} - L_k/c) \xi(\boldsymbol{\rho}'', n) \quad (2.23)$$

$$\xi_b(\boldsymbol{\rho}, n, k) = \int d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_{n,k}) T_k(\boldsymbol{\rho}') \xi_t(\boldsymbol{\rho}', n, k). \quad (2.24)$$

The reader will note that in Eq. (2.22) the temporal and spatial nature of the detected light factor, so the matched filter only discriminates between pulses, scales the measurement, and provides some shot-noise averaging. Defining this scaling factor

and the integrated shot noise as

$$\tau'_p = \left(\int d\tau |p(\tau)|^4 \right)^{-1} \quad (2.25)$$

$$\Delta i_b(n, k) = \int d\tau |p(\tau - \tau_{n,k})|^2 \Delta i_b(\tau), \quad (2.26)$$

and subsequently the total number of photons arriving from target k 's illumination by pulse n as

$$\mathbb{N}_b(n, k) = \mathbb{N}_s \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} |\xi_b(\boldsymbol{\rho}, n, k)|^2, \quad (2.27)$$

our measurement matrix can be written as

$$i_b(n, k) = \frac{q\eta}{\tau'_p} \mathbb{N}_b(n, k) + \Delta i_b(n, k). \quad (2.28)$$

Given knowledge of $\mathbb{N}_b(n, k)$ the noise terms $\{\Delta i_b(n, k)\}$ are zero-mean, uncorrelated with each other, and have variance

$$\langle \Delta^2 i_b(n, k) | \mathbb{N}_b(n, k) \rangle = q^2 \eta \mathbb{N}_b(n, k) / \tau_p''^2 \quad (2.29)$$

where $\tau_p'' = (\int dt |p(t)|^6)^{-1/2}$.

2.5 Summary

To conclude our framework development, we will summarize and discuss our fundamental assumptions and approximations, section by section.

Transmitter

We assume a periodically-pulsed, structured-illumination transmitter whose output consists of a sequence or random spatial patterns that modulate a train of identical temporal pulses. These spatial patterns are modeled as independent, identically distributed, complex-Gaussian random process with a Gaussian-Schell model covariance

function having intensity radius a_0 much greater than its coherence length ρ_0 . The temporal pulses have duration τ_p that is much shorter than the pulse train's pulse repetition period τ_s .

Propagation

Propagation from the transmitter to the target and from the target to the bucket detector is assumed to be through atmospheric turbulence. The path lengths involved are assumed to satisfy the far-field condition for a partially-coherent transmitter, i.e., $k_0^2 a_0 \rho_0 / 2L \ll 1$, where k_0 is the transmitter's wavenumber and L is the range to the target. We note that this far-field condition is far less stringent than the corresponding far-field condition for a fully coherent transmitter, namely $k_0^2 a_0^2 / 2L \ll 1$.

The difference in these far-field conditions can be interpreted as follows. The partially-coherent far-field condition applies to the propagation of the field's *coherence function* [24, 25]. Once we are in this regime the field is again modeled as a complex-Gaussian random process following the same Gaussian-Schell model, but with a new coherence length ρ_L and intensity radius a_L given by $\rho_L = 2L/k_0 a_0$, and $a_L = 2L/k_0 \rho_0$. Put another way, there is now a Fourier transform relationship between the coherence functions at the two planes, but not necessarily between the fields themselves. Once in the more restrictive far-field condition associated with a coherent transmitter, there is also a Fourier transform relationship between the fields at the two planes.

For each propagation path—transmitter to target and target to detector—we use the extended Huygens-Fresnel principle to characterize the combined effects of diffraction and turbulence. Our far-field assumption simplifies these propagation kernels to

$$h_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t) = \frac{k_0 e^{ik_0(L + |\boldsymbol{\rho}|^2/2L - \boldsymbol{\rho} \cdot \boldsymbol{\rho}'/L)} e^{\psi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)}}{i2\pi L}. \quad (2.30)$$

Many of our propagation-statistics calculations require only the mutual coherence function of this kernel, and so for analytic simplicity we employ the square-law

approximation to the 5/3-law behavior of Kolmogorov-spectrum turbulence. This approximation is widely used in ghost-imaging analyses [3–5, 14, 26, 27], but care must be taken because it does not extend to higher-order moments.

Target

The target is taken to be a constellation of quasi-planar, quasi-Lambertian rough surfaces located at discrete ranges that will be resolved by the transmitter’s pulse duration, i.e., the minimum distance between two planes, $\min_{0 \leq k \leq K-1} (L_k - L_{k-1})$, is greater than $c\tau_p/2$. We also assume the overall depth (range spread) of the targets is small enough to preclude ambiguities between returns from successive transmitted pulses. This implies that the total depth spread, $L_{K-1} - L_0$, is less than $c\tau_s/2$.

Photodetection

We assume signal shot-noise limited photodetection with post-detection matched filtering followed by sampling at range delays associated with target components. Thus, rather than develop a full three-dimensional image of the target, we process data one range-slice at a time. As a result, in the remainder of the thesis we will drop the index used in this chapter to denote the target’s range component.

Chapter 3

Ghost Imaging

In this chapter we apply the Ch. 2 framework to develop computational ghost imaging in reflection through atmospheric turbulence, much of which has been published in [5]. This comprises an analysis of the resolution and signal-to-noise ratio of the correlation-generated ghost image, as well as a comparison to a standard LADAR system. We then review recent advances in ghost imaging, in which alternatives to correlation-based processing are employed. First, however, we discuss the history of ghost imaging, and its connection to our current exploration of standoff structured-illumination imaging.

The field of ghost imaging arose from experiments [7] that used beams of quantum-entangled signal and idler photons in the imaging configuration shown in Fig. 3-1. The signal and idler were generated by spontaneous parametric downconversion (SPDC) and separated, for type-II phase-matched SPDC, using a polarizing beam splitter after which they propagated over independent optical paths of the same length. Idler photon arrivals were measured by a spatially resolving photon-counting detector, while signal photons passed through an object transparency before their arrivals were measured by a bucket detector located behind the mask. When a signal photon is measured by the bucket detector, that detection provides no information about the spatial location at which it passed through the mask and hence reveals nothing about the mask's spatial pattern. Conversely, idler photons are detected in a spatially-resolved manner, but they did not encounter the mask, so they so they provide no

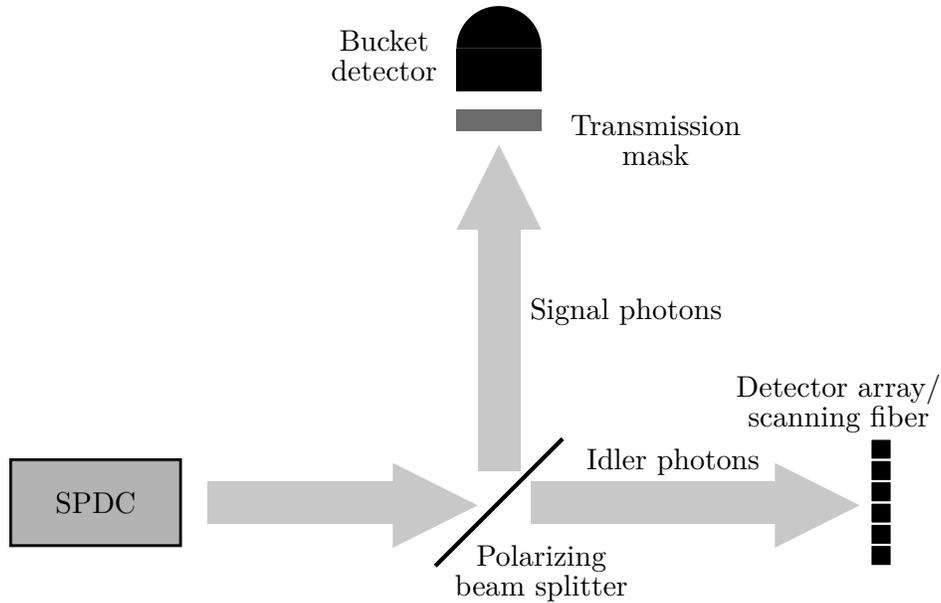


Figure 3-1: The setup for the initial transmissive ghost imaging experiments. Entangled signal and idler photons are generated by SPDC and separated by a polarizing beam splitter. The signal photons pass through a transmission mask, and a bucket detector records their arrival times. A spatially-resolving detector (a scanning pinhole fiber in the initial experiments [7]) records photon arrival times on the idler arm, and signal-idler coincidences are accumulated for every pixel to image the transmission profile of the mask.

information about the mask’s spatial pattern.

However, each photon in the signal beam produced by SPDC is entangled in momentum with a companion idler photon. Therefore, detection of an idler photon at a particular transverse location tells us the transverse location of its associated signal photon as it impinged on the transmission mask. It follows that the average number of signal-idler coincidence detections between the bucket detector and a particular idler location is proportional to the mask’s intensity transmission at that location. In this way we can build the transmission profile of an object without ever making a spatial measurement of a photon that directly interacted with that object. Because neither the signal nor idler measurements alone could image the mask, but that object’s transmission pattern emerged from the correlation of the two measurements, the term “ghost imaging” was applied to the system.

Ghost imaging was originally interpreted as a quantum-mechanical effect arising

from the entanglement of the signal and idler photons, so that non-local two photon interference was the only explanation for the image. Subsequent experiments [8–10] using classical light sources cast doubt on this interpretation, as they indicated that classical correlation between the two measurements was sufficient to form a ghost image. A complete Gaussian-state analysis, which provided a unified framework for both entangled light (produced by SPDC) and pseudothermal light (produced by passing a laser beam through a rotating ground-glass diffuser), showed that the downconverter approach produced better contrast and near-field resolution, and that quantum and semiclassical treatments produced quantitatively identical results for the pseudothermal source [11, 12, 28].

As further evidence that ghost imaging does not require entanglement, a theory of computational ghost imaging was developed [18]. Here the signal field is deterministically created by shining a laser on a spatial light modulator (SLM) whose pixels impart a sequence of pseudorandom phase shifts on the reflected light. Because the phase-shift sequence is known, no reference-arm measurement is needed to determine the on-target irradiance patterns; they can now be calculated from diffraction theory¹. Since only one light field is used to form the computational ghost image, there is no entanglement; this approach we later experimentally verified [31].

This computational approach shows that ghost imaging is simply a form of structured illumination imaging. We create a sequence of pseudo-randomly generated fields that illuminate the target, and the reflectivity pattern is inferred by correlating these patterns with the bucket detector measurement for each transverse location on the target. Highly transmissive areas of the target will be more strongly correlated with the bucket detector measurements than more opaque areas, and this is all that the ghost image is measuring. This understanding opened the door to alternative methods of image formation from structured-illumination measurements [32].

Initial work—both theory and experiments—on ghost imaging presumed a transmissive geometry, in which bucket detector sits behind the target and collects the

¹The SLM approach was also adapted to permit experiments that used classical phase-sensitive correlation to form the ghost image [29, 30]. Previous pseudothermal experiments had always relied on phase-insensitive correlations.

transmitted light. Later table-top demonstrations showed that ghost imaging could be performed in reflection [33, 34], where the bucket detector faces the target and collects reflected light. Those experiments created interest in using ghost imaging for standoff sensing. A comprehensive framework for long-range reflective ghost imaging through atmospheric turbulence was developed [3–5, 27], much in the spirit of the original transmissive analysis, and recent experiments have demonstrated long range reflective ghost imaging in a real-world setting [35].

3.1 Imaging Framework

For this analysis we use the configuration shown in Fig. 3-2, in which a partially coherent source radiates a series of pulses through atmospheric turbulence to a target that is standing kilometers away. Each pulse reflects off the surface and propagates back to the detector which, following our model from Ch. 2, generates measurements that are driven by the number of detected photons per pulse.

As was shown in [5], reflective ghost imaging can be used for three-dimensional imaging by sampling the bucket-detector’s photo-current at time delays associated with reflected light from different ranges, and forming a ghost image for each range independently. This requires that the reference field be recalculated for each range, and some additional assumptions about the pulse-repetition period and the range separation between different target components. The Ch. 2 framework incorporates these assumptions, so without loss of generality we will limit our consideration to a ghost image formed for a target at the single range L . This does not preclude simultaneously creating ghost images for other ranges, but dropping all explicit references to the multiple ranges will simplify the analysis and be more amenable to extension later in the thesis.

Given the bucket detector measurements defined in Eq. (2.28), the computational ghost image is formed from the covariance between this sequence of measurements and the corresponding computed on-target irradiance patterns. These computed patterns

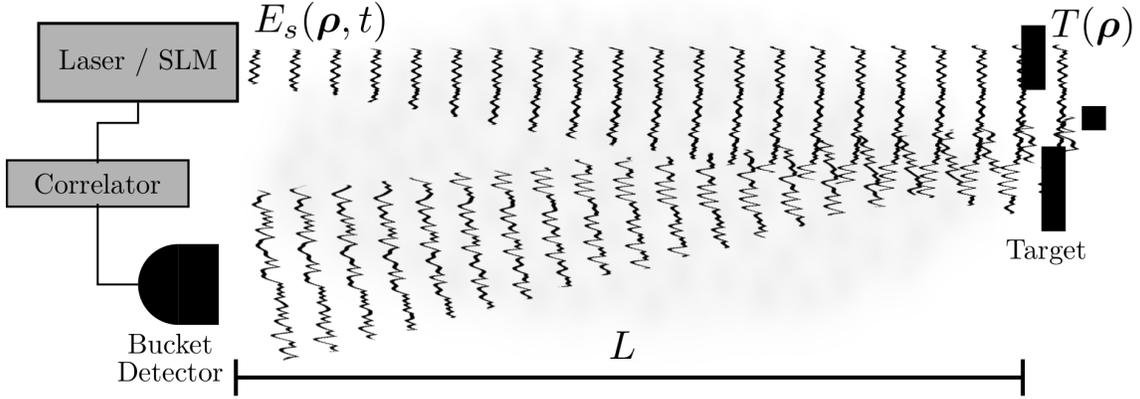


Figure 3-2: Setup for a reflective ghost imager. A laser and SLM are used to generate a series of partially-coherent pulses that propagate through atmospheric turbulence. The pulses reflect off an extended-range target, and some of the reflected light reaches a bucket detector that is facing the scene. Bucket detector measurements are then correlated with the on-target irradiance patterns (computed from the known SLM phase patterns) to form the image.

are

$$I(\boldsymbol{\rho}, n) = \left| \int d\boldsymbol{\rho}' \xi(\boldsymbol{\rho}', n) \frac{k_0 e^{ik_0|\boldsymbol{\rho}-\boldsymbol{\rho}'|^2/2L}}{i2\pi L} \right|^2, \quad (3.1)$$

where for analytic convenience we have omitted some constant factors that would appear if a physical reference was used. This scaling will not affect our analysis of the spatial resolution and signal-to-noise ratio. The N -pulse ghost image at location $\boldsymbol{\rho}$ is the empirical covariance of the sequence of reference pattern values at this location and the sequence of bucket detector measurements, viz.,

$$\mathcal{G}_N(\boldsymbol{\rho}) = \frac{1}{N-1} \sum_{n=0}^{N-1} \tilde{i}_b(n) \Delta I(\boldsymbol{\rho}, n), \quad (3.2)$$

where

$$\tilde{i}_b(n) \equiv i_b(n) - \frac{1}{N} \sum_{n=0}^{N-1} i_b(n) \quad (3.3)$$

is the empirically centered bucket detector measurements for pulse n , and for the

reference pattern we use the transmitter statistics to calculate

$$\Delta I(\boldsymbol{\rho}, n) \equiv I(\boldsymbol{\rho}, n) - \langle I(\boldsymbol{\rho}, n) \rangle. \quad (3.4)$$

Note that the original entangled-photon ghost imaging experiment formed an image from correlation, not covariance [7]. Because this experiment's signal and reference beams were obtained from SPDC in the low-flux regime, a background-free ghost image was obtained. Later, when pseudothermal light was used to form ghost images via correlation, it was found that the correlation-image rode on a strong, featureless background. However, high contrast images were obtained by correlating the measurements with the statistically centered irradiance fluctuations [8]. The Gaussian-state analyses in [4, 11, 12] showed that for a continuous-wave transmitter a DC-block was needed for a high contrast (low background) image. We are considering a pulsed source, and therefore to achieve a high contrast image we are preemptively removing the mean values from both $I(\boldsymbol{\rho}, n)$ and $i_b(n)$, making our ghost image the covariance between the measurements and the reference patterns.

3.2 Spatial Resolution

The spatial resolution of a ghost image is found by evaluating the ensemble average of Eq. (3.2). Since the spatial patterns $\{\xi(\boldsymbol{\rho}, n)\}$ are a sequence of statistically independent, identically distributed (iid) random processes, this simplifies to

$$\langle \mathcal{G}_N(\boldsymbol{\rho}) \rangle = \langle i_b(n) I(\boldsymbol{\rho}, n) \rangle - \langle i_b(n) \rangle \langle I(\boldsymbol{\rho}, n) \rangle. \quad (3.5)$$

The mean value of the reference irradiance pattern is

$$\begin{aligned} \langle I(\boldsymbol{\rho}, n) \rangle &= \int d\boldsymbol{\rho}_1 \int d\boldsymbol{\rho}_2 \langle \xi(\boldsymbol{\rho}_1, n) \xi^*(\boldsymbol{\rho}_2, n) \rangle \frac{k_0^2 e^{ik_0(|\boldsymbol{\rho}_1|^2 - |\boldsymbol{\rho}_2|^2 - 2\boldsymbol{\rho} \cdot (\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2)) / 2L}}{(2\pi L)^2} \\ &\approx \frac{2}{\pi a_L^2} e^{-2|\boldsymbol{\rho}|^2 / a_L^2}, \end{aligned} \quad (3.6)$$

where $a_L = \lambda_0 L / \pi \rho_0$ is the intensity radius of the field at range L , and the approximation in the second line follows from our far-field assumption $k_0^2 a_0 \rho_0 / 2L \ll 1$, and the partial-coherence condition $\rho_0 \ll a_0$.

To compute the mean of the bucket detector measurement we first back-propagate $\xi_b(\boldsymbol{\rho}, n)$ to $\xi_t(\boldsymbol{\rho}', n)$ with Eq. (2.24), so that

$$\begin{aligned}
\langle i_b(n) \rangle &= \frac{q\eta\mathbb{N}_s}{\tau_p'} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \langle |\xi_b(\boldsymbol{\rho}, n)|^2 \rangle \\
&= \frac{q\eta\mathbb{N}_s}{\tau_p'} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \int d\boldsymbol{\rho}'_1 \int d\boldsymbol{\rho}'_2 \langle \xi_t(\boldsymbol{\rho}'_1, n) \xi_t^*(\boldsymbol{\rho}'_2, n) \rangle \\
&\quad \times \langle T(\boldsymbol{\rho}'_1) T^*(\boldsymbol{\rho}'_2) \rangle \langle h_r(\boldsymbol{\rho}, \boldsymbol{\rho}'_1, \tau_n) h_r^*(\boldsymbol{\rho}, \boldsymbol{\rho}'_2, \tau_n) \rangle \\
&= \frac{q\eta\mathbb{N}_s}{\tau_p' L^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \int d\boldsymbol{\rho}'_1 \mathcal{T}(\boldsymbol{\rho}'_1) \langle \xi_t(\boldsymbol{\rho}'_1, n) \xi_t^*(\boldsymbol{\rho}'_1, n) \rangle, \tag{3.7}
\end{aligned}$$

where we have used the statistical independence of the transmitter's spatial patterns, turbulence, and the target's surface roughness to separate the averages. The surface-roughness correlation was evaluated with Eq. (2.15) for $\rho_T^2 = \lambda_0^2 / \pi$, a standard assumption² in laser radar theory [22]. We evaluate the remaining correlation of the on-target spatial patterns by back-propagating $\xi_t(\boldsymbol{\rho}', n)$ to the transmitter with Eq. (2.23), so that

$$\begin{aligned}
&\langle \xi_t(\boldsymbol{\rho}'_1, n) \xi_t^*(\boldsymbol{\rho}'_1, n) \rangle \\
&= \int d\boldsymbol{\rho}''_1 \int d\boldsymbol{\rho}''_2 \langle \xi_s(\boldsymbol{\rho}''_1, n) \xi_s^*(\boldsymbol{\rho}''_2, n) \rangle \langle h_s(\boldsymbol{\rho}'_1, \boldsymbol{\rho}''_1, \tau_n - L/c) h_s^*(\boldsymbol{\rho}'_1, \boldsymbol{\rho}''_2, \tau_n - L/c) \rangle \\
&= \frac{2}{\pi a_L^2} e^{-2|\boldsymbol{\rho}'_1|^2 / a_L^2}, \tag{3.8}
\end{aligned}$$

where we have again employed our far-field assumption, as well as the partial coherence assumption that $\rho_0 \ll \min(a_0, \varrho_s / \sqrt{c'_s})$. The expected value of the measurement

²While the correlation length will not be exactly known, it can be taken to be $\sim \lambda_0$ because smaller-scale features produce evanescent, non-propagating waves. This assertion that $\rho_T^2 = \lambda_0^2 / \pi$ simplifies the analysis without affecting the resolution or SNR analysis.

is now

$$\langle i_b(n) \rangle = \frac{2q\eta\mathbb{N}_s A_b}{\pi a_L^2 \tau_p' L^2} \int d\boldsymbol{\rho}' \mathcal{T}(\boldsymbol{\rho}') e^{-2|\boldsymbol{\rho}'|^2/a_L^2}, \quad (3.9)$$

where we have defined $A_b \equiv \int d\boldsymbol{\rho} \mathcal{A}_b^2(\boldsymbol{\rho})$ to be the detector's photosensitive area. Finally, we will assume the on-target intensity radius a_L is sufficiently large that the entire target is uniformly illuminated on average, which reduces Eq. (3.9) to

$$\langle i_b(n) \rangle = \frac{2q\eta\mathbb{N}_s A_b}{\pi a_L^2 \tau_p' L^2} \int d\boldsymbol{\rho}' \mathcal{T}(\boldsymbol{\rho}'). \quad (3.10)$$

We now turn to computing the expectation of the signal bearing term in Eq. (3.5), the cross-correlation of the reference patterns and bucket measurements. We again use Eqs. (2.23) and (2.24) to back-propagate the field patterns, explicitly exposing the transmitter field pattern $\xi(\boldsymbol{\rho}'', n)$, the target reflection pattern $T(\boldsymbol{\rho}')$, and the turbulence kernels $\psi_s(\boldsymbol{\rho}', \boldsymbol{\rho}'', \tau_n)$ and $\psi_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_n)$. We then use the independence of these processes to separate their evaluations. This becomes a fourth-order moment of the field and second-order moments of the target surface and the propagation kernels. The fourth-order field moment can be written in terms of the second-order moments in Eq. (2.2) and Eq. (2.3) by using the Gaussian moment factoring theorem³, which we detail in App. B.2.

Employing the preceding assumptions, the mean ghost image term resolves to

$$\langle \mathcal{G}_N(\boldsymbol{\rho}) \rangle = \frac{4q\eta\mathbb{N}_s A_b \rho_L^2}{\pi a_L^4 \tau_p' L^2} \int d\boldsymbol{\rho}' \mathcal{T}(\boldsymbol{\rho}') \frac{e^{-|\boldsymbol{\rho}-\boldsymbol{\rho}'|^2/\alpha\rho_L^2}}{\pi\alpha\rho_L^2}, \quad (3.11)$$

where $\rho_L \equiv \lambda_0 L / \pi a_0$ is the range- L coherence length of the transmitted field and

$$\alpha = 1 + \frac{a_0^2 \zeta_s'}{2\varrho_s^2} \quad (3.12)$$

is the resolution-degradation factor imposed by the turbulence on the path from the transmitter to the target.

³The factorization of this fourth-order moment of the field is presented as an example in Eq. (B.8).

From Eq. (3.11) we see that the ensemble average image for range L is the average intensity reflectivity of the target convolved with a Gaussian point-spread function (PSF) with width $\sqrt{\alpha}\rho_L$ that sets the limit on feature sizes resolvable by the ghost imager. In the absence of turbulence, $\alpha = 1$, and the best-case spatial resolution is $\sqrt{\rho_L}$. Notably, the turbulence on the return path (from the target to the detector) has no effect on the spatial resolution. Furthermore, the turbulence on the transmitter-to-target path only affects the resolution when $\varrho_s/\sqrt{\zeta'_s}$ becomes smaller than the intensity radius of the source. Therefore, if the source is placed in a location where there is weak turbulence, we do not expect turbulence will affect the spatial resolution at all.

3.3 Signal-to-Noise Ratio

We defined the signal-to-noise ratio (SNR) as the ratio of the squared mean signal to its variance, viz.

$$\text{SNR}_{\mathcal{G}}(\boldsymbol{\rho}) = \frac{\langle \mathcal{G}_N(\boldsymbol{\rho}) \rangle^2}{\text{var}[\mathcal{G}_N(\boldsymbol{\rho})]} = \frac{\langle \mathcal{G}_N(\boldsymbol{\rho}) \rangle^2}{\langle \mathcal{G}_N^2(\boldsymbol{\rho}) \rangle - \langle \mathcal{G}_N(\boldsymbol{\rho}) \rangle^2}. \quad (3.13)$$

To simplify the its evaluation we will assume that the features on the target are large enough that they are resolved by the imager, making $\int d\boldsymbol{\rho}' \mathcal{T}(\boldsymbol{\rho}') e^{-|\boldsymbol{\rho}' - \boldsymbol{\rho}|^2/\alpha\rho_L^2} \approx \pi\alpha\rho_L^2 \mathcal{T}(\boldsymbol{\rho})$. With this assumption the square of the mean signal in Eq. (3.11) is

$$\langle \mathcal{G}_N(\boldsymbol{\rho}) \rangle^2 = \left[\frac{4q\eta\mathbb{N}_s A_b \rho_L^2}{\pi a_L^4 \tau'_p L^2} \mathcal{T}(\boldsymbol{\rho}) \right]^2. \quad (3.14)$$

This leaves the rather tedious task of evaluating $\langle \mathcal{G}_N^2(\boldsymbol{\rho}) \rangle$, the second moment of the ghost image, viz.,

$$\langle \mathcal{G}_N^2(\boldsymbol{\rho}) \rangle = \frac{1}{(N-1)^2} \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} \langle \tilde{i}_b(n) \Delta I(\boldsymbol{\rho}, n) \tilde{i}_b(m) \Delta I(\boldsymbol{\rho}, m) \rangle. \quad (3.15)$$

If we expand $\tilde{i}_b(n)$ and $\Delta I(\boldsymbol{\rho}, n)$ with Eq. (3.3) and Eq. (3.4) this becomes the sum of the averages of sixteen different terms. We use Eq. (2.29) to evaluate the variance

of the shot noise, and the remaining terms can be processed with the same method we used to evaluate the first moment, i.e., we use Eqs. (2.23) and (2.24) to back-propagate the fields, explicitly exposing the transmitter field patterns, turbulence kernels, and the target's field-reflection coefficient. These fluctuations are statistically independent, so we can separate their averages.

The above decomposition results in the following new statistical averages: eighth-order and sixth-order moments of the transmitter field patterns $\{\xi(\boldsymbol{\rho}, n)\}$, fourth-order moments of the turbulence fluctuations incurred from the propagation to and from the target, and a fourth-order moment of the target's rough surface. The averages over the field patterns and rough surface can again be rewritten in terms of their second-order moments via the Gaussian moment factoring theorem in App. B.2. However this greatly increases the complexity of the derivation, as an eighth-order moment becomes the sum of 105 terms. Fortunately, these calculations are greatly simplified by the field having no phase-sensitive correlation (as shown in Eq. (2.2)) and symmetry in the phase-insensitive correlation in Eq. (2.3). Evaluation of the fourth-order moment of the turbulence, however, is more complicated.

Fourth-order Turbulence Moments

Within the region of validity of the Rytov approximation, the turbulence log-amplitude and phase fluctuations ($\chi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ and $\phi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$) that appear in the extended Huygens-Fresnel principle are jointly Gaussian random processes [23]. Although this permits a moment-factoring approach to obtaining the fourth-order moment we need to complete our SNR calculation, we cannot do so using the square-law approximation that is convenient for second-order moment calculations. This is because use of the square-law approximation in conjunction with $\chi(\cdot)$ and $\phi(\cdot)$ being jointly Gaussian implies that the turbulence fluctuations in the input and output planes are piston phase shifts plus phase tilts, which is not the case in the atmosphere.

Our approach is then to assume that in Eq. (2.8) the strength-profile $C_{n,m}^2(z)$ is of weak-to-moderate strength or concentrated near the target. This allows us to ignore the coordinate dependence of the turbulence in the transmitter and detector

pupils, and to assume that the log-amplitude coherence length at range L is larger than the field-pattern coherence length ρ_L . In essence, this means that the turbulence will not significantly impact the imager's spatial resolution, and that its fourth-order moments will reduce to those of amplitude fluctuations at the target and detector. This approximation will only be used when evaluating the ghost imager's SNR.

The delta-function nature of the target's coherence function plus the assumptions made in the preceding paragraph reduce the required turbulence fourth-order moments to

$$\langle e^{\psi_s(\boldsymbol{\rho}_1, \boldsymbol{\rho}'_1, \tau_n)} e^{\psi_s^*(\boldsymbol{\rho}_1, \boldsymbol{\rho}'_2, \tau_n)} e^{\psi_s(\boldsymbol{\rho}_1, \boldsymbol{\rho}'_3, \tau_{n'})} e^{\psi_s^*(\boldsymbol{\rho}_1, \boldsymbol{\rho}'_4, \tau_{n'})} \rangle \quad (3.16)$$

$$\langle e^{\psi_r(\boldsymbol{\rho}'_1, \boldsymbol{\rho}_1, \tau_n)} e^{\psi_r^*(\boldsymbol{\rho}'_2, \boldsymbol{\rho}_1, \tau_n)} e^{\psi_r(\boldsymbol{\rho}'_3, \boldsymbol{\rho}_1, \tau_{n'})} e^{\psi_r^*(\boldsymbol{\rho}'_4, \boldsymbol{\rho}_1, \tau_{n'})} \rangle. \quad (3.17)$$

Ignoring the coordinate dependence at the transmitter and detector planes amounts to setting $\boldsymbol{\rho}'_1 = \boldsymbol{\rho}'_2 = \boldsymbol{\rho}'_3 = \boldsymbol{\rho}'_4 = \mathbf{0}$ in Eqs. (3.16) and (3.17), where $\mathbf{0}$ denotes the centers of the transmitter and detector pupils. Applying these substitutions and expanding the turbulence exponent as the sum of its phase and log-amplitude components, we see that the phase terms now cancel, leaving

$$\langle e^{\psi_s(\boldsymbol{\rho}_1, \mathbf{0}, \tau_n)} e^{\psi_s^*(\boldsymbol{\rho}_1, \mathbf{0}, \tau_n)} e^{\psi_s(\boldsymbol{\rho}_1, \mathbf{0}, \tau_{n'})} e^{\psi_s^*(\boldsymbol{\rho}_1, \mathbf{0}, \tau_{n'})} \rangle = \langle e^{2\chi_s(\boldsymbol{\rho}_1, \mathbf{0}, \tau_n) + 2\chi_s(\boldsymbol{\rho}_1, \mathbf{0}, \tau_{n'})} \rangle \quad (3.18)$$

$$\langle e^{\psi_r(\mathbf{0}, \boldsymbol{\rho}_1, \tau_n)} e^{\psi_r^*(\mathbf{0}, \boldsymbol{\rho}_1, \tau_n)} e^{\psi_r(\mathbf{0}, \boldsymbol{\rho}_1, \tau_{n'})} e^{\psi_r^*(\mathbf{0}, \boldsymbol{\rho}_1, \tau_{n'})} \rangle = \langle e^{2\chi_r(\mathbf{0}, \boldsymbol{\rho}_1, \tau_n) + 2\chi_r(\mathbf{0}, \boldsymbol{\rho}_1, \tau_{n'})} \rangle. \quad (3.19)$$

Because $\chi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ is Gaussian, these correlation functions reduce to

$$\langle e^{2\chi_r(\mathbf{0}, \boldsymbol{\rho}_1, \tau_n) + 2\chi_r(\mathbf{0}, \boldsymbol{\rho}_1, \tau_{n'})} \rangle = e^{4\sigma_r^2 K(\tau_s(n-n'))} \quad (3.20)$$

$$\langle e^{2\chi_s(\boldsymbol{\rho}_1, \mathbf{0}, \tau_n) + 2\chi_s(\boldsymbol{\rho}_1, \mathbf{0}, \tau_{n'})} \rangle = e^{4\sigma_s^2 K(\tau_s(n-n'))}, \quad (3.21)$$

where

$$\sigma_m^2 = 0.562 k_0^{7/6} \int_0^L dz C_{n,m}^2(z) \left[\frac{z(L-z)}{L} \right]^{5/6} \quad (3.22)$$

is the Rytov-approximation variance [23] of the log-amplitude fluctuations on path m ,

and $K(\cdot)$ is the normalized time correlation of these fluctuations, such that $K(0) = 1$.

The SNR of the reconstruction will be affected by the number of turbulence coherence times spanned during the image acquisition time. To simplify our final results we define an additional term to capture this time-varying impact of the turbulence,

$$\Psi = \frac{1}{N} \sum_{n=0}^{N-1} \left[e^{4\sigma_s^2 K(n\tau_s) + 4\sigma_r^2 K(n\tau_s)} - 1 \right]. \quad (3.23)$$

When the turbulence decorrelates pulse-to-pulse, so that $K(n\tau_s) = \delta_{n,0}$, we get $\Psi = (e^{4(\sigma_s^2 + \sigma_r^2)} - 1)/N$. When the turbulence coherence time become much longer than the acquisition time we approach the other asymptote, for which $\Psi = e^{4(\sigma_s^2 + \sigma_r^2)} - 1$.

Origins of Noise in the Signal-to-Noise Ratio

To complete the SNR calculation we need to more precisely define a few aspects of the system. Although most photodetectors have rectangular or circular indicator-function pupils for their photosensitive regions, we will assume our detector has a Gaussian pupil function. This results in closed-form expressions that more clearly expose the dependence on various parameters (such as the detector radius) while giving qualitatively similar behavior to what would be found with an indicator-function pupil. We still take the detector area to be $A_b = \int d\boldsymbol{\rho} \mathcal{A}_b^2(\boldsymbol{\rho})$, so the field-transmission pupil for the bucket detector is $\mathcal{A}_b(\boldsymbol{\rho}) = e^{-|\boldsymbol{\rho}|^2/r_b^2}$, where r_b is the radius of the circular detector we are approximating.

To clarify the presentation we also defined $A_T = \int d\boldsymbol{\rho} \mathcal{T}(\boldsymbol{\rho})$ and $A'_T = \int d\boldsymbol{\rho} \mathcal{T}^2(\boldsymbol{\rho})$, which are two measures of the total reflective area of target. We also define the ratio of the area of the bucket detector to the area of the transmitter as $\beta = r_b^2/a_0^2$. The SNR for the ghost image at coordinate $\boldsymbol{\rho}$ and range L is then [5]

$$\text{SNR}_{\mathcal{G}_N}(\boldsymbol{\rho}) = \frac{\mathcal{T}^2(\boldsymbol{\rho})}{\Delta^2 \mathcal{S} + \Delta^2 \mathcal{R} + \Delta^2 \mathcal{D} + \Delta^2 \mathcal{F}} \quad (3.24)$$

where the terms in the noise denominator break down as follows:

$$\Delta^2 \mathcal{S} = \frac{A'_T(1 + \beta^{-1})}{\pi \rho_L^2 N} e^{4(\sigma_s^2 + \sigma_r^2)} \quad (3.25)$$

$$\Delta^2 \mathcal{R} = \mathcal{T}^2(\boldsymbol{\rho}) \frac{1 + 2(1 + \beta)\Psi}{1 + 2\beta} \quad (3.26)$$

$$\Delta^2 \mathcal{D} = \frac{A_T a_L^2 L^2}{2A_b \pi \eta \mathbb{N}_s N \rho_L^4} \quad (3.27)$$

$$\Delta^2 \mathcal{F} = \frac{A_T^2 (e^{4(\sigma_s^2 + \sigma_r^2)} - 1 - \Psi)}{\pi^2 N \rho_L^4}. \quad (3.28)$$

These noise terms account for randomness inherent in the ghost image. The first term, $\Delta^2 \mathcal{S}$, is the noise produced by transmitter's randomness. It is directly proportional to the number of on-target resolution cells, $A'_T/\pi \rho_L^2$, and inversely proportional to the number of pulses N . Unless we employ more pulses than there are resolution cells, this term can limit the SNR. In essence, $\Delta^2 \mathcal{S}$ sets the acquisition time for a high-quality ghost image. Physically, this is brought about by the need to average over enough field patterns to distinguish the contribution of each pixel to the measurements. Turbulence weakens the correlation between each pixel and each measurement thus increasing the acquisition time.

The $\Delta^2 \mathcal{R}$ term in the noise denominator arises from the random fading that results from the rough-surface target reflection combined with the effects of atmospheric turbulence. When the total acquisition time spans many turbulence coherence times, $\Psi \rightarrow 0$ and this term simplifies. However, because the rough-surfaced target is assumed to be constant over the measurement interval, its contribution to the noise term remains. However, it can be mitigated by aperture averaging [21], i.e., by increasing the physical size of the bucket detector to capture and average more of the scattered light⁴.

The noise term $\Delta^2 \mathcal{D}$ is the contribution from the bucket detector's shot noise, and is inversely proportional to the number of transmitted photons, \mathbb{N}_s . For fixed optics

⁴This target-induced speckle can also be mitigated if multiple wavelengths are used, each casting the same pattern for each pulse, and the detector is sensitive to all of the wavelengths [2, 21]. However, this approach invites additional technical challenges and multi-spectral ghost imaging is beyond the scope of this thesis.

and target, this noise term is only decreased by increasing the number of photons transmitted per pulse, or using additional pulses so that the total number of detected photons over the image-acquisition time is increased.

The final noise term, $\Delta^2\mathcal{F}$ is a direct consequence of time-varying turbulence. If the turbulence is frozen, so that $\Psi \rightarrow e^{4(\sigma_s^2 + \sigma_r^2)} - 1$, this term disappears. When the turbulence is time varying, however, those variations create additional uncertainty about the relative spatial distribution of photons from pulse-to-pulse, making it harder to determine each resolution cell's reflectivity from the sample covariance of the reference patterns and the measurements. In the worst-case scenario for this term, when the turbulence decorrelates pulse-to-pulse, $\Delta^2\mathcal{F}$ scales as $A_T^2/N\rho_L^4$. Moreover, like $\Delta^2\mathcal{S}$ and $\Delta^2\mathcal{D}$, it is inversely proportional to the number of pulses, but while $\Delta^2\mathcal{S}$ grows linearly with the number of resolution cells, the worst-case for $\Delta^2\mathcal{F}$ grows with the *square* of the number of resolution cells. Consequently the number of pixels we need to reach the saturation SNR, described below, becomes quadratic in the number of measurements. This scaling can have a significant impact on the image acquisition time.

Signal-to-Noise Ratio Asymptotes

Having detailed the different sources of noise in the covariance-based ghost image, we turn to two physically important asymptotes for its SNR. As noted earlier, $\Delta^2\mathcal{S}$, $\Delta^2\mathcal{D}$, and $\Delta^2\mathcal{F}$ are all inversely proportional to the number of pulses, N . In contrast, the noise from the rough-surfaced target, $\Delta^2\mathcal{R}$, is independent of the number of pulses used, because the target surface is static throughout the imaging process. Target-induced speckle thus sets an ultimate upper bound on the achievable SNR for ghost imaging; when enough photons, pulses, and time have been used to average out all of the other sources of noise, the SNR will saturate at

$$\text{SNR}_{\mathcal{G}_{N,\text{sat}}}(\boldsymbol{\rho}) = 1 + 2\beta. \quad (3.29)$$

This limit is due solely to the time-independent target speckle and the relative size of the source and detector pupils. It is independent of the target's range and its underlying intensity reflection pattern. To increase this saturation SNR one must increase $\beta = r_b^2/a_0^2$ i.e., the ratio of the area of the detector to the area of the transmitter beam. It is preferable to increase β by increasing the size of the detector rather than decreasing the size of the transmitter beam. This is because when we vary a_0 we encounter a fundamental trade-off between the achievable SNR and spatial resolution, which is set by the coherence length $\rho_L = \lambda L/\pi a_0$. If we decrease a_0 we increase the SNR limit in Eq. (3.29), but we also increase ρ_L and thus degrade the ghost imager's spatial resolution.

At the other extreme, when the number of detected photons is so low that shot-noise dominates all other fluctuations in the ghost image, we find the shot-noise-limited SNR

$$\text{SNR}_{\mathcal{G}_{N,\text{shot}}}(\boldsymbol{\rho}) = \frac{2\pi\eta\mathbb{N}_s N A_b \rho_L^4 \mathcal{T}^2(\boldsymbol{\rho})}{A_T a_L^2 L^2}. \quad (3.30)$$

Unlike the saturation SNR, which is purely a function of the size of the pupil areas employed for transmission and detection, the shot-noise limited SNR depends on the target's range and reflectivity, both of which affect the number of detected photons.

3.4 Comparison to a Floodlight Laser Radar

We have shown that ghost imaging is a viable option for standoff imaging, but we need to put its performance in context. To do that we compare the performance of reflective ghost imaging to that of a state-of-the-art floodlight illumination LADAR. In this system the target is illuminated by a sequence of N laser pulses, and the reflected light is focused by a lens onto a CCD sitting at its focal plane. This system is shown in Fig. 3-3.

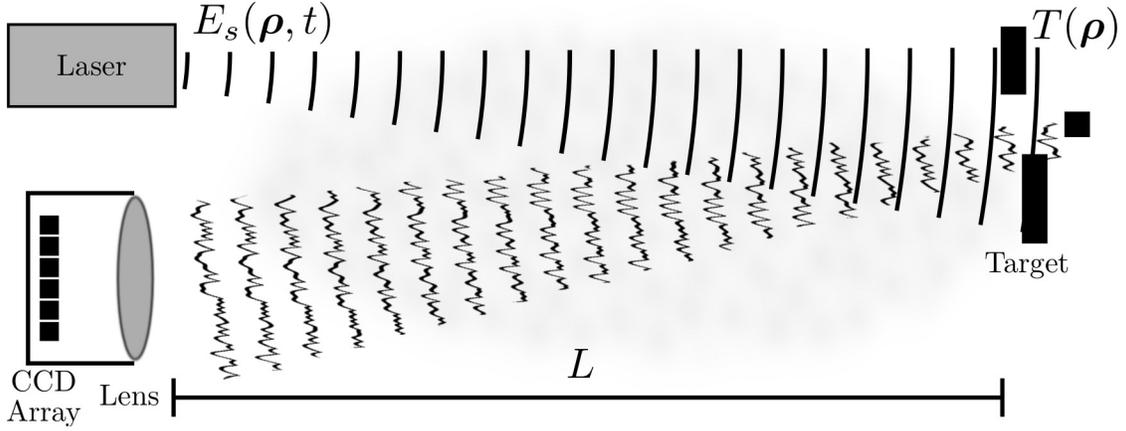


Figure 3-3: Setup for LADAR. A coherent source is propagated through atmospheric turbulence, where it floodlight illuminates the target. The returning light is focused by a lens onto a CCD array.

3.4.1 Framework

Most of the framework derived in Ch. 2 for structured illumination applies to the LADAR system, with the primary exception being the transmitter's spatial patterns. We still assume that the source is a series of pulses riding a spatial pattern as in Eq. (2.1), but instead of $\xi(\boldsymbol{\rho}, n)$ being a pseudo-randomly generated field, it is a laser beam with the Gaussian profile

$$\xi(\boldsymbol{\rho}, n) = \sqrt{\frac{2}{\pi w_0^2}} e^{-|\boldsymbol{\rho}|^2/w_0^2}, \quad (3.31)$$

where w_0 is the beam waist. For a fair comparison with the ghost imager, we set this waist equal to the coherence length of the ghost imager's transmitter, $w_0 = \rho_0$, so that the on-target fields have the same average intensity profile. The fields are still propagated with Eq. (2.5) over the same distance, reflect off the same collection of targets, and propagate back to the same location. The other difference is the method of photodetection; for the LADAR a lens is placed at the receiver plane that focuses the light onto a CCD array which produces pixel-wise photocurrents $\{i_p(t) : 1 \leq p \leq M\}$. The photocurrents are followed by a matched filter and then

sampled to produce the measurement matrix for pulse n and range L_k ,

$$i_p(n, k) = \int d\tau g(\tau_{n,k} - \tau) \left[q\eta \int_{\mathcal{A}_p^2} d\boldsymbol{\rho} |E_p(\boldsymbol{\rho}, \tau)|^2 + \Delta i_p(\tau) \right] \quad (3.32)$$

where: \mathcal{A}_p is the real-valued field-transmission pupil function defining the photosensitive region of pixel p , which we take to be centered at $\boldsymbol{\rho} = -\boldsymbol{\rho}_p$ to compensate for image inversion; $\tau_{n,k}$ is the delay associated with the measurement produced by pulse n and the target at range L_k , as defined in Eq. (2.19); $E_p(\boldsymbol{\rho}, \tau)$ is the complex envelope of the light impinging on the CCD plane at point $\boldsymbol{\rho}$ and time τ ; and $\Delta i_p(\tau)$ is the photocurrent shot noise for pixel p .

In practice, the lens's focal length is much shorter than the distance to the target so the lens will cast a minified image on the CCD. This scaling does not affect our spatial resolution or SNR analysis, so for convenience we assume the focal length is chosen to realize 1 : 1 imaging for range L_ℓ so that

$$E_p(\boldsymbol{\rho}, t) = \int_{\mathcal{A}_\ell} d\boldsymbol{\rho}' E_r(\boldsymbol{\rho}', t, L_\ell) e^{-ik_0|\boldsymbol{\rho}|^2/L_\ell} \frac{k_0 e^{ik_0(L_\ell + |\boldsymbol{\rho} - \boldsymbol{\rho}'|^2/2L_\ell)}}{i2\pi L_\ell}, \quad (3.33)$$

where \mathcal{A}_ℓ is the lens' real-valued field-transmission pupil function, $E_r(\boldsymbol{\rho}', t, L)$ is the pupil-plane target-return field defined by Eq. (2.5), and we have neglected an unimportant absolute phase factor and time delay resulting from propagation inside the radar receiver. To facilitate analytic comparison with the ghost imager, we will take the lens' pupil function to be $\mathcal{A}_\ell(\boldsymbol{\rho}) \equiv \exp(-|\boldsymbol{\rho}|^2/r_\ell)$.

To complete the framework we follow the derivation in Ch. 2 and assume the target components are sufficiently spaced, the pulse width τ_p sufficiently short, and the pulse repetition rate τ_s sufficiently long that our detector can discriminate between returns from different ranges and different pulses. Then, following the derivation in Eq. (2.22), the temporal effects of the propagation decouple from the transformation of the spatial patterns and we have

$$|E_p(\boldsymbol{\rho}, \tau)|^2 = \mathbb{N}_s \sum_{n=0}^{N-1} \sum_{k=0}^{K-1} |p(\tau - \tau_{n,k})|^2 |\xi_p(\boldsymbol{\rho}, n, k)|^2. \quad (3.34)$$

Again, the only effects of the matched filter are to discriminate between pulses, scale the measurement, and provide some shot-noise averaging. The spatial pattern $\xi_p(\boldsymbol{\rho}, n, k)$ at pixel p is found from the spatial pattern at the lens, $\xi_b(p', n, k)$, and is given by

$$\xi_p(\boldsymbol{\rho}, n, k) = \int_{\mathcal{A}_\ell} d\boldsymbol{\rho}' \xi_b(\boldsymbol{\rho}', n, k) e^{-ik_0|\boldsymbol{\rho}|^2/L_\ell} \frac{k_0 e^{ik_0(L_\ell + |\boldsymbol{\rho} - \boldsymbol{\rho}'|^2/2L_\ell)}}{i2\pi L_\ell}, \quad (3.35)$$

where $\xi_b(\boldsymbol{\rho}', n, k)$ can be back-propagated to the transmitter with Eqs. (2.24) and (2.23). We define the effective post-filter pulse width τ'_p with Eq. (2.25), the total number of received photons on pixel p for light returning from range L_k for pulse n as

$$\mathbb{N}_p(n, k) = \mathbb{N}_s \int_{\mathcal{A}_p^2} d\boldsymbol{\rho} |\xi_p(\boldsymbol{\rho}, n, k)|^2, \quad (3.36)$$

and the shot noise for that same detection is

$$\Delta i_p(n, k) = \int d\tau |p(\tau - \tau_{n,k})|^2 \Delta i_p(\tau). \quad (3.37)$$

The measurement matrix of sampled photocurrents in Eq. (3.32) then becomes

$$i_p(n, k) = \frac{q\eta}{\tau'_p} \mathbb{N}_p(n, k) + \Delta i_p(n, k). \quad (3.38)$$

At this point in ghost imaging analysis we noted that we could perform a reconstruction independently for each range L_k . For the LADAR, the story is slightly more complicated. When $k_0 r_\ell^2/2L \ll 1$ for all target ranges of interest the LADAR can be focused at infinity with no loss of spatial resolution, and all ranges in this regime can be processed independently. However, when this is not the case the laser must operate within the depth of focus of the lens or it will suffer resolution degradation. To avoid resolution loss we will assume operation in the far-field where $k_0 r_\ell^2/2L \ll 1$, but this will not always be the case in operation. We now drop the range index k , but note that the results only apply for in-focus targets. The LADAR then produces

an image by pixel-wise averaging the sampled photocurrents from the N pulses,

$$\mathcal{L}_N(\boldsymbol{\rho}_p) = \frac{1}{N} \sum_{n=0}^{N-1} i_p(n). \quad (3.39)$$

3.4.2 Spatial Resolution

The LADAR's spatial resolution is found from its ensemble-averaged image,

$$\langle \mathcal{L}_N(\boldsymbol{\rho}_p) \rangle = \langle i_p(n) \rangle. \quad (3.40)$$

Our analysis in this section follows the same approach as the ghost imaging spatial resolution analysis. We expand the photocurrent, and then back-propagate $\xi_p(\boldsymbol{\rho}, n)$ first through the lens, then to the target, and then to the transmitter using Eqs. (3.35), (2.24), and (2.23). Our averaging then produces second-order moments of the turbulence kernels and the rough-surfaced target. Evaluating these expectations, we arrive at

$$\langle \mathcal{L}_N(\boldsymbol{\rho}_p) \rangle = \frac{2q\eta\mathbb{N}_s A_\ell}{\pi L^2 w_L^2 \tau_p'} \int d\boldsymbol{\rho}' \mathcal{T}(\boldsymbol{\rho}') e^{-2|\boldsymbol{\rho}'|^2/w_L^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho} \frac{e^{-|\boldsymbol{\rho}+\boldsymbol{\rho}'|^2/\alpha'\rho_L'^2}}{\pi\alpha'\rho_L'^2}. \quad (3.41)$$

In this expression: $w_L = \lambda_0 L / \pi w_0$ is the laser beam's intensity radius at range L , which we will assume is sufficiently large that $\mathcal{T}(\boldsymbol{\rho}') e^{-2|\boldsymbol{\rho}'|^2/w_L^2} \approx \mathcal{T}(\boldsymbol{\rho}')$; $A_\ell \equiv \int d\boldsymbol{\rho} \mathcal{A}_\ell^2(\boldsymbol{\rho}) = \pi r_\ell^2 / 2$ is the lens' effective area; $\rho_L' = \lambda_0 L / \sqrt{2} \pi r_\ell$ is the diffraction-limited (no-turbulence) spatial resolution for an image-plane point detector; and $\alpha' \equiv 1 + r_\ell^2 \zeta_r / \varrho_r^2$ is a resolution-degradation factor caused by turbulence. Thus, atmospheric turbulence begins to impair the LADAR's spatial resolution when the receiver-plane coherence length of the turbulence kernel, $\varrho_r / \sqrt{\zeta_r}$, becomes comparable to the receiver's lens radius, r_ℓ . Hence, when the turbulence on the return path is either sufficiently weak, or concentrated near the target, we get $\alpha' \approx 1$ and there is no loss of resolution. However, for sufficiently strong turbulence, or turbulence concentrated near the receiver's lens, significant resolution degradation is incurred.

The computational ghost imager can calculate its reference field for continuous

coordinates $\boldsymbol{\rho}$, but the LADAR must use pixels of finite size in order to collect any photons. To obtain a closed-form result from the pixel integration in Eq. (3.41) we will use $\mathcal{A}_p(\boldsymbol{\rho}) \equiv e^{-|\boldsymbol{\rho}+\boldsymbol{\rho}_p|^2/r_p^2}$, and arrive at our final expression for the LADAR's average image,

$$\langle \mathcal{L}_N(\boldsymbol{\rho}_p) \rangle = \frac{2q\eta\mathbb{N}_s A_\ell A_p}{\pi L^2 w_L^2 \tau'_p} \int d\boldsymbol{\rho}' \mathcal{T}(\boldsymbol{\rho}') \frac{e^{-|\boldsymbol{\rho}'-\boldsymbol{\rho}_p|^2/(\beta'+\alpha')\rho_L'^2}}{\pi(\beta'+\alpha')\rho_L'^2}, \quad (3.42)$$

where r_p is the pixel's radius, $A_p \equiv \int d\boldsymbol{\rho}' \mathcal{A}_p^2(\boldsymbol{\rho}') = \pi r_p^2/2$ is the pixel's effective area, and $\beta' \equiv r_p^2/2\rho_L'^2$. Equation (3.42) quantifies the LADAR's loss of spatial resolution—its degradation from ρ_L' , the diffraction-limited, point-detector value—when the lens area exceeds a turbulence coherence area and/or the pixel area exceeds the diffraction-limited spot size.

3.4.3 Signal-to-Noise Ratio

We again define the signal-to-noise ratio as the ratio of the image's squared mean to variance,

$$\text{SNR}_{\mathcal{L}}(\boldsymbol{\rho}_p) = \frac{\langle \mathcal{L}_N(\boldsymbol{\rho}_p) \rangle^2}{\text{var}[\mathcal{L}_N(\boldsymbol{\rho}_p)]} = \frac{\langle \mathcal{L}_N(\boldsymbol{\rho}_p) \rangle^2}{\langle \mathcal{L}_N^2(\boldsymbol{\rho}_p) \rangle - \langle \mathcal{L}_N(\boldsymbol{\rho}_p) \rangle^2}. \quad (3.43)$$

To evaluate the effects of turbulence on the SNR we make the same assumptions that we made in the ghost image analysis. Namely, we assume the target's features are large enough that the LADAR resolves them, and that the turbulence is weak enough that we can ignore the coordinate dependence at the transmitter and receiver pupils. With the first assumption, Eq. (3.42) yields

$$\langle \mathcal{L}_N(\boldsymbol{\rho}'_p, L) \rangle^2 = \left[\frac{2q\eta\mathbb{N}_s A_\ell A_p}{\pi L^2 w_L^2 \tau'_p} \mathcal{T}(\boldsymbol{\rho}'_p) \right]^2. \quad (3.44)$$

As was the case for the ghost imager, the primary difficulty encountered in SNR evaluation is finding the image's second moment. The LADAR's second-moment calculation, however, is substantially simpler than that for the ghost imager in that

the LADAR case only requires turbulence and target fourth-order moments and the shot noise's second-order moment. The evaluation proceeds as follows.

First, we use the statistical independence of the turbulence, target, and shot noise to separate averages involving these three fluctuations. Then we employ Gaussian moment-factoring to reduce the target's fourth-order moment into a sum of products of second-order moments, and our assumption about the coordinate-independence of the turbulence fluctuations to evaluate that fourth-order moment as we did for the ghost imager. Next, we perform the resulting multi-dimensional Fourier transforms of the Gaussian functions that arise from far-field optical propagation in conjunction with the Gaussian pupil functions we have assumed. Finally, under our assumption that the LADAR resolves all significant target detail, we are left with

$$\text{SNR}_{\mathcal{L}_N}(\boldsymbol{\rho}_p) = \frac{\mathcal{T}(\boldsymbol{\rho}_p)}{\mathcal{T}(\boldsymbol{\rho}_p)\left(\frac{1+(2+\beta')\Psi}{1+\beta'}\right) + \frac{\pi w_L^2 L^2}{2\eta N_s N A_\ell A_p}}, \quad (3.45)$$

where Ψ is the time-averaged turbulence factor from Eq. (3.23).

The noise-denominator terms in the preceding SNR formula have the following physical interpretations. The first term is due to the time-independent target speckle, exacerbated, to some degree, by the turbulence-induced scintillation on the transmitter-to-target and target-to-receiver paths. The second term is due to shot noise. Thus when NN_s —the number of photons transmitted over all N pulses—is sufficiently high, the SNR reaches a finite maximum value, namely the saturation signal-to-noise ratio, given by

$$\text{SNR}_{\mathcal{L}_N, \text{sat}}(\boldsymbol{\rho}_p) = 1 + \beta', \quad (3.46)$$

which depends only on $\beta' = r_p^2/2\rho_L'^2$, the number of diffraction-limited spots within the pixel area. Increasing β' , e.g., by increasing the pixel size, will increase the saturation value, but doing so degrades the radar's spatial resolution, as seen in Eq. (3.42). Conversely, when the SNR is much lower than its saturation limit, it takes the shot-

noise limited form

$$\text{SNR}_{\mathcal{L}_{N,\text{shot}}}(\boldsymbol{\rho}_p) = \frac{2\eta\mathbb{N}_s N A_\ell A_p \mathcal{T}(\boldsymbol{\rho}_p)}{\pi w_L^2 L^2}. \quad (3.47)$$

3.5 Performance Comparison

Having completed spatial resolution and SNR analyses for both the computational ghost imager and the LADAR, we are ready to compare their capabilities. Before proceeding, two points deserve further explanation. The first concerns ensuring that our spatial resolution and SNR comparisons are fair. Toward that end we will take $w_0 = \rho_0$, so that the LADAR's on-target intensity pattern matches the ghost imager's average on-target intensity pattern. We will also assume that $r_\ell = a_0$, because: (1) r_ℓ and a_0 correspond to intensity radii (r_ℓ for the the intensity transmission of the radar receiver, and a_0 for the intensity transmission of the ghost imager's exit optics); and (2) r_ℓ and a_0 determine the diffraction-limited spatial resolutions of these two imagers.

In addition, we note that for both systems we denoted the number of photons transmitted per pulse with \mathbb{N}_s , while the number of pulses was N . These quantities need not be the same for both imagers, so to differentiate them we introduce subscripts of \mathcal{G} and \mathcal{L} , e.g., the ghost imager transmits $\mathbb{N}_{\mathcal{G}}$ photons for each of its $N_{\mathcal{G}}$ pulses, and the LADAR transmits $\mathbb{N}_{\mathcal{L}}$ photons for each of its $N_{\mathcal{L}}$ pulses. We will require that both systems use the same number of photons for image acquisition, so $\mathbb{N}_{\mathcal{G}}N_{\mathcal{G}} = \mathbb{N}_{\mathcal{L}}N_{\mathcal{L}}$, but we afford the LADAR the opportunity to concentrate its photons into fewer pulses, or even a single, bright pulse. In contrast, the ghost imager fundamentally requires many pulses to form a high quality image.

Our second pre-comparison note is cautionary. Our analysis has employed Gaussian functions for partially-coherent and coherent optical beams, and for photodetector pupil-functions, etc., in order to obtain closed-form expressions for our systems' spatial resolutions and SNRs. Although the use of Gaussian functions should yield the correct dependence of these performance metrics on system parameters, the constant

factors that appear in our results would be different had we numerically evaluated the spatial resolutions and SNRs for circular-pupil optics.

3.5.1 Spatial-Resolution Comparison

We have shown that the ghost imager and LADAR’s average images convolve $\mathcal{T}(\boldsymbol{\rho}')$ with Gaussian point-spread functions of the form $P(\boldsymbol{\rho}') = e^{-|\boldsymbol{\rho}'|^2/r_{\text{res}}^2}/\pi r_{\text{res}}^2$, where

$$r_{\text{res}} = \begin{cases} \sqrt{\alpha} \rho_L, & \text{ghost imager} \\ \sqrt{\beta' + \alpha'} \rho'_L, & \text{LADAR} \end{cases}, \quad (3.48)$$

with ρ_L and ρ'_L being the diffraction-limited resolutions, $\alpha \geq 1$ and $\alpha' \geq 1$ accounting for turbulence-induced resolution loss, and $\beta' \geq 0$ for the resolution lost from using finite pixel sizes for the LADAR.

Diffraction-Limited Resolution

When there is no turbulence, both resolutions are diffraction limited. Enforcing our constraint of $r_\ell = a_0$, these become

$$r_{\text{res}} \rightarrow \begin{cases} \lambda_0 L / \pi a_0, & \text{ghost imager} \\ \sqrt{\beta' + 1} \lambda_0 L / \sqrt{2} \pi a_0, & \text{LADAR} \end{cases}, \quad (3.49)$$

and we immediately see that the resolution difference is a factor of $\sqrt{(\beta' + 1)/2}$; if we set $\beta' = 0$, the LADAR has a $\sqrt{2}$ resolution advantage. The source of this advantage can be traced back to the physics of the imaging scenarios. The LADAR’s spatial resolution is set by its lens’ radius—which is now a_0 —but the ghost imager’s spatial resolution comes from an intensity convolution of two beams produced by a transmitter of intensity radius a_0 . This convolution causes the $\sqrt{2}$ degradation in the effective resolution, despite both systems using the same pupil sizes⁵.

⁵The exact numerical value ($\sqrt{2}$) of this resolution advantage depends on our use of Gaussian functions, but the fact that the LADAR’s diffraction-limited spatial resolution is better than that of the computational ghost imager does not.

As noted in Sec. 3.4, the LADAR must employ finite-sized pixels, and Eq. (3.45) shows that—other system parameters being held constant—forcing $\beta' \ll 1$ will push the system into its shot-noise limited regime as the pixel area, A_p , is decreased. Thus a prudent compromise might be to size the pixels to satisfy $\beta' = 1$, so that the LADAR’s spatial resolution, in the absence of turbulence, only suffers a $\sqrt{2}$ -factor degradation from its diffraction-limited value. This choice of pixel size, however, removes the LADAR’s spatial resolution advantage over the computational ghost imager, i.e., the no-turbulence resolution of the former then exactly matches the diffraction-limited resolution of the latter.

Turbulence Limited Resolution

When turbulence is strong enough to control both imager’s spatial resolution, we get

$$r_{\text{res}} \rightarrow \begin{cases} \lambda_0 L \sqrt{\zeta'_s} / \pi \sqrt{2} \varrho_s, & \text{ghost imager} \\ \lambda_0 L \sqrt{\zeta'_r} / \pi \sqrt{2} \varrho_r, & \text{LADAR} \end{cases}, \quad (3.50)$$

where we have continued our use of $r_\ell = a_0$. For a situation in which $\varrho_s / \sqrt{\zeta'_s} = \varrho_r / \sqrt{\zeta'_r}$ —such as when the LADAR’s receiver is co-located with the ghost-imager’s transmitter—we find the turbulence-limited resolution to be equal for both systems. More generally, either system could have more favorable spatial-resolution behavior in the presence of turbulence, because different turbulence-strength profiles could exist on the transmitter-to-target and target-to-receiver paths, and the ghost imager’s spatial resolution is only sensitive to turbulence on the transmitter-to-target path while the LADAR’s spatial resolution is only sensitive to the turbulence on the target-to-receiver path.

3.5.2 Signal-to-Noise Ratio Comparison

For our SNR comparison we shall consider the two imagers’ speckle limited, saturation SNRs and their shot-noise limited SNRs. Throughout we will assume that any turbulence that might be present has no effect on spatial resolution, so that

$\alpha = \alpha' = 1$, and that both imagers have sufficient resolution to resolve all significant detail in $\mathcal{T}(\boldsymbol{\rho})$.

Speckle Limited Signal-to-Noise Ratio

Equations (3.29) and (3.46) specify the speckle limited, saturation SNRs for the ghost imager and the LADAR, and give us

$$\frac{\text{SNR}_{\mathcal{L}_N, \text{sat}}(\boldsymbol{\rho}_p)}{\text{SNR}_{\mathcal{G}_N, \text{sat}}(\boldsymbol{\rho}_p)} = \frac{1 + \beta'}{1 + 2\beta}. \quad (3.51)$$

For $r_\ell = a_0$ and $\beta' = 1$ (for which both systems have the same spatial resolution) the preceding ratio of their saturation SNRs becomes $2/(1 + 2\beta)$. Consequently, the computational ghost-imager's saturation SNR will greatly exceed that of the LADAR when $\beta = r_b^2/a_0^2 \gg 1$, i.e., when the bucket detector's area is much larger than the source area. There is a simple physical explanation for this behavior: the saturation SNR is due to the time-independent speckle created by reflection from the rough-surfaced target. These speckles are $\sim a_0$ in radius in the receiver's pupil plane, so when $\beta \gg 1$, the bucket detector is averaging over many statistically independent speckles, driving up the ghost imager's saturation SNR.

There is, however, an intrinsic unfairness in the preceding favorable view of the ghost imager's saturation SNR, because by fixing $r_\ell = a_0$ but letting $\beta \gg 1$, we are allowing the bucket detector to have a much larger receiving aperture than the LADAR. Suppose, instead, that we constrain the systems to have the same size receiving aperture, so that $r_b = r_\ell$, but maintain $\beta \gg 1$. In that case the LADAR's diffraction-limited spatial resolution would be better than that of the ghost imager by a factor of $r_b/a_0 \gg 1$ but its SNR would be worse. However, the LADAR could then use $\beta' = 2r_b^2/a_0^2 \gg 1$ and: (1) have its no-turbulence spatial resolution match the diffraction limit of the ghost imager; and (2) have its saturation SNR equal that of the ghost imager. In this case, the time-independent speckle is being averaged in the LADAR's image plane, because the large pixel comprises a great many independent speckle lobes.

For a general comparison, we set all optics to be the same size, i.e., $a_0 = r_b = r_\ell$; this causes $\beta = 1$. We then set the LADAR's pixel size to effect $\beta' = 1$, so that both systems have the same resolution, and we find $\text{SNR}_{\mathcal{L}_N, \text{sat}}(\boldsymbol{\rho}_p) = 2 \text{SNR}_{\mathcal{G}_N, \text{sat}}(\boldsymbol{\rho}_p)/3$.

To summarize what we have seen so far concerning saturation-SNR behavior, fair comparisons between the ghost imager and the LADAR indicate that neither system will enjoy a significant advantage. There is, however, a fine point to be considered concerning atmospheric turbulence. We have assumed the relevant turbulence coherence lengths to be long enough that our imagers' spatial resolutions do not suffer any turbulence-induced degradation. Nevertheless, the target-speckle terms in the noise denominators of Eqs. (3.24) and (3.45) both contain the time-averaged scintillation factor Ψ . When a sufficient number of pulses are averaged—as has been assumed in the saturation SNR formulas—we get $\Psi \rightarrow 0$. For the ghost imager, averaging the returns from a large number of pulses is an intrinsic requirement for image formation, but a LADAR can form its image with a single pulse, in which case we get

$$\text{SNR}_{\mathcal{L}_1}(\boldsymbol{\rho}'_p, L) \longrightarrow \frac{1 + \beta'}{1 + (2 + \beta')\Psi}, \quad (3.52)$$

for $N_{\mathcal{L}}$ sufficiently high. For $\beta' = 2r_b^2/a_0^2 \gg 1$, this SNR can be substantially *worse* than the ghost imager's saturation SNR, because $\Psi \geq e^{4\sigma_s^2}$ in this single-pulse case. The LADAR will then need to employ more than a single pulse to approach its full saturation SNR—and to match that of the computational ghost imager at the same spatial resolution—because of scintillation.

Shot-Noise Limited Signal-to-Noise Ratio

Now let us turn to SNR behavior when neither system's N and N values are sufficient to reach SNR saturation by comparing their shot-noise limited performance. Here, using $r_\ell = a_0$, Eqs. (3.30) and (3.47) give us

$$\frac{\text{SNR}_{\mathcal{L}_N, \text{shot}}(\boldsymbol{\rho}_p)}{\text{SNR}_{\mathcal{G}_N, \text{shot}}(\boldsymbol{\rho}_p)} = \frac{A_T \beta'}{2\pi \rho_L^2 \beta \mathcal{T}(\boldsymbol{\rho}_p)}. \quad (3.53)$$

For a fair comparison we again set $\beta = \beta' = 1$, to give the systems the same pupil sizes and spatial resolutions. The preceding SNR ratio then reduces to $A_T/2\pi\rho_L^2\mathcal{T}(\rho'_p)$. For a range- L target whose intensity-reflection coefficient has limited spatial variation, this number is approximately its number of reflective spatial-resolution cells, making the LADAR's shot-noise limited SNR far superior to that of the computational ghost imager for large, highly reflective images. Unlike what we found for the LADAR's saturation SNR, its shot-noise limited SNR is the same at all $N_{\mathcal{L}}N_{\mathcal{L}}$ values, i.e., Eq. (3.53) applies even when the LADAR forms its image from a single pulse.

This discrepancy in the shot-noise limited SNR of the two systems arises because in ghost imaging all useful information in the bucket detector photocurrent is carried in the pulse-to-pulse fluctuations in received number of photons, but all received photons contribute to the shot noise. Indeed, the first step in ghost imaging is to center the bucket detector measurements and reference patterns, because for each pulse there is a strong, mean photon count that is contributing only to noise in the bucket detector. In contrast, each detected photon in the LADAR will contribute to both the shot noise and the signal. This also explains the dependence of Eq. (3.53) on $\mathcal{T}(\rho)$. In the ghost imager there is always noise because of the background term, but if pixel p is not reflective there will be no detected photons to cause noise. This means that, when both systems are shot-noise limited, non-reflecting pixels in a ghost image will be much noisier than in a LADAR image⁶.

3.5.3 Conclusions

Computational ghost imaging is, in many respects, a dual of floodlight-illumination LADAR. The computational ghost imager's system complexity is in its transmitter, whose sequence of SLM patterns creates the structured illumination that provides the imager's spatial resolution. The LADAR's complexity lies in its CCD receiver, which provides its spatial resolution. Consequently, the size of the ghost imager's transmitter pupil sets its diffraction-limited spatial resolution, whereas the LADAR's

⁶Our analysis has neglected dark current and background light, both of which will cause some noise for non-reflecting locations in the LADAR, but we still expect much more noise for the ghost imager.

no-turbulence spatial resolution is set by the size of its receiver pupil in conjunction with that of its CCD pixels. Thus only turbulence on the transmitter-to-target path can impair the ghost imager’s spatial resolution, while the LADAR’s spatial resolution is only impacted by turbulence on the target-to-receiver path. Therefore, ghost imaging and LADAR systems that are designed to have equal spatial resolutions in the absence of turbulence could have significantly different performance in a bistatic configuration, in which their transmitters and receivers are not co-located, so that significantly different turbulence distributions are encountered on these two paths. In such situations either one could offer the better spatial-resolution performance, depending on which path had its turbulence concentrated near the resolution-controlling pupil.

Aside from this turbulence issue, our analysis indicates that a fair comparison between the computational ghost imager and the floodlight-illumination LADAR shows them to have equal spatial resolutions, *except* for the following two caveats: (1) the LADAR can form its image from a single pulse, making it far better for imaging moving targets; and (2) the computational ghost imager has infinite depth of focus, whereas the LADAR will not have this property for ranges satisfying $k_0 r_\ell^2 / 2L \gg 1$.

Both the ghost imager and the LADAR have signal-to-noise ratios that are shot-noise limited at low- N_s , low- N values and saturate at high- N_s , high- N values. When their optics are sized for equal spatial resolutions, with $N_G = N_L$ and $N_G = N_L$, there is little difference in their saturation SNRs. This contrasts strongly with their shot-noise limited SNR behaviors, where the LADAR outperforms the ghost imager by a factor approximately equal to the number of reflective spatial-resolution cells on the target. As a result, we can expect that the ghost imager will require significantly more time than the LADAR to achieve a desired SNR when both are operating in the shot-noise limited regime. This key disadvantage for correlation-based ghost imaging could be mitigated to some degree, however, by the use of compressed-sensing techniques, which enable many fewer pulses to suffice for ghost-image formation. Recent work has demonstrated this possibility in table-top ghost imaging done in reflection [34], and the need to reduce the number of pulses for ghost imaging is a strong motivating

factor for the second half of this thesis.

What then are the possible advantages of ghost imaging in comparison with LADAR? The principal such advantage identified by our analysis accrues in bistatic configurations wherein, for operational reasons, the transmitter can be located in a region of weak turbulence but the receiver necessarily is in a strongly-turbulent region. Beyond that, however, there are some technological possibilities. The ghost imager only requires a single-pixel detector, whereas the LADAR needs a detector array. For wavelength regions in which high-performance single-pixel detectors are available but similar-quality detector arrays are not, ghost imagers would provide active-imaging capability that LADARs could not. A related technological advantage arises for ghost imaging in multi-static configurations, in which a network of simple, small, single-pixel detectors view a target region that is floodlit by a single, structured-illumination transmitter. Individual images could be formed from each detector’s outputs to capture multiple views of the target, and allow for more averaging of the target-induced speckle. A corresponding multi-static LADAR would require high-resolution CCDs at each receiver location, making it more complicated and more expensive than the ghost imager.

3.6 Advances in Ghost Imaging

Since the preliminary experiments and analysis of ghost imaging, several new methods of image formation have been proposed. These methods have been developed for the transmissive ghost imaging modality—in which the bucket detector is behind the target and we image its transmissivity pattern—but they can be extended to the reflective case.

In ghost imaging experiments it was observed that the total number of on-target photons varied from pulse to pulse. This is partially explained by varying emission levels from the laser, partially by the pseudorandom patterns—a random fraction of the light will always be cast outside the scene—and partially by experimental errors. In Eq. (3.2) we formed the ghost image by correlating centered version of

the reference pattern and bucket detector measurements, $\Delta I(\boldsymbol{\rho}, n)$ and $\tilde{i}_b(n)$, where $\tilde{i}_b(n)$ was computed in Eq. (3.3) by subtracting the empirical mean $\frac{1}{N} \sum_n i_b(n)$ from the measurement. However, this is not the best way to center the bucket detector measurements

We define the total number of on-target photons as,

$$R_n = \int_{\mathcal{A}_T} d\boldsymbol{\rho} I(\boldsymbol{\rho}, n), \quad (3.54)$$

where \mathcal{A}_T is the area of the target and $I(\boldsymbol{\rho}, n)$ is the reference pattern. Because R_n varies from pulse-to-pulse, the mean of the bucket detector measurements is not a constant, but a vector in which each element is weighted by the relative number photons that were used to create that measurement as

$$\bar{i}_b(n) = \frac{\sum_m i_b(m)}{\sum_\ell R_\ell} R_n. \quad (3.55)$$

In differential ghost imaging [36], the ghost image is found from a covariance calculation in Eq. (3.2) as normal, except that $\tilde{i}_b(n) = i_b(n) - \bar{i}_b(n)$, where $\bar{i}_b(n)$ is computed with Eq. (3.55). This was done by making an auxiliary measurement of R_n for each pulse. Interestingly, this adjustment means that the estimator is actually for a centered version of the target, i.e., the pattern of differences from the target's mean value, hence the name differential ghost imaging. This method showed significant improvement versus traditional ghost imaging when the scene was mostly transmissive, and the performance of the two methods converged when the scene was mostly opaque.

Normalized ghost imaging [37] makes the same adjustment as differential ghost imaging when calculating $\bar{i}_b(n)$, but then also scales each measurement by $1/R_n$, i.e., it correlates $\tilde{i}_b(n)/R_n$ with the centered reference patterns. This method achieved slightly better performance than differential ghost imaging. While both of these methods provide improved reconstruction versus traditional ghost imaging, the difference is only prominent when the target is mostly transmissive. Moreover, they require an additional measurement of the integrated on-target intensity pattern, R_n .

However, in reflective ghost imaging, this measurement is not feasible. R_n can be approximated from our computation of the reference field, but due to turbulence and rough surface scattering this computation will only approximate the true R_n , and it is unclear if this computed value will help the inference. Instead, in Ch. 6, we derive the LMMSE estimator. The form of this estimator will be similar to the normalized ghost image estimator, but also incorporates additional information about a prior distribution on the target.

Differential and normalized ghost imaging improved the linear covariance estimator, but that is not the only way to reconstruct the target. We observe that in transmissive ghost imaging the bucket detector measurement is a projection of the on-target intensity pattern onto the intensity transmission profile of the mask. That is, for bucket detector measurement vector \mathbf{y} , target vector \mathbf{x} and measurements matrix \mathbf{A} (where each row is a vector representing an on-target intensity pattern), we have $\mathbf{y} = \mathbf{A}\mathbf{x}$. Since \mathbf{A} is composed of iid random variables, this fits into the compressed sensing paradigm [38–40], which finds a sparse solution via ℓ_1 -norm minimization. Natural images are sparse in discrete wavelet and discrete cosine bases, so this approach allows for considerably fewer measurements than traditional covariance-based ghost imaging, and was first applied in [32].

Compressed sensing approach was also extended to reflective imaging in [35]. However, to overcome the target speckle they employed a telescope for their bucket detector, whose pupil was over 400 times the size of transmitter pupil. Thus, per our analysis in this Sec. 3.2, they sacrificed orders-of-magnitude in resolution in order to achieve the measurement reduction afforded by compressed sensing methods. Our goal in Chs. 8 and 9 is to develop an algorithm that can achieve this same reduction in measurements to achieve a high quality image while using much smaller bucket detectors.

Chapter 4

Discretization of the On-Target Irradiance and the Target's Field-Reflection Coefficient

Structured illumination imaging inherently involves continuous processes: the transmitter emits a field in continuous space and time; the target has a field-reflection coefficient that is a function of a continuous spatial coordinate; and the photodetector's current is a continuous-time waveform. However, to bring modern inference techniques to bear on image formation requires processing with a computer, which means the measurements and target must be discretized. We have already done this for the bucket detector's output by sampling its filtered photocurrent, but each measurement also needs to be constructed as a function of a discretized (pixelated) target in order to proceed.

As developed in Eq. (2.28), by sampling the photocurrent at delays $\tau_n = 2L/c + n\tau_s$ we create a vector of measurements \mathbf{y} with elements $y_n = i_b(n)$. The measurements are time samples of a shot-noise process that is driven by a conditionally Poisson process with rate function proportional to the detected photon-flux, $\eta P_b(t)$. The total number of received photons, ($\mathbb{N}_b(n)$ from Eq. (2.27)), then drives the statistics of the samples of the filtered photocurrent. From Eqs. (2.28) and (2.29) we have

that the means and covariances of the $\{y_n\}$ conditioned on \mathbb{N}_b are

$$\langle y_n | \mathbb{N}_b \rangle = \frac{q\eta}{\tau_p'} \mathbb{N}_b(n) \quad (4.1)$$

and

$$\langle (y_n - \langle y_n | \mathbb{N}_b \rangle)(y_m - \langle y_m | \mathbb{N}_b \rangle) \rangle = \delta_{nm} \frac{q^2 \eta \mathbb{N}_b(n)}{\tau_p''^2}. \quad (4.2)$$

For convenience going forward, we define a vectorized form of the bucket-detected photons for pulse n as $\mathcal{N}_n \equiv \mathbb{N}_b(n)$. Our remaining discretization task is then to pixelate

$$\mathcal{N}_n = \mathbb{N}_s \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \left| \int d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_n) T(\boldsymbol{\rho}') \xi_t(\boldsymbol{\rho}', n) \right|^2, \quad (4.3)$$

which requires us to discretize the target's field-reflection coefficient $T(\cdot)$, and its coupling into each measurement.

The pixels we will introduce in this chapter are not related to a physical sensor, and in theory we can choose them to be any size we want. There are trade-offs, however, to consider. Very small pixels lead to less approximation error, but they increase the computational complexity of the image construction. Moreover, this implies that the light hitting neighboring pixels will be highly correlated, which becomes problematic later when we design algorithms in Ch. 8 that rely on each pixel seeing independent illumination. Conversely, while large pixels lead to independent illumination on each pixel, the discretization error can be so large that there is no hope of performing accurate imaging.

A natural choice for the discretization scale is the coherence length of the illuminating field. This choice will ensure that neighboring pixels have light that is only weakly correlated, yet also keeps the discretization error to a manageable level. We thus start by discretizing the continuous reference irradiance pattern from Eq. (3.1) at its coherence length. Next we express the target's field-reflection coefficient in terms of a discretized version of its average intensity reflection $\mathcal{T}(\boldsymbol{\rho})$. Finally, we develop

the transformation of the target to the measurements, conditioned on knowledge of our discretized reference pattern, and determine sufficient statistics to describe it.

4.1 Reference-Pattern Discretization

The continuous-space on-target reference irradiance pattern is defined in Eq. (3.1). To discretize this pattern we integrate it over the pixel's reflective area \mathcal{A}_p . However, we also want to scale it appropriately to account for the average photon loss as the reflected field propagates back to the detector plane. Including this scaling now will simplify many equations later in this thesis. The new scaled, discretized reference is then

$$I_{np} = \mathcal{F} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho} \left| \int d\boldsymbol{\rho}' \xi(\boldsymbol{\rho}', n) \frac{k_0 e^{ik_0|\boldsymbol{\rho}'-\boldsymbol{\rho}|^2/2L}}{i2\pi L} \right|^2. \quad (4.4)$$

where the scaling coefficient is the Fresnel number product,

$$\mathcal{F} = \frac{\pi \varrho_T^2 A_b}{\lambda_0^2 L^2}, \quad (4.5)$$

that defines the fraction of photons measured by a bucket detector with area A_b when the light is scattered by a surface with correlation length ϱ_T . The reference pattern is dimensionless, but its transverse pattern is proportional to the number of photons illuminating pixel p for pulse n .

The characteristics of the $\{I_{np}\}$ will play a key role in this thesis, so we take this opportunity to discuss their more important properties. First, they are not independent, random variables, but they are close to being so. Having pixelated at the correlation length, there is some correlation between the field illuminating neighboring pixels, but it is small enough that it can be safely ignored.

Even assuming the $\{I_{np}\}$ are independent random variables, they need not be identically distributed. This follows from the spatial dependence of $\langle |\xi_t(\boldsymbol{\rho}, n)|^2 \rangle$, the average intensity envelope of the partially-coherent field. We define the normalized

intensity envelope of the reference as

$$\epsilon_p = \frac{\sum_n \langle I_{np} \rangle}{\sum_{n,q} \langle I_{nq} \rangle}, \quad (4.6)$$

so that the vector ϵ is the relative distribution of photons across the scene of interest, ensemble averaged across the series of pulses. It will be higher near the center of the beam, and smoothly fall off as we move toward the beam's edges.

In the Gaussian-Schell model we have used for our ghost imaging analysis¹, the envelope for pixel p —centered at ρ_p —is proportional to $e^{-2|\rho_p|^2/a_L^2}$, where a_L is the intensity radius, so that

$$\epsilon_p = \frac{e^{-2|\rho_p|^2/a_L^2}}{\sum_q e^{-2|\rho_q|^2/a_L^2}}. \quad (4.7)$$

We are also interested in the total number of on-target photons per pulse. On average this is the same for each pulse, but in operation the pseudo-random pattern will put a variable number of photons on-target, while the rest go outside the boundaries of the scene. The total number of on-target photons (scaled by the loss factor \mathcal{F}) for pulse n is

$$\mu_n = \sum_p I_{np}, \quad (4.8)$$

allowing us to construct the conditional mean of the reference matrix, given knowledge of μ , as the outer product of these two vectors,

$$\bar{\mathbf{I}} = \langle \mathbf{I} \rangle = \boldsymbol{\mu} \boldsymbol{\epsilon}^T. \quad (4.9)$$

This matrix is deterministically calculated from both the transmitter statistics and the reference patterns \mathbf{I} . A centered version of the reference matrix can be defined

¹The Gaussian-Schell model presumes a true Gaussian random process for the source, while the spatial light modulator (SLM) is better modeled as an array of square pixels that receive a pseudo-random phase modulation. A more accurate computation for the SLM geometry is presented in App. A.2

by removing this mean matrix, $\tilde{\mathbf{I}} = \mathbf{I} - \bar{\mathbf{I}}$.

4.2 Target Field-Reflection Coefficient

We next discretize the target's field reflection coefficient from Eq. (2.14), which we reproduce here,

$$T(\boldsymbol{\rho}') = \sqrt{\mathcal{T}(\boldsymbol{\rho}')} \vartheta(\boldsymbol{\rho}'). \quad (4.10)$$

$\mathcal{T}(\boldsymbol{\rho}')$ is the average intensity-reflection coefficient, while $\vartheta(\boldsymbol{\rho}')$ represents the contribution from the target's surface roughness, and is modeled as a complex-Gaussian random process. The correlation length of $\vartheta(\boldsymbol{\rho}')$ is on the order of a wavelength, and we cannot discretize at a small enough scale to capture its behavior. Instead we discretize $\mathcal{T}(\boldsymbol{\rho}')$, which varies slowly enough that we can approximate it as a grid of pixels, so that

$$\mathcal{T}(\boldsymbol{\rho}') = \mathcal{T}_p \quad \text{for } \boldsymbol{\rho}' \text{ in } \mathcal{A}_p, \quad (4.11)$$

where \mathcal{A}_p is the w_p by w_p square that defines reflective area of pixel p , with w_p being approximately the coherence length of the reference pattern. The target's field-reflection coefficient is now the sum

$$T(\boldsymbol{\rho}') = \sum_p \sqrt{\mathcal{T}_p} \vartheta(\boldsymbol{\rho}') \mathcal{A}_p(\boldsymbol{\rho}'). \quad (4.12)$$

Substituting this new definition into the received photon number in Eq. (4.3) we arrive at

$$\mathcal{N}_n = \mathbb{N}_s \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \left| \sum_p \sqrt{\mathcal{T}_p} \int_{\mathcal{A}_p} d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_n) \vartheta(\boldsymbol{\rho}') \xi_t(\boldsymbol{\rho}', n) \right|^2. \quad (4.13)$$

4.3 Target-to-Measurements Transformation

In the discrete formulation in Eq. (4.13), pixel p 's contribution to the received photon number for pulse n contains an effective field reflection coefficient that is the product of the square root of the average intensity reflection coefficient $\sqrt{\mathcal{T}_p}$, and the speckle-field coefficient

$$\Phi_{np}(\boldsymbol{\rho}) = \int_{\mathcal{A}_p} d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_n) \vartheta(\boldsymbol{\rho}') \xi_t(\boldsymbol{\rho}', n), \quad (4.14)$$

which is the result of the propagation of the interrogating field back from pixel p to the detector after having after being modulated by the rough surface, $\vartheta(\boldsymbol{\rho}')$. The $\{\Phi_{np}(\boldsymbol{\rho})\}$ are Gaussian random processes in $\boldsymbol{\rho}$ whose correlation length is inversely proportional to the pixel size we have chosen for discretizing the reference pattern. The number of received photons can now be rewritten as a double summation over the target's pixel index,

$$\mathcal{N}_n = \mathbb{N}_s \sum_p \sum_q \sqrt{\mathcal{T}_p \mathcal{T}_q} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \Phi_{np}(\boldsymbol{\rho}) \Phi_{nq}^*(\boldsymbol{\rho}). \quad (4.15)$$

For each pulse n , Eq. (4.15) can be viewed as the summation of the elements of a matrix indexed by p and q . We can separate that summation into two parts: the trace of the matrix (summation of the diagonal elements, where $p = q$), and the summation of the off-diagonal elements. The trace is a linear combination of the target's average intensity-reflection coefficient, weighted by coefficients that depend on the rough-surface induced speckle and the intensity of the light hitting the pixel. It is a direct, linear coupling of each pixel in the target to the measurements. The off-diagonal terms in Eq. (4.15) result from the integration of inter-pixel interference patterns over the detector's surface. Although they depend on the target, they do not transfer much information. Their randomness is dominated by pulse-dependent relative phase shifts, unique to each term. Both of these effects are captured by a

$\mathcal{I}(p, q, n)$ for pixels p and q , when they are illuminated by pulse n ,

$$\mathcal{I}(p, q, n) = \mathbb{N}_s \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \Re[\Phi_{np}(\boldsymbol{\rho})\Phi_{nq}^*(\boldsymbol{\rho})], \quad (4.16)$$

so that

$$\mathcal{N}_n = \sum_p \mathcal{T}_p \mathcal{I}(p, p, n) + 2 \sum_p \sum_{q < p} \sqrt{\mathcal{T}_p \mathcal{T}_q} \mathcal{I}(p, q, n). \quad (4.17)$$

Although \mathcal{N}_n is clearly the sum of many random variables, they are not independent, hence we cannot indiscriminately apply the Central Limit Theorem. To say anything about the distribution of the number of received photons, we need a characterization of the coupling coefficient $\mathcal{I}(p, q, n)$, *conditioned* on knowledge of the reference pattern from Eq. (4.4) (all other fluctuations are treated as noise). The true conditional distribution of the coupling coefficient is complicated and not easily derived, so we approximate its mean and variance using the linear minimum mean-squared error (LMMSE) estimator [41].

4.3.1 LMMSE Estimation of the Coupling Coefficient

When estimating a random vector \mathbf{x} from measurements \mathbf{y} of the random vector \mathbf{y} , the LMMSE estimator has the least expected squared-error among all estimators that are linear in the measurements. Given our analytical constraints, in an important sense it is our closest available estimate of the conditional mean $\langle \mathbf{x} | \mathbf{y} \rangle$, which is the unconditional minimum mean-squared error (MMSE) estimator for \mathbf{x} given measurements \mathbf{y} . The general form for the LMMSE estimator is

$$\hat{\mathbf{x}}^{(L)} = \bar{\mathbf{x}} + \mathbf{C}_{\mathbf{xy}} \mathbf{C}_{\mathbf{y}}^{-1} (\mathbf{y} - \bar{\mathbf{y}}), \quad (4.18)$$

where $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ are the means of \mathbf{x} and \mathbf{y} , $\mathbf{C}_{\mathbf{xy}}$ is their cross-covariance matrix, and $\mathbf{C}_{\mathbf{y}}^{-1}$ is the inverse of the autocovariance matrix for \mathbf{y} . The autocovariance matrix of

the estimator's error, $\mathbf{e}(\mathbf{x}) = \mathbf{x} - \widehat{\mathbf{x}}^{(L)}$, is then

$$\mathbf{C}_{\mathbf{e}(\mathbf{x})} = \mathbf{C}_{\mathbf{x}} - \mathbf{C}_{\mathbf{xy}}\mathbf{C}_{\mathbf{y}}^{-1}\mathbf{C}_{\mathbf{yx}}. \quad (4.19)$$

We now seek to estimate the vector of coupling coefficients $\mathcal{I}(p, q, n)$ for $q \leq p$, conditioned on knowledge of the reference matrix \mathbf{I} ; for the prescribed notation this means $\mathbf{x} \equiv \mathcal{I}(p, q, n)$ for all $n, p, q \leq p$, and $\mathbf{y} \equiv \mathbf{I}$, where all terms have been vectorized.

The Estimator

Evaluating the estimator requires computation of the mean and covariance of the reference intensity, the mean of the coupling coefficients, and the covariance of the coupling coefficients and reference, $\mathbf{cov}[\mathcal{I}(p, q, n), I_{n'p'}]$. The mean of the reference pattern, $\bar{\mathbf{I}}$, is calculated in Eq. (4.9), and the remaining quantities are listed below:

$$\mathbf{cov}[I_{np}, I_{n'p'}] = \delta_{nn'}\kappa_1(p, p')\bar{I}_{np}\bar{I}_{n'p'} \quad (4.20)$$

$$\langle \mathcal{I}(p, q, n) \rangle = \delta_{pq}\bar{I}_{np} \quad (4.21)$$

$$\mathbf{cov}[\mathcal{I}(p, q, n), I_{n'p'}] = \delta_{nn'}\delta_{pq}\kappa_\alpha(p, p')\bar{I}_{np}^2. \quad (4.22)$$

Here κ_α captures the discretization error arising from the illumination's irradiance variation over a pixel, viz.

$$\kappa_\alpha(p, p') = \frac{1}{\alpha A_p^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho} \int_{\mathcal{A}_{p'}^2} d\boldsymbol{\rho}' e^{-|\boldsymbol{\rho}-\boldsymbol{\rho}'|^2/\alpha\rho_L^2}, \quad (4.23)$$

where α is the turbulence resolution-degradation factor from our ghost imaging analysis (defined in Eq.(3.12)), and $\kappa_1(p, p')$ is the no-turbulence case for which $\alpha = 1$. Where appropriate we will use the shorthand $\kappa_\alpha = \kappa_\alpha(p, p)$.

To complete the estimator we need to calculate $\mathbf{K}_\alpha\mathbf{K}_1^{-1}$ where \mathbf{K}_α is the matrix composed of elements $\kappa_\alpha(p, p')$. We immediately note that if there is no turbulence, $\mathbf{K}_\alpha = \mathbf{K}_1$ and this matrix product becomes an identity matrix. However, if there

is turbulence it becomes more complicated. Assuming a reasonable pixelation size ($w_p = 1.3\rho_L$), then for pixel p' being the closest neighbor of pixel p , $\kappa_1(p, p')$ has already reached the e^{-1} falloff relative to $\kappa_1(p, p)$. For all other pixels q' , $\kappa_1(p, q') < e^{-2}\kappa_1(p, p)$.

This is initial evidence that the matrix product will be close to an identity matrix. To verify this we computed this matrix product $\mathcal{K}_\alpha \mathcal{K}_1^{-1}$ assuming a 10% loss of resolution due to turbulence. We found that $\mathcal{K}_\alpha \mathcal{K}_1^{-1}$ had diagonal elements with value 0.924, and off-diagonal elements whose largest value was 0.013. Indeed, the sum of the off-diagonal elements was an order of magnitude less than the sum of the diagonal elements. Moreover, $\kappa_\alpha/\kappa_1 = 0.942$, which is very close to the value found for the diagonal elements.

In all that follows we will assume the matrix product $\mathcal{K}_\alpha \mathcal{K}_1^{-1}$ can be approximated with a scaled identity matrix, $\mathbf{1}(\kappa_\alpha/\kappa_1)$. Physically, this approximation is saying that when atmospheric turbulence reduces our knowledge of the on-target irradiance pattern, the neighboring pixels are sufficiently uncorrelated that knowing their value does not help our estimation much more than knowing the average intensity matrix.

Under these conditions the LMMSE estimate becomes

$$\widehat{\mathcal{I}}^{(L)}(p, q, n) = \delta_{pq} \left[\frac{\kappa_\alpha}{\kappa_1} I_{np} + \left(1 - \frac{\kappa_\alpha}{\kappa_1} \right) \bar{I}_{np} \right]. \quad (4.24)$$

We note that all terms where $p \neq q$ are zero-mean, implying that only the diagonal terms $\mathcal{I}(p, p, n)$ contribute to the mean of the received photon number, \mathcal{N}_n . The off-diagonal terms then contribute an additive zero-mean noise term.

Estimator's Error Covariance

To find the error in our estimates we first need to calculate the unconditional covariance of the coupling coefficients. This computation follows the same approach we used in our ghost imaging analysis in Ch. 3. That is, we back-propagate the field patterns with Eq. (2.24) and (2.23), and use the statistical independence of the turbulence, rough surface, and transmitter patterns to separate their averages. Also, we employ

the same assumptions when evaluating the fourth-order moments of the turbulence as we did in Ch. 3. The resulting covariance expression we obtain is then² is

$$\begin{aligned} \mathbf{cov}[\mathcal{I}(p, q, n), \mathcal{I}(p', q', n')] &= \frac{e^{4(\sigma_s^2 + \sigma_r^2)}}{2} \delta_{nn'} [\delta_{pq} \delta_{pp'} \delta_{pq'} \bar{I}_{np}^2 (2\kappa_1 + \Gamma_0(p, p)) + \delta_{pp'} \delta_{qq'} \Gamma_0(p, q) \bar{I}_{np} \bar{I}_{nq}] \\ &+ \delta_{pq} \left[e^{4(\sigma_s^2 + \sigma_r^2)} \delta_{pp'} \delta_{pq'} \Gamma_1(p, p) \bar{I}_{np} \bar{I}_{n'p} + (e^{4(\sigma_s^2 + \sigma_r^2)} - 1) \delta_{p'q'} \bar{I}_{np} \bar{I}_{n'p'} \right], \end{aligned} \quad (4.25)$$

where $\Gamma_0(p, q)$ is the integrated interference pattern from pixels p and q , i.e., the Fourier transform of each pixel shape to difference coordinates at the detector, and its integration over the detector. Similarly, $\Gamma_1(p, q)$ is the Fourier transform of the pixel shapes multiplied by the on-target intensity correlation function,

$$\Gamma_0(p, q) = \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_q^2} d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} \quad (4.26)$$

$$\Gamma_1(p, q) = \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_q^2} d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} e^{-\frac{|\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2|^2}{\rho_L^2}}. \quad (4.27)$$

Where convenient we will use the shorthand $\Gamma_i(p, p) = \Gamma_i$. For specific calculations of these values for square pixels and detectors see App. A.4.

We calculate the error covariance matrix for $\widehat{\mathcal{I}}^{(L)}(p, q, n)$ using Eqs. (4.25), (4.20), and (4.22). However, the elements are functions of 6 variables. To clarify this, we specify the tuple $\mathbf{s} = (p, q, n)$ to index the rows of $\mathbf{C}_{e(\mathcal{I})}$, and $\mathbf{s}' = (p', q', n')$ to specify the column, so that its elements can generally be written as

$$\begin{aligned} (\mathbf{C}_{e(\mathcal{I})})_{\mathbf{ss}'} &= \mathbf{cov}[\mathcal{I}(p, q, n), \mathcal{I}(p', q', n')] - \frac{\kappa_\alpha^2}{\kappa_1} \delta_{nn'} \delta_{pp'} \delta_{qq'} \delta_{pq} \\ &= \frac{e^{4(\sigma_s^2 + \sigma_r^2)}}{2} \bar{I}_{np} \bar{I}_{n'q} \delta_{pp'} \delta_{qq'} [\delta_{pq} (\delta_{nn'} \Gamma_0 + 2\Gamma_1) + \delta_{nn'} \Gamma_0(p, q)] \\ &+ \bar{I}_{np} \bar{I}_{n'p'} \delta_{pq} \delta_{p'q'} \left[\delta_{nn'} \delta_{pp'} \left(e^{4(\sigma_s^2 + \sigma_r^2)} \kappa_1 - \frac{\kappa_\alpha^2}{\kappa_1} \right) + (e^{4(\sigma_s^2 + \sigma_r^2)} - 1) \right]. \end{aligned} \quad (4.28)$$

This covariance matrix is somewhat complicated, but we can break it down into

²Equation (4.25) has been simplified by noting that since $q \leq p$ and $q' \leq p'$, the only way for $p = q'$ and $q = p'$ to be simultaneously true is if $p = q$.

three main cases. First, the estimator's off-diagonal terms $\widehat{\mathcal{I}}^{(L)}(p, q, n)$ (where $q < p$), are uncorrelated with each other, as well as with the estimator's diagonal terms $\widehat{\mathcal{I}}^{(L)}(p, p, n)$, since for $q < p$

$$(\mathbf{C}_{\mathbf{e}(\mathcal{I})})_{ss'} \rightarrow \delta_{nn'} \delta_{pp'} \delta_{qq'} \frac{e^{4(\sigma_s^2 + \sigma_r^2)}}{2} \bar{I}_{np} \bar{I}_{nq} \Gamma_0(p, q). \quad (4.29)$$

Equation (4.29), which evaluates to 0 unless $p = p'$, $q = q'$, and $n = n'$, is initial evidence that the second term in Eq. (4.17) really can be viewed as additive Gaussian noise, independent of the first term in that equation. This conjecture is verified in Sec. 4.4.1 via simulation.

The next main feature of the error covariance matrix in Eq. (4.28) is that, due to the presence of turbulence, there is a weak correlation between the estimator's diagonal terms $\{\widehat{\mathcal{I}}^{(L)}(p, p, n)\}$. Specifically, when $p = q$ and $p' = q'$ but $p \neq p'$, we get

$$(\mathbf{C}_{\mathbf{e}(\mathcal{I})})_{ss'} \rightarrow \bar{I}_{np} \bar{I}_{n'p'} \left(e^{4(\sigma_s^2 + \sigma_r^2)} - 1 \right). \quad (4.30)$$

When turbulence is weak, $e^{4(\sigma_s^2 + \sigma_r^2)} \rightarrow 1$ and this term disappears. Finally, Eq. (4.28) shows that there is an underlying correlation across pulses n for estimates of the diagonal terms $\{\widehat{\mathcal{I}}^{(L)}(p, p, n)\}$, because for $p = q = p' = q'$ but $n \neq n'$,

$$(\mathbf{C}_{\mathbf{e}(\mathcal{I})})_{ss'} \rightarrow \bar{I}_{np} \bar{I}_{n'p} \left[e^{4(\sigma_s^2 + \sigma_r^2)} (\Gamma_1 + 1) - 1 \right].$$

This correlation results from each pulse having the same non-zero intensity envelope ϵ , and the light being scattered by the same pixel on the rough surface. It is a fundamental characteristic of intensity-based reflective imaging, and the source the saturation SNR term seen in our ghost imaging analysis in Eq. (3.29).

Throughout the rest of the thesis, we will make use of this estimate of the coupling coefficient, *conditioned* on the reference. However, explicitly denoting this conditioning throughout the derivations is somewhat tedious. So, anytime an expectation is taken over the coupling coefficients it will be implicitly conditioned on knowledge of the reference. The mean is approximated by Eq. (4.24), while the covariance is taken

from (4.28).

4.3.2 Model of Received Photon Number

So far we have developed a model for estimating the coupling coefficients, but this does not immediately translate into a model for the number of received photons. To start development of that model, we divide \mathcal{N}_n into a signal term \mathbf{z}_n , and a zero-mean additive noise term \mathbf{v}_n , where their designations are justified by the estimation results that we just developed, i.e.,

$$\mathbf{z}_n = \sum_p \mathcal{I}(p, p, n) \mathcal{T}_p \quad (4.31)$$

$$\mathbf{v}_n = 2 \sum_p \sum_{q < p} \sqrt{\mathcal{T}_p \mathcal{T}_q} \mathcal{I}(p, q, n), \quad (4.32)$$

so that

$$\mathcal{N}_n = \mathbf{z}_n + \mathbf{v}_n. \quad (4.33)$$

Both the signal and noise terms are the summation of many random variables, but that is not sufficient to invoke the Central Limit Theorem (CLT) and claim that they are Gaussian distributed. Simple statements of the CLT require the summation of independent, identically distributed (iid) random variables, which we cannot easily show for our case, but there are extensions to sums of dependent random variables that satisfy sufficient conditions. Again, our framework is not amenable to such analysis, but we can make the following observations. First, the randomness in the signal terms is driven by the rough-surface speckle and the light patterns from the transmitter, both of which are are vectors of nearly iid random variables. The noise terms are driven by these same fields, but their randomness is primarily due to by pulse-dependent relative phase shifts³ that do not affect the signal terms. Because all of the different signal and noise terms are primarily driven by different sources, it is not unreasonable to expect their sums to approach Gaussian distributions. To

³For each pulse the phase of the on-target field for different pixels is independent, resulting in a unique random-fading for each interference term $\mathcal{I}(p, q, n)$ for $p \neq q$.

complete this argument we provide numerical simulation results in Sec. 4.4.1 to show that the \mathbf{z} and \mathbf{v} follow independent Gaussian distributions.

Throughout this thesis we will need statistics for \mathbf{z} and \mathbf{v} , sometimes conditioned on knowledge of the target. Fortunately, we can use iterated expectation to find unconditional statistics from conditional ones. For some vector functions $\mathbf{f}(\cdot)$ and $\mathbf{g}(\cdot)$ of a random vector \mathbf{y} that is a function of another random vector \mathbf{x} , the iterated expectation formulas for the mean of $\mathbf{f}(\mathbf{y})$ and covariance of $\mathbf{f}(\mathbf{y})$ and $\mathbf{g}(\mathbf{y})$ are

$$\langle \mathbf{f}(\mathbf{y}) \rangle = \langle \langle \mathbf{f}(\mathbf{y}) | \mathbf{x} \rangle \rangle_{\mathbf{x}} \quad (4.34)$$

$$\text{cov}[\mathbf{f}(\mathbf{y}), \mathbf{g}(\mathbf{y})] = \langle \text{cov}[\mathbf{f}(\mathbf{y}), \mathbf{g}(\mathbf{y}) | \mathbf{x}] \rangle_{\mathbf{x}} + \text{cov}_{\mathbf{x}}[\langle \mathbf{f}(\mathbf{y}) | \mathbf{x} \rangle, \langle \mathbf{g}(\mathbf{y}) | \mathbf{x} \rangle]. \quad (4.35)$$

Below we shall provide just the conditional results (given the target's intensity reflection coefficients \mathcal{T}_p) and will calculate the unconditional statistics in subsequent chapters when we introduce specific distributions for the $\{\mathcal{T}_p\}$.

Mean of the Received Photon Number

The conditional means of the signal and noise terms are

$$\langle \mathbf{z}_n | \mathcal{T} \rangle = \sum_p \hat{\mathcal{I}}^{(L)}(p, p, n) \mathcal{T}_p \quad (4.36)$$

$$\langle \mathbf{v}_n | \mathcal{T} \rangle = 0. \quad (4.37)$$

Therefore the mean of the received photon number is entirely driven by \mathbf{z}_n , and its conditional mean is simply

$$\langle \mathcal{N}_n | \mathcal{T} \rangle = \sum_p \hat{\mathcal{I}}^{(L)}(p, p, n) \mathcal{T}_p. \quad (4.38)$$

Covariance of the Received Photon Number

To find the covariance matrix of the received photon number, we first note that the noise terms are uncorrelated with each other viz.,

$$\begin{aligned} \mathbf{cov}[\mathbf{v}_n, \mathbf{v}_m | \mathcal{T}] &= 4 \sum_{p, p'} \sum_{\substack{q < p \\ q' < p'}} \sqrt{\mathcal{T}_p \mathcal{T}_q \mathcal{T}_{p'} \mathcal{T}_{q'}} \langle \mathcal{I}(p, q, n) \mathcal{I}(p', q', m) \rangle \\ &= \delta_{nm} 2e^{4(\sigma_s^2 + \sigma_r^2)} \sum_p \sum_{q < p} \bar{I}_{np} \bar{I}_{nq} \Gamma_0(p, q) \mathcal{T}_p \mathcal{T}_q, \end{aligned} \quad (4.39)$$

as well as with the signal terms,

$$\mathbf{cov}[\mathbf{z}_n, \mathbf{v}_m | \mathcal{T}] = 2 \sum_{p, p'} \sum_{q < p} \sqrt{\mathcal{T}_p \mathcal{T}_{p'} \mathcal{T}_q} \langle \mathcal{I}(p, p, n) \mathcal{I}(p', q', m) \rangle = 0. \quad (4.40)$$

These results are consequences of Eq. (4.29), which showed that the off-diagonal terms $\mathcal{I}(p, q, n)$ are uncorrelated with each other and uncorrelated with the diagonal terms. Moreover, since $\langle \mathbf{v}_n | \mathcal{T} \rangle = 0$, Eq. (4.35) implies that these terms are also unconditionally uncorrelated, viz.

$$\mathbf{cov}[\mathbf{v}_n, \mathbf{v}_{m \neq n}] = 0 \quad (4.41)$$

$$\mathbf{cov}[\mathbf{z}_n, \mathbf{v}_m] = 0. \quad (4.42)$$

So, because there is no cross-covariance between the signal and noise, we can now find the conditional covariance matrix of \mathbf{N} by adding the conditional autocovariance matrices of \mathbf{z} and \mathbf{v} ,

$$\mathbf{C}_{\mathbf{N} | \mathcal{T}} = \mathbf{C}_{\mathbf{z} | \mathcal{T}} + \mathbf{C}_{\mathbf{v} | \mathcal{T}}. \quad (4.43)$$

The covariance matrix $\mathbf{C}_{\mathbf{v} | \mathcal{T}}$ is a diagonal matrix whose elements are defined by Eq. (4.39). The covariance matrix for \mathbf{z} is more complicated, so we evaluate its diagonal elements (variances of each term) separately from its off-diagonal terms

(cross-covariances). The elements $(\mathbf{C}_{\mathbf{z}|\mathcal{T}})_{nm}$ for $n = m$ are

$$\begin{aligned} \mathbf{var}[\mathbf{z}_n|\mathcal{T}] &= \sum_p \bar{I}_{np}^2 \left[e^{4(\sigma_s^2 + \sigma_r^2)} (\Gamma_0 + \Gamma_1 + \kappa_1) - \frac{\kappa_\alpha^2}{\kappa_1} \right] \mathcal{T}_p^2 \\ &\quad + \sum_{p,q} \bar{I}_{np} \bar{I}_{nq} \left(e^{4(\sigma_s^2 + \sigma_r^2)} - 1 \right) \mathcal{T}_p \mathcal{T}_q, \end{aligned} \quad (4.44)$$

while for off-diagonal terms we have

$$\mathbf{cov}[\mathbf{z}_n, \mathbf{z}_{m \neq n}|\mathcal{T}] = \sum_{p,q} \bar{I}_{np} \bar{I}_{mq} \left[\delta_{pq} e^{4(\sigma_s^2 + \sigma_r^2)} \Gamma_1 + \left(e^{4(\sigma_s^2 + \sigma_r^2)} - 1 \right) \right] \mathcal{T}_p \mathcal{T}_q. \quad (4.45)$$

From inspection we see that the diagonal and off-diagonal elements contain many of the same terms, and can be recombined into one expression for the conditional covariance,

$$\begin{aligned} \mathbf{cov}[\mathbf{z}_n, \mathbf{z}_m|\mathcal{T}] &= \delta_{nm} \sum_p \bar{I}_{np}^2 \left[e^{4(\sigma_s^2 + \sigma_r^2)} (\Gamma_0 + \kappa_1) - \frac{\kappa_\alpha^2}{\kappa_1} \right] \mathcal{T}_p^2 \\ &\quad + \sum_{p,q} \bar{I}_{np} \bar{I}_{mq} \left[\delta_{pq} e^{4(\sigma_s^2 + \sigma_r^2)} \Gamma_1 + \left(e^{4(\sigma_s^2 + \sigma_r^2)} - 1 \right) \right] \mathcal{T}_p \mathcal{T}_q. \end{aligned} \quad (4.46)$$

The preceding covariance expressions can be changed from summations over matrix elements $\{\bar{I}_{np}\}$ to summations over vectors by using the rank-one decomposition of the reference mean matrix from Eq. (4.9). The variance of \mathbf{v}_n reduces to

$$\mathbf{var}[\mathbf{v}_n|\mathcal{T}] = 2e^{4(\sigma_s^2 + \sigma_r^2)} \mu_n^2 \sum_p \epsilon_p \mathcal{T}_p \sum_{q < p} \Gamma_0(p, q) \epsilon_q \mathcal{T}_q. \quad (4.47)$$

For the signal term, this rank-one decomposition lets us reformulate the covariance matrix $\mathbf{C}_{\mathbf{z}|\mathcal{T}}$ as the sum of a diagonal matrix with non-zero elements

$$(\tilde{\mathbf{C}}_{\mathbf{z}|\mathcal{T}})_{nn} = \mu_n^2 \left(e^{4(\sigma_s^2 + \sigma_r^2)} (\Gamma_0 + \kappa_1) - \frac{\kappa_\alpha^2}{\kappa_1} \right) \sum_p \epsilon_p^2 \mathcal{T}_p^2, \quad (4.48)$$

and a rank-one matrix

$$\bar{\mathbf{C}}_{\mathbf{z}|\mathcal{T}} = \boldsymbol{\mu}\boldsymbol{\mu}^T \left[e^{4(\sigma_s^2 + \sigma_r^2)} \Gamma_1 \sum_p \epsilon_p^2 \mathcal{T}_p^2 + \left(e^{4(\sigma_s^2 + \sigma_r^2)} - 1 \right) \left[\sum_p \epsilon_p \mathcal{T}_p \right]^2 \right], \quad (4.49)$$

so that

$$\mathbf{C}_{\mathbf{z}|\mathcal{T}} = \tilde{\mathbf{C}}_{\mathbf{z}|\mathcal{T}} + \bar{\mathbf{C}}_{\mathbf{z}|\mathcal{T}}. \quad (4.50)$$

4.4 Discrete Simulator

From our discretized model we can now construct a computer simulator for structured illumination imaging. We start by generating random phase patterns as an SLM would, and propagate the resultant transmitter field to the target⁴. Once there, we simulate the target’s rough surface by subdividing each pixel into 64 sub-pixels, in which each sub-pixel is an iid complex-Gaussian multiplied by the square-root of the average intensity reflectivity of the pixel. This method approximates the continuous interaction with the rough surface across each pixel. The reflected light is then propagated back to the detector where the magnitude-squared of the field is recorded, appropriately scaled, and used to generate a Poisson process for the measurement outcomes.

For this thesis we are using 100 scenes chosen randomly from the Microsoft Research Cambridge Object Recognition Image Database v1.0 [42]. In all cases the scenes are first cropped to make them rectangular, and then scaled to the dimensions required by the simulation. Since it is not feasible to provide example scene reconstructions for the entire dataset, we have chosen 3 representative scenes to serve as a visual benchmark. The 3 scenes are presented in Fig. 4-1; they were chosen to provide a sufficient variety to demonstrate the performance of the structured-illumination imagers for real-world imaging.

We start with the most basic scene, some cars in a parking lot. This is a low-detail

⁴We derive the best way to simulate this propagation in App. A.2.

scene, but also lets us see how well we can differentiate two similar objects in the same scene. We then consider some flowers, which have a lot of structure and sharp edges. Finally we have a sign, which will allow us to gauge how well text can be read in the structured-illumination images.

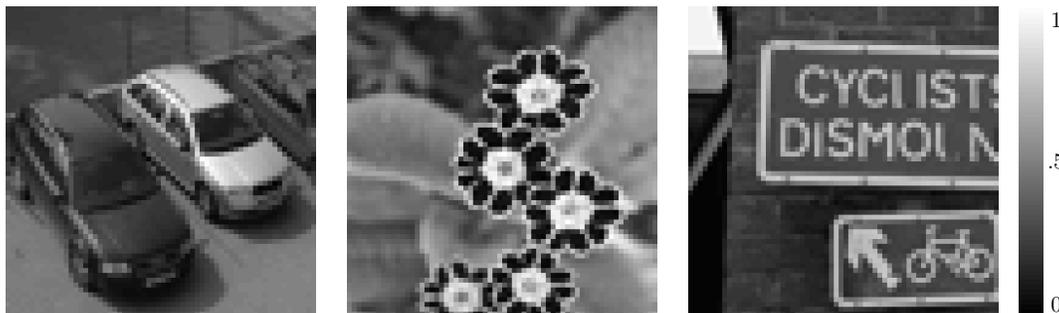


Figure 4-1: Example images that will be used to compare the different estimation methods developed in this thesis.

4.4.1 Statistics of z_n and v_n via Simulation

Beyond letting us test our reconstruction method, the simulator also lets us investigate some of the assumptions we have made in our discretization. Chief among our assumptions is our modeling the signal $\{z_n\}$ and noise $\{v_n\}$ terms as independent Gaussian random variables. To test this assumption we generated 5000 pairs of signal and noise measurements, (z_n, v_n) , by making 100 measurements per scene for 50 of the scenes in our dataset. We then examined the empirical marginal distributions of z and v , and their empirical joint distribution. To combine data across scenes (whose different reflection patterns affect the average detected photon count) we normalized both the signal and noise terms for each scene by the average signal term for that scene. That is, for $\bar{z} = \frac{1}{100} \sum_{n=1}^{100} z_n$ for a particular scene we analyze z_n/\bar{z} and v_n/\bar{z} .

The empirical marginal distributions for these normalized signal and noise terms, shown in Fig. 4-2, are indeed well fit by Gaussian distributions. To rule out any correlation between the signal and noise terms we can analyze the correlation coefficient

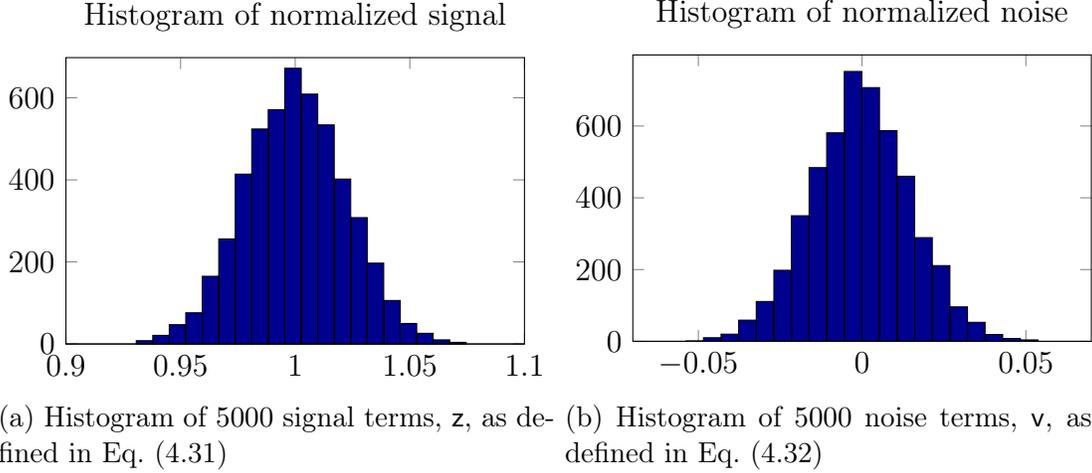


Figure 4-2: Histograms showing the empirical marginal distribution of the signal and noise. 100 z_n and v_n were calculated for 50 different scenes, resulting in 5000 data points. To combine across scenes, both z_n and v_n were normalized by the average signal term for their scene.

of these two vectors as

$$\rho_{z,v} = \frac{\mathbf{cov}[z, v]}{\sqrt{\mathbf{var}[z]\mathbf{var}[v]}}, \quad (4.51)$$

where $\rho_{z,v} = 1$ indicates perfect correlation, $\rho_{z,v} = -1$ indicates perfect anti-correlation, and $\rho_{z,v} = 0$ signifies that the variables are uncorrelated. The (co)variances are computed from the dataset as follows,

$$\mathbf{cov}[z, v] = \frac{1}{N-1} \sum_{n=1}^N (z_n - \bar{z}_n)(v_n - \bar{v}_n) \quad (4.52)$$

$$\mathbf{var}[z] = \frac{1}{N-1} \sum_{n=1}^N (z_n - \bar{z}_n)^2 \quad (4.53)$$

$$\mathbf{var}[v] = \frac{1}{N-1} \sum_{n=1}^N (v_n - \bar{v}_n)^2, \quad (4.54)$$

where the means are calculated over the whole dataset as $\bar{z}_n = \frac{1}{N} \sum_{n=1}^N z_n$ and $\bar{v}_n = \frac{1}{N} \sum_{n=1}^N v_n$. Since we average over 50 scenes, $N = 5000$.

From this computation we find the correlation coefficient to be $\rho_{z,v} = -0.026$, indicating that, in agreement with our previous assertion, there is very little corre-

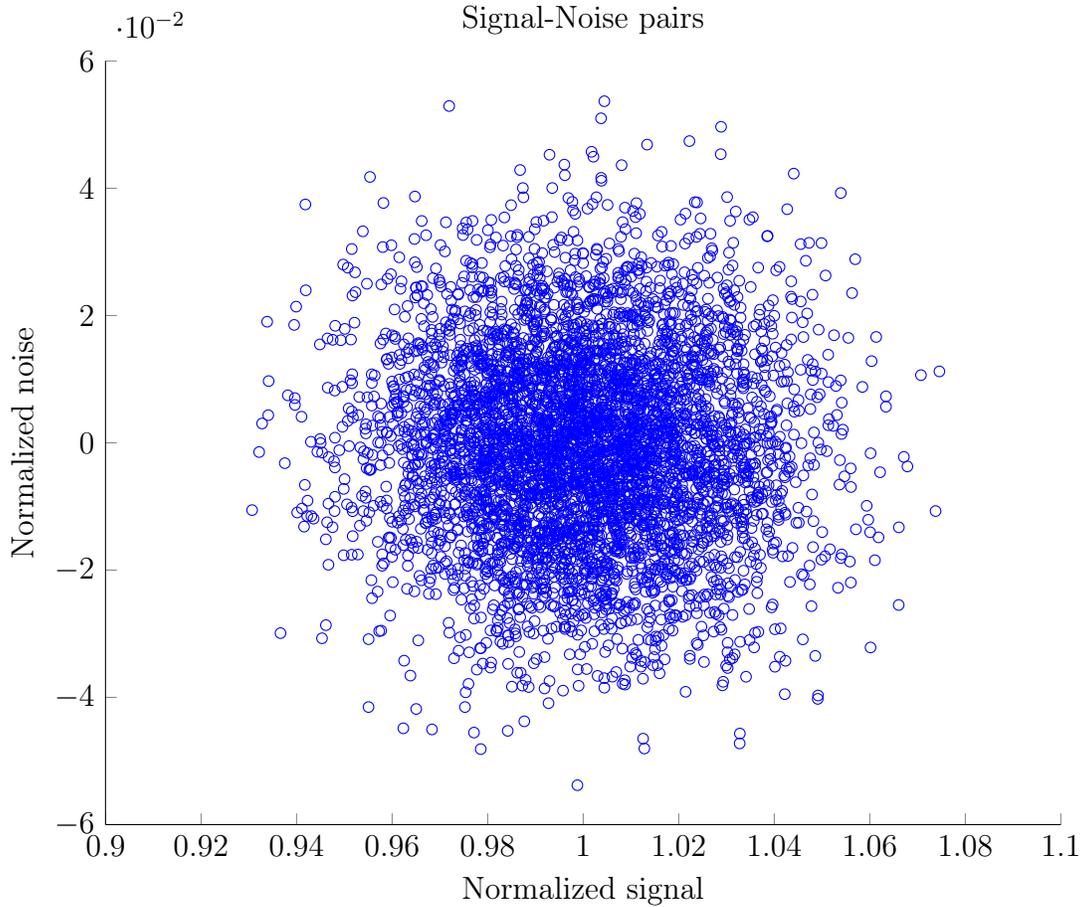


Figure 4-3: A scatter plot of 5000 signal-noise pairs (z_n, v_n) . 100 pairs were recorded for 50 different scenes. All z_n and v_n were normalized by the average of z_n for their respective scene before being plotted.

lation between the signal and the noise. As further evidence, a scatter plot of the signal-noise pairs are provided in Fig. 4-3, and the joint distribution appears to the product of independent Gaussian distributions for the signal and noise.

Chapter 5

Cramér-Rao Bound

We began this thesis by developing a framework for structured-illumination imaging, which we used to analyze ghost imaging. The ghost image reconstruction method—a sample covariance of the bucket detector measurements and the known on-target illumination patterns—was arrived at heuristically in the original experiments [7], and we should not expect it to be the optimal use of the measurements. In the second half of the thesis we explore novel algorithms to make better use of our measurements, but they provide no guarantee they are the best way to use the measurements. Therefore, before developing new image construction methods we would like to establish a bound for the ultimate performance of *any* estimator.

5.1 A Bound on Mean-Squared Error

Herein we derive the Cramér-Rao bound (CRB), which provides a lower bound on the error covariance matrix for all possible unbiased estimators¹ [15]. More formally, for an estimate $\hat{\mathbf{x}}$ of an unknown vector \mathbf{x} from the vector of measurements \mathbf{y} , the CRB is stated as

$$\text{cov}[\hat{\mathbf{x}}, \hat{\mathbf{x}}] \geq \mathbf{J}_{\mathbf{y}}^{-1}(\mathbf{x}), \quad (5.1)$$

¹The bias of an estimator is the expected difference between the estimate and the parameter being estimated. An unbiased estimator \hat{x} for x is one for which $\langle \hat{x} - x \rangle = 0$ for all x .

where $\mathbf{J}_{\mathbf{y}}(\mathbf{x})$ is the Fisher information in \mathbf{y} about \mathbf{x} , and the inequality implies that the matrix $\mathbf{cov}[\hat{\mathbf{x}}, \hat{\mathbf{x}}] - \mathbf{J}_{\mathbf{y}}^{-1}(\mathbf{x})$ is positive semi-definite. For our analysis we are more concerned with bounding the mean-squared error (MSE) of the estimation, which is the trace of the error covariance matrix. However, positive semi-definite matrices have non-negative diagonal elements, so this inequality implies that all of the diagonal elements of $\mathbf{cov}[\hat{\mathbf{x}}, \hat{\mathbf{x}}]$ are no smaller than the corresponding element in the inverse Fisher information, i.e.,

$$(\mathbf{cov}[\hat{\mathbf{x}}, \hat{\mathbf{x}}])_{nn} \geq (\mathbf{J}_{\mathbf{y}}^{-1}(\mathbf{x}))_{nn}. \quad (5.2)$$

The trace of a matrix, $\text{tr}(\cdot)$, is the summation of its diagonal elements, so we are then guaranteed that

$$\text{MSE}(\hat{\mathbf{x}}) = \text{tr}(\mathbf{cov}[\hat{\mathbf{x}}, \hat{\mathbf{x}}]) \geq \text{tr}(\mathbf{J}_{\mathbf{y}}^{-1}(\mathbf{x})). \quad (5.3)$$

To bound the MSE of any possible unbiased estimate we thus need to find the trace of the inverse Fisher information, or when that is not possible a lower bound on it.

Up until this point in the thesis we have always treated the target as a random vector with a prior distribution. The traditional formulation of the CRB in Eq. (5.1) is non-Bayesian, meaning that \mathbf{x} is treated as a deterministic but unknown quantity. In a Bayesian framework we have a conditional distribution of the measurements \mathbf{y} given \mathbf{x} , $p_{\mathbf{y}|\mathbf{x}}(\mathbf{y}|\mathbf{x})$, while in the non-Bayesian framework we have a distribution for the measurements $p_{\mathbf{y}}(\mathbf{y}; \mathbf{x})$ that is parameterized by \mathbf{x} . This parameterized distribution is then related to the conditional distribution by

$$p_{\mathbf{y}}(\mathbf{y}; \mathbf{x}) \equiv p_{\mathbf{y}|\mathbf{x}}(\mathbf{y}|\mathbf{x} = \mathbf{x}). \quad (5.4)$$

We therefore can use the conditional statistics derived in Ch. 4 by replacing the target with \mathbf{x} .

Although the CRB was first derived for non-random parameters, there is a Bayesian version as well. It is an extension of the normal CRB, in which the Bayesian Fisher

information can be calculated from the non-Bayesian Fisher information and the prior information about \mathbf{x} . We therefore first review the CRB as defined in Eq. (5.1), and then extend it to the Bayesian case.

Adapting the CRB to our Imaging Scenario

We are concerned with estimating the target's average intensity-reflection coefficient, so we start by defining $\mathbf{x} \equiv \mathcal{T}$. Our estimates are formed from the photocurrent measurements, so $y_n \equiv i_b(n)$. The Fisher information for this setup is, unfortunately, quite difficult to calculate. We can, however, make a few assumptions that will yield a somewhat looser bound, but make the derivation tractable. First, we will ignore all detection noise, and focus on a bound in which the received number of photons \mathbf{N} is the measurement vector. Next, we discard the additive noise \mathbf{v} so that signal vector \mathbf{z} becomes the measurement vector. Finally we will assume there is no turbulence on the optical paths. Having removed many of the sources of noise, the bound we obtain will be looser than the real CRB, but will approach it when: (1) the number of received photons is high enough that shot-noise can be neglected; (2) the bucket detector is large enough that the noise \mathbf{v} is weak compared to the signal \mathbf{z} ; and (3) turbulence is weak.

We now seek to evaluate the weakened CRB using the Fisher information for \mathbf{z} , $\mathbf{J}_{\mathbf{z}}^{-1}(\mathbf{x})$. For \mathbf{x} having M components, our goal is to calculate the normalized trace

$$\mathcal{C}(\mathbf{x}) = \frac{1}{M} \text{tr}(\mathbf{J}_{\mathbf{z}}^{-1}(\mathbf{x})), \quad (5.5)$$

which lower bounds the normalized MSE of any unbiased estimator via

$$\frac{1}{M} \text{MSE}(\hat{\mathbf{x}}) \geq \mathcal{C}(\mathbf{x}). \quad (5.6)$$

For now on when we refer to the CRB, we mean this bound on the normalized MSE.

5.2 Non-Bayesian Cramér-Rao Bound

The Fisher information in a random vector \mathbf{z} about a non-random parameter \mathbf{x} can be defined in the following two equivalent ways,

$$\begin{aligned} (\mathbf{J}_{\mathbf{z}}(\mathbf{x}))_{mk} &= \left\langle \left(\frac{\partial}{\partial x_m} \log p_{\mathbf{z}}(\mathbf{z}; \mathbf{x}) \right) \left(\frac{\partial}{\partial x_k} \log p_{\mathbf{z}}(\mathbf{z}; \mathbf{x}) \right)^T \right\rangle_{\mathbf{z}} \\ &= - \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \log p_{\mathbf{z}}(\mathbf{z}; \mathbf{x}) \right\rangle_{\mathbf{z}}. \end{aligned} \quad (5.7)$$

Either formula can be more useful for specific distributions, and for our purposes it is convenient to choose the latter. From our previously derived model in Ch. 4, \mathbf{z} is a length- N vector of jointly Gaussian random variables with the probability distribution²

$$p_{\mathbf{z}}(\mathbf{z}; \mathbf{x}) = \frac{e^{-\frac{1}{2}(\mathbf{z}-\bar{\mathbf{z}})^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1}(\mathbf{z}-\bar{\mathbf{z}})}}{(2\pi)^{N/2} |\mathbf{C}_{\mathbf{z}|\mathbf{x}}|^{1/2}}, \quad (5.8)$$

where $|\mathbf{C}_{\mathbf{z}|\mathbf{x}}|$ is the determinant of $\mathbf{C}_{\mathbf{z}|\mathbf{x}}$, so that our log-likelihood is

$$\log p_{\mathbf{z}}(\mathbf{z}; \mathbf{x}) = -\frac{1}{2}(\mathbf{z}-\bar{\mathbf{z}})^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1}(\mathbf{z}-\bar{\mathbf{z}}) - \frac{1}{2} \log [(2\pi)^N |\mathbf{C}_{\mathbf{z}|\mathbf{x}}|]. \quad (5.9)$$

Using the framework developed in Ch. 4, the signal vector \mathbf{z} is defined in Eq. (4.31) as a linear transformation of the target. Its mean is given by Eq. (4.36) as the transformation of the target through the linear estimate $\widehat{\mathcal{I}}^{(L)}(p, p, n)$; for clarity in the derivation to follow we substitute the matrix \mathbf{A} for this linear estimate so that

$$\bar{\mathbf{z}} = \mathbf{A}\mathbf{x}. \quad (5.10)$$

Because we are assuming no turbulence we have that $A_{np} = \widehat{\mathcal{I}}^{(L)}(p, p, n) = I_{np}$, where I_{np} is the reference pattern.

Because expectation and differentiation are linear operations, the Fisher informa-

²The CRB has some weak regularity conditions [15] on $p_{\mathbf{z}}(\mathbf{z}; \mathbf{x})$, but they are satisfied by Eq. (5.8). Namely, $\frac{\partial}{\partial x_j} \log p_{\mathbf{z}}(\mathbf{z}; \mathbf{x})$ exists and is finite, and $p_{\mathbf{z}}(\mathbf{z}; \mathbf{x})$ is continuously differentiable.

tion can be separated as $\mathbf{J}_{\mathbf{z}}(\mathbf{x}) = \mathbf{J}_{\mathbf{z}}^{(1)}(\mathbf{x}) + \mathbf{J}_{\mathbf{z}}^{(2)}(\mathbf{x})$, where

$$(\mathbf{J}_{\mathbf{z}}^{(1)}(\mathbf{x}))_{mk} = \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \frac{1}{2} (\mathbf{z} - \bar{\mathbf{z}})^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} (\mathbf{z} - \bar{\mathbf{z}}) \right\rangle_{\mathbf{z}} \quad (5.11)$$

$$(\mathbf{J}_{\mathbf{z}}^{(2)}(\mathbf{x}))_{mk} = \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \frac{1}{2} \log [(2\pi)^N |\mathbf{C}_{\mathbf{z}|\mathbf{x}}|] \right\rangle_{\mathbf{z}}. \quad (5.12)$$

so that the contribution from the determinant can be evaluated separately.

To evaluate the first term we begin by applying the derivative chain rule to express $(\mathbf{J}_{\mathbf{z}}^{(1)}(\mathbf{x}))_{mk}$ in terms of derivatives of $\bar{\mathbf{z}}$ and the inverse covariance matrix $\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1}$. We then evaluate the first partial derivative of the mean of the signal vector as $\frac{\partial}{\partial x_k} \bar{\mathbf{z}} = \mathbf{A}_k$, where $\mathbf{A}_k \equiv [A_{1k}, A_{2k} \dots A_{Nk}]^T$ is column k from the matrix \mathbf{A} . Applying these transformations, along with the identity $\langle \mathbf{z}_n - \bar{z}_n \rangle = 0$, $\mathbf{J}_{\mathbf{z}}^{(1)}(\mathbf{x})$ reduces to

$$(\mathbf{J}_{\mathbf{z}}^{(1)}(\mathbf{x}))_{mk} = \frac{1}{2} \left\langle (\mathbf{z} - \bar{\mathbf{z}})^T \frac{\partial^2 \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1}}{\partial x_m \partial x_k} (\mathbf{z} - \bar{\mathbf{z}}) \right\rangle_{\mathbf{z}} + \frac{1}{2} \mathbf{A}_k^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \mathbf{A}_m + \frac{1}{2} \mathbf{A}_m^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \mathbf{A}_k \quad (5.13)$$

To calculate the derivatives of the inverse covariance matrix we can repeatedly apply the general formula [43],

$$\frac{\partial}{\partial x} \mathbf{\Lambda}^{-1} = -\mathbf{\Lambda}^{-1} \frac{\partial \mathbf{\Lambda}}{\partial x} \mathbf{\Lambda}^{-1}, \quad (5.14)$$

until we are left with a function of derivatives of the covariance matrix itself,

$$\frac{\partial^2 \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1}}{\partial x_m \partial x_k} = \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_k} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} + \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_k} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} - \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial^2 \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m \partial x_k} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1}. \quad (5.15)$$

We substitute this result back into Eq. (5.13), and for clarity switch from a matrix notation to summations. We use the definition of the covariance matrix $(\mathbf{C}_{\mathbf{z}|\mathbf{x}})_{i,\ell} = \langle (\mathbf{z}_i - \bar{z}_i)(\mathbf{z}_\ell - \bar{z}_\ell) \rangle_{\mathbf{z}}$ when evaluating the expectations, and subsequently make use of the inverse matrix relationship $\sum_i (\mathbf{C}_{\mathbf{z}|\mathbf{x}})_{mi} (\mathbf{C}_{\mathbf{z}|\mathbf{x}})^{-1}_{ik} = \delta_{mk}$. Finally, using the symmetry of the covariance matrix and its inverse to combine equivalent terms,

we arrive at

$$\begin{aligned}
(\mathbf{J}_{\mathbf{z}}^{(1)}(\mathbf{x}))_{mk} &= \sum_{i,\ell} A_{im}(\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1})_{i\ell} A_{\ell k} - \frac{1}{2} \sum_{a,b} (\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1})_{b,a} \frac{\partial^2 (\mathbf{C}_{\mathbf{z}|\mathbf{x}})_{a,b}}{\partial x_m \partial x_k} \\
&+ \sum_{a,b,c,d} \frac{\partial (\mathbf{C}_{\mathbf{z}|\mathbf{x}})_{a,b}}{\partial x_k} (\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1})_{b,c} \frac{\partial (\mathbf{C}_{\mathbf{z}|\mathbf{x}})_{c,d}}{\partial x_m} (\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1})_{d,a}
\end{aligned} \tag{5.16}$$

The second two terms can be rewritten as trace operations, so $\mathbf{J}_{\mathbf{z}}^{(1)}$ can be written more succinctly as

$$(\mathbf{J}_{\mathbf{z}}^{(1)}(\mathbf{x}))_{mk} = \mathbf{A}_m^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \mathbf{A}_k - \frac{1}{2} \text{tr} \left(\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial^2 \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m \partial x_k} \right) + \text{tr} \left(\frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_k} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \right). \tag{5.17}$$

We now turn to the second part of the Fisher matrix from Eq. (5.12), which we can immediately simplify to

$$(\mathbf{J}_{\mathbf{z}}^{(2)}(\mathbf{x}))_{mk} = \frac{1}{2} \frac{\partial^2}{\partial x_m \partial x_k} \log |\mathbf{C}_{\mathbf{z}|\mathbf{x}}|. \tag{5.18}$$

To continue the derivation we need the following two matrix properties. First, the derivative of the logarithm of the determinant of a matrix is the trace of the product of inverse of the matrix and its derivative [43], viz.,

$$\frac{\partial}{\partial x} \log |\mathbf{\Lambda}| = \text{tr} \left(\mathbf{\Lambda}^{-1} \frac{\partial \mathbf{\Lambda}}{\partial x} \right). \tag{5.19}$$

Secondly, the derivative of a trace is the trace of a derivative, so that

$$\frac{\partial}{\partial x} \text{tr}(\mathbf{\Lambda}) = \text{tr} \left(\frac{\partial \mathbf{\Lambda}}{\partial x} \right). \tag{5.20}$$

Taken together, these two properties let us rewrite Eq. (5.18) as

$$(\mathbf{J}_{\mathbf{z}}^{(2)}(\mathbf{x}))_{mk} = \frac{1}{2} \text{tr} \left(\frac{\partial}{\partial x_m} \left[\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_k} \right] \right). \tag{5.21}$$

Applying the chain rule and using Eq. (5.14) for the derivative of the inverse matrix,

Eq. (5.21) becomes

$$\begin{aligned}
(\mathbf{J}_{\mathbf{z}}^{(2)}(\mathbf{x}))_{mk} &= \frac{1}{2} \text{tr} \left(-\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_k} + \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial^2 \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m \partial x_k} \right) \\
&= -\frac{1}{2} \text{tr} \left(\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_k} \right) + \frac{1}{2} \text{tr} \left(\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial^2 \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m \partial x_k} \right)
\end{aligned} \tag{5.22}$$

where in the second line we used the fact that the trace of a sum is the sum of the trace. Both of these terms appear in $\mathbf{J}_{\mathbf{z}}^{(1)}(\mathbf{x})$, so recombining with Eq. (5.17) we find the total Fisher information simplifies to

$$(\mathbf{J}_{\mathbf{z}})_{mk}(\mathbf{x}) = \mathbf{A}_m^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \mathbf{A}_k + \frac{1}{2} \text{tr} \left(\frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_k} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \frac{\partial \mathbf{C}_{\mathbf{z}|\mathbf{x}}}{\partial x_m} \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \right). \tag{5.23}$$

To finish our evaluation we need the partial derivative of $\mathbf{C}_{\mathbf{z}|\mathbf{x}}$, along with the inverse of $\mathbf{C}_{\mathbf{z}|\mathbf{x}}$. These both require an explicit form for $\mathbf{C}_{\mathbf{z}|\mathbf{x}}$, which was derived in Ch. 4 as the sum of a diagonal matrix $\tilde{\mathbf{C}}_{\mathbf{z}|\mathbf{x}}$ and a rank-one matrix $\bar{\mathbf{C}}_{\mathbf{z}|\mathbf{x}}$ that are defined in Eqs. (4.48) and (4.49). We originally derived the CRB for this covariance matrix, but found that in simulations fairly simple estimators could, in some cases, produce an average MSE below what we calculated for the CRB.

The root cause of this apparent conflict is that we estimate the coupling coefficient $\langle \mathcal{I}(p, p, n) | \mathbf{I} \rangle$ with a linear estimator $\hat{\mathcal{I}}^{(L)}(p, p, n)$. This leads to an over-estimation of the discretization error, causing $\text{var}[\mathcal{I}(p, p, n) | \mathbf{I}]$ to be too large, which means that CRB derived for this framework is not guaranteed to be a lower bound. To find a true lower bound, our approach is to remove the discretization process, which is equivalent to assuming there is uniform illumination over each field. This requires re-deriving the covariance matrix, but does not affect the mean under our no-turbulence assumption.

5.2.1 Covariance Matrix for no Discretization Error

To finish calculating the Fisher information we now assume that the target is illuminated in a manner that yields a uniform field over each target pixel³ so that the coupling coefficient $\mathcal{I}(p, p, n)$ seen in Eq. (4.31) becomes the product of the

³We also maintain our previous assumption of no turbulence on the optical paths.

reference pattern and a normalized integrated speckle coefficient for that pixel as $\mathcal{I}(p, p, n) = I_{np}\mathcal{S}_p$, where, for our no-turbulence assumption, the speckle term is

$$\mathcal{S}_p = \frac{1}{\mathcal{F}A_p} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \left| \int_{\mathcal{A}_p} d\boldsymbol{\rho}' \frac{k_0 e^{ik_0|\boldsymbol{\rho}-\boldsymbol{\rho}'|^2/2L}}{i2\pi L} \vartheta(\boldsymbol{\rho}') \right|^2, \quad (5.24)$$

and the signal terms are then

$$z_n = \sum_p I_{np}\mathcal{S}_p x_p. \quad (5.25)$$

This integrated speckle coefficient is more thoroughly investigated in Ch. 9; for now we will just calculate its mean and variance. From its normalization the mean is simply $\langle \mathcal{S}_p \rangle = 1$, while its variance turns out to be $\mathbf{var}[\mathcal{S}_p] = \Gamma_0$. This is the same Γ_0 that was introduced in (4.26), which now has the physical interpretation of the variance of the integrated speckle for a pixel.

As a consequence of the preceding results, the mean of \mathbf{z}_n remains $\bar{z}_n = \sum_p I_{np}x_p$, while the covariance of the signal vector components are $\mathbf{cov}[z_n, z_m | \mathbf{x}] = \Gamma_0 \sum_p I_{np}I_{mp}x_p^2$. We note that the mean matrix is unchanged (under the no-turbulence assumption of this bound), so we maintain $\mathbf{A} = \mathbf{I}$ in the rest of the chapter, making the covariance matrix elements

$$(\mathbf{C}_{\mathbf{z}|\mathbf{x}})_{i\ell} = \Gamma_0 \sum_p A_{ip}x_p^2 A_{\ell p}, \quad (5.26)$$

and the relevant derivative of this covariance matrix is

$$\frac{\partial}{\partial x_k} \mathbf{C}_{\mathbf{z}|\mathbf{x}} = 2\Gamma_0 \mathbf{A}_k x_k \mathbf{A}_k^T. \quad (5.27)$$

Applying this to the Fisher information in Eq. (5.23) we get

$$(\mathbf{J}_{\mathbf{z}}(\mathbf{x}))_{mk} = \mathbf{A}_m^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \mathbf{A}_k + \frac{1}{2} \text{tr} \left(4x_k x_m \Gamma_0^2 \mathbf{A}_k \mathbf{A}_k^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \mathbf{A}_m \mathbf{A}_m^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \right). \quad (5.28)$$

The trace is cyclical, i.e. $\text{tr}(\mathbf{ABC}) = \text{tr}(\mathbf{CAB})$, and the trace of a scalar is just the

scalar [43], so $\mathbf{J}_z(\mathbf{x})$ can be further simplified to

$$(\mathbf{J}_z(\mathbf{x}))_{mk} = \mathbf{A}_m^T \mathbf{C}_{z|x}^{-1} \mathbf{A}_k + 2x_k x_m \Gamma_0^2 \mathbf{A}_k^T \mathbf{C}_{z|x}^{-1} \mathbf{A}_m \mathbf{A}_m^T \mathbf{C}_{z|x}^{-1} \mathbf{A}_k. \quad (5.29)$$

The Fisher information is now dependent on the calculation of $\mathbf{A}_m^T \mathbf{C}_{z|x}^{-1} \mathbf{A}_k$. Although this can be numerically computed, we derive an analytic solution for a special case.

5.2.2 Fisher Information when Oversampling

We consider the system to be oversampled when the number measurements N is greater than or equal to the number of components in the target, i.e., $N \geq M$. The measurement matrix \mathbf{A} has dimensions of $N \times M$, and if the components are selected randomly we can be confident that the columns are linearly independent⁴.

To complete the derivation of $\mathbf{A}_m^T \mathbf{C}_{z|x}^{-1} \mathbf{A}_k$ we make use of the pseudoinverse, specifically the Moore-Penrose pseudoinverse [43]. The pseudoinverse of matrix \mathbf{A} is denoted \mathbf{A}^+ , and for the specific case in which the matrix has linearly independent columns (such as when oversampling) it has the form

$$\mathbf{A}^+ = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T. \quad (5.30)$$

This Moore-Penrose pseudoinverse is also known as the left inverse, since $\mathbf{A}^+ \mathbf{A} = \mathbf{1}$. When the matrix \mathbf{A} is square and invertible, the pseudoinverse becomes inverse, i.e. $\mathbf{A}^+ = \mathbf{A}^{-1}$. Applying Eq. (5.30) to the inverse of the covariance matrix of \mathbf{z} , we can immediately say that

$$\mathbf{C}_{z|x}^{-1} = \mathbf{C}_{z|x}^+ = (\mathbf{A} \mathbf{D}_{\mathcal{X}} \mathbf{A}^T)^+ \quad (5.31)$$

where $\mathbf{D}_{\mathcal{X}}$ is the diagonal matrix whose diagonal vector \mathcal{X} has the elements $\mathcal{X}_m = \Gamma_0 x_m^2$.

In general, the pseudoinverse of the product of matrices \mathbf{A} and \mathbf{B} is $(\mathbf{A}\mathbf{B})^+ =$

⁴In a random matrix we are not guaranteed this will be true, but we could choose the columns pseudo-randomly to maintain the incoherent form of the matrix while guaranteeing their linear independence.

$(\mathbf{A}^+\mathbf{A}\mathbf{B})^+(\mathbf{A}\mathbf{B}\mathbf{B}^+)^+$, but when \mathbf{A} has linearly independent columns and \mathbf{B} has linearly independent rows this simplifies to $(\mathbf{A}\mathbf{B})^+ = \mathbf{B}^+\mathbf{A}^+$. This property, along with the commutative nature of the pseudoinverse and transpose operations, gives us

$$(\mathbf{A}\mathbf{D}_{\mathcal{X}}\mathbf{A}^T)^+ = (\mathbf{A}^+)^T\mathbf{D}_{\mathcal{X}}^+\mathbf{A}^+. \quad (5.32)$$

Since \mathbf{A}^+ is the left inverse of \mathbf{A} , Eq. (5.32) immediately yields

$$\mathbf{A}_m^T\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1}\mathbf{A}_k = \mathbf{A}_m^T(\mathbf{A}^+)^T\mathbf{D}_{\mathcal{X}}^+\mathbf{A}^+\mathbf{A}_k = (\mathbf{D}_{\mathcal{X}}^+)_{mk}. \quad (5.33)$$

Because $\mathbf{D}_{\mathcal{X}}$ is a diagonal matrix it is invertible, i.e., $(\mathbf{D}_{\mathcal{X}}^+)_{mk} = \delta_{mk}\mathcal{X}_m^{-1}$. Substituting back into Eq. (5.29), we get

$$\begin{aligned} (\mathbf{J}_{\mathbf{z}}(\mathbf{x}))_{mk} &= \delta_{mk}\mathcal{X}_m^{-1} + 2x_kx_m\Gamma_0^2\delta_{mk}\mathcal{X}_m^{-2} \\ &= \frac{\Gamma_0^{-1} + 2}{x_m^2}, \end{aligned} \quad (5.34)$$

The Fisher information matrix is diagonal and thus so is its inverse, whose elements become $(\mathbf{J}_{\mathbf{z}}^{-1}(\mathbf{x}))_{mm} = \frac{x_m^2}{2+\Gamma_0^{-1}}$. Thus our bound for this oversampled case is

$$\mathcal{C}(\mathbf{x}) = \frac{\Gamma_0}{2\Gamma_0 + 1} \frac{1}{M} \sum_m x_m^2. \quad (5.35)$$

While this result is very simple, it is to be expected: as posed, this is a system of $N > M$ linear equation for M variables, and if there was no speckle it would be directly solvable. The speckle means there is always some uncertainty in estimation, but in this regime additional measurements don't help. As the speckle variance decreases, so does the bound.

5.2.3 Fisher Information when Undersampling

A system is said to be undersampled where there are $N < M$ measurements made for a target with M components. When oversampling, \mathbf{A} had linearly independent

columns, allowing us to use $(\mathbf{A}\mathbf{D}_{\mathbf{x}}\mathbf{A}^T)^+ = (\mathbf{A}^+)^T\mathbf{D}_{\mathbf{x}}^+\mathbf{A}^+$. With undersampling, \mathbf{A} has linearly independent rows, but dependent columns, so we cannot use the previous approach to solve for the Fisher information. Since the covariance matrix $\mathbf{C}_{\mathbf{z}|\mathbf{x}}$ is invertible, for a particular scene we can numerically compute its inverse. We then calculate the Fisher information in Eq. (5.29), invert the matrix, take the trace, and normalize by M to obtain $\mathcal{C}(\mathbf{x})$ for the undersampled case.

5.2.4 Comparison to Ghost Imaging Theory

The ghost imaging theory developed in Ch. 3 is for a continuous target. Nevertheless, we can make a discrete approximation to enable comparison with the CRB. So long as the pixelation width is approximately a coherence length of the field, the discretization error that is being neglected should not affect the general trend of the results.

We start by noting that the ghost image is a biased non-Bayesian estimator. To compare its performance to the CRB we need to create an unbiased ghost image. Fortunately this just involves scaling the estimate. The average ghost image is given in Eq. (3.11). If we assume the target features are larger than the coherence length of the field and adopt the notation $x_p \equiv \mathcal{T}(\boldsymbol{\rho}_p)$, the expected ghost image at $\boldsymbol{\rho}_p$ simplifies to

$$\langle \mathcal{G}_N(\boldsymbol{\rho}_p) \rangle = \frac{4q\eta\mathbb{N}_s A_b \rho_L^2}{\pi a_L^4 \tau_p' L^2} x_p. \quad (5.36)$$

The unbiased estimator then simply undoes the scaling. We note that this is equivalent to dividing by the expected ghost image when $x_p = 1$, so that our unbiased estimator is

$$\bar{\mathcal{G}}_N(\boldsymbol{\rho}_p) = \frac{\mathcal{G}_N(\boldsymbol{\rho}_p)}{\langle \mathcal{G}_N(\boldsymbol{\rho}_p) | x_p = 1 \rangle}, \quad (5.37)$$

and we now have that $\langle \bar{\mathcal{G}}_N(\boldsymbol{\rho}_p) \rangle = x_p$. The variance of the unbiased ghost image at pixel location $\boldsymbol{\rho}_p$ is then inverse of the SNR equation in Eq. (3.24), with $\mathcal{T}(\boldsymbol{\rho})$ replaced by x_p in the denominator of Eq. (3.24), and by 1 in its numerator.

The CRB was derived under the assumption of no turbulence and no detector noise. Applying those same assumption to the ghost image, the theoretical variance of the unbiased ghost image at $\boldsymbol{\rho}_p$ is

$$\mathbf{var}[\bar{\mathcal{G}}_N(\boldsymbol{\rho}_p)] = \frac{A'_T(1 + \beta^{-1})}{\pi\rho_L^2 N} + x_p^2 \frac{1 + 2(1 + \beta)}{1 + 2\beta}. \quad (5.38)$$

Equation (5.38) is the mean-squared error in estimating the reflectivity of location $\boldsymbol{\rho}_p$ given only the physical parameters of the system (correlation length ρ_L , image effective area A'_T , ratio of transmitter and detector areas β). The MSE of the unbiased ghost image is the sum of Eq. (5.38) over the pixels. For comparison purposes, we are concerned with the MSE normalized by the number of pixels,

$$\mathbb{M}_{\bar{\mathcal{G}}_N} = \frac{1}{M} \sum_p \mathbf{var}[\bar{\mathcal{G}}_N(\boldsymbol{\rho}_p)]. \quad (5.39)$$

This was computed for each scene in our database.

With the theoretical result in hand, we turn to empirically determining the performance of the normalized ghost image via simulation. We run the simulator as described in Sec. 4.4 for each scene in our database, producing a vector of measurements \mathbf{y} and a measurement matrix \mathbf{A} for each run. We then calculate the normalized ghost image for each simulation, and compute the empirical variance of the image,

$$\Delta^2 \bar{\mathcal{G}}_N = \frac{1}{M-1} \sum_p (\bar{\mathcal{G}}_N(\boldsymbol{\rho}_p) - x_p)^2. \quad (5.40)$$

Using the measurement matrices \mathbf{A} produced by this same run of the simulator, we can compute the CRB value $\mathcal{C}(\mathbf{x})$. These three quantities are calculated for every scene, and for two different sizes of bucket detectors: the first is when the detector is the same size as the transmitter ($\beta = 1$), and the second is when the detector is three times the size of the transmitter ($\beta = 3$). We present a comparison of these quantities, averaged over the results from the different scenes, in Fig. 5-1.

We see that ghost-imaging theory is not a good match for the simulation result. This is expected, as the theory was developed for a continuous system, and assumes

Ghost Imaging MSE vs. Cramér-Rao Bound

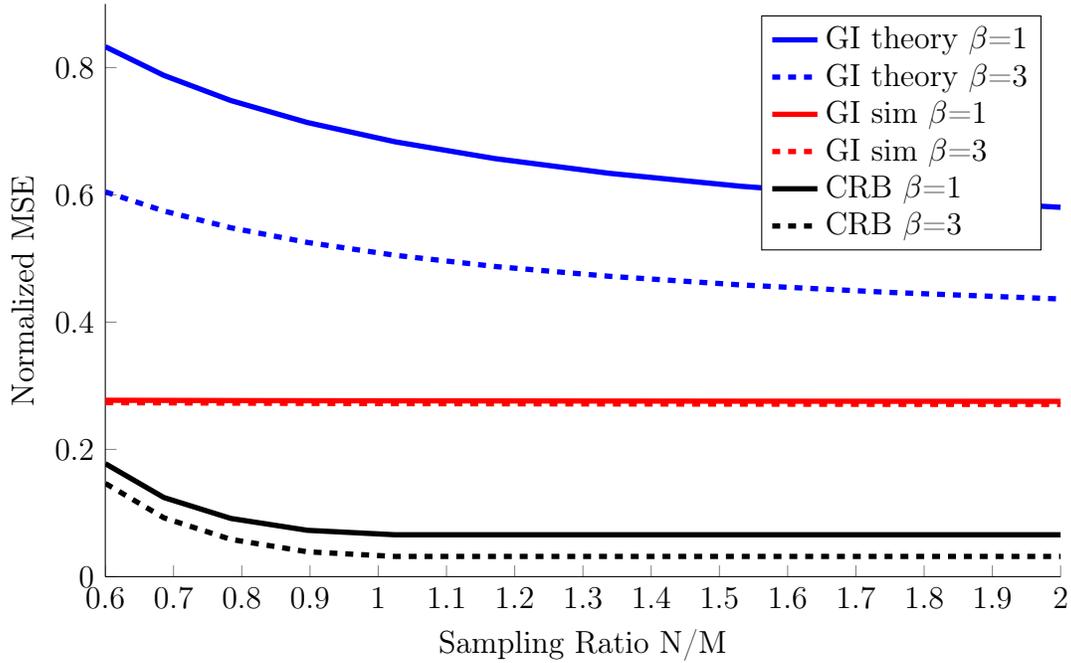


Figure 5-1: Plots versus the sampling ratio (number of measurements over number of pixels) of the ghost image theoretical and empirical variances, with the CRB provided for comparison. Simulation results were averaged over our 100 scene dataset, and each simulation was for a 64×64 scene.

a Gaussian detector. The unexpected result is that the ghost image’s normalized MSE only decreases slightly as the number of measurements increases. Despite this behavior’s indication that more measurements are not helping the ghost image, a look at the generated reconstructions indicates otherwise. The images are very noisy, but subjectively their quality improves as more measurements are used.

To illustrate this improvement, we show two sets of reconstructions in Fig. 5-2. The first has a sampling ratio of 1 (there were as many measurements as pixels), while the second has a sampling ratio of 2. The additional measurements make a noticeable difference in the reconstruction, but in both cases it is very hard to tell what the original scene is (these are presented in Fig. 4-1). At these sampling ratios ghost imaging does not provide a high quality reconstruction. In ghost imaging experiments [29], high quality ghost images are obtained when the number of measurements is an order of magnitude greater than the number of resolution cells, i.e., the sampling ratio

is greater than 10.

However, employing significant oversampling will still not let us perfectly reconstruct the reflectivity profile of the target with ghost imaging. The reflectivity of each pixel is modulated by a speckle coefficient, so that the effective reflectivity measured by the detector is randomly scaled for each pixel. This speckle coefficient, \mathcal{S}_p , is explored in-depth in Ch. 9, but a simplistic version (for uniform illumination and no turbulence) is defined in Eq. (5.24). As a result, the ghost imager will produce a speckled version of the scene. Examples of the speckled reflectivity patterns are shown in Fig. 5-3.

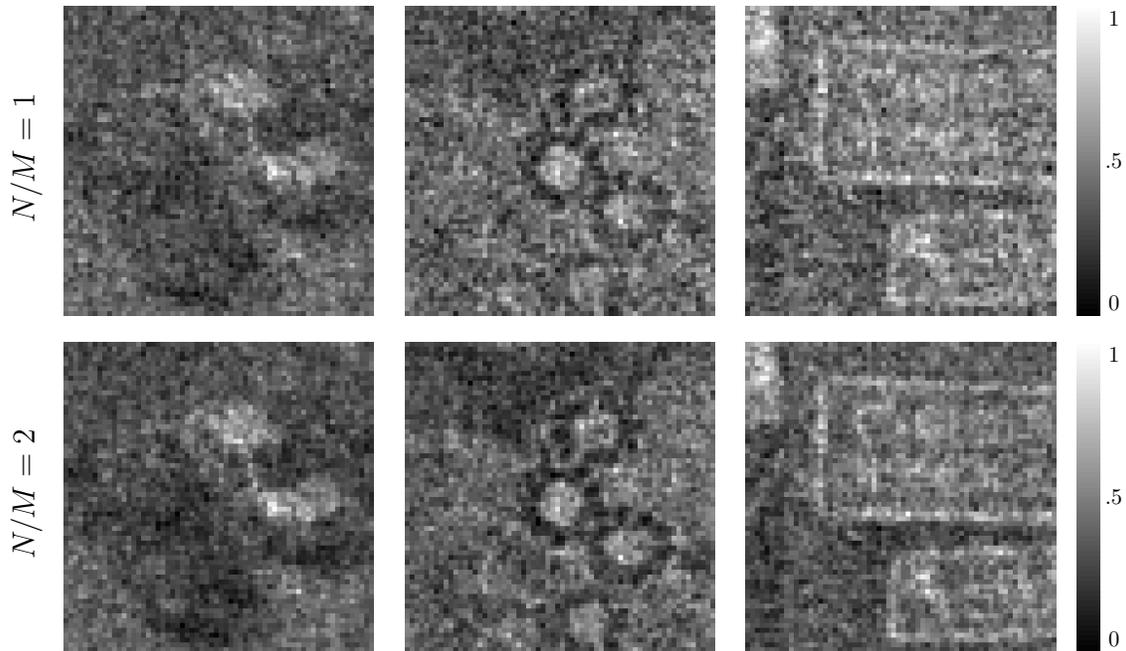


Figure 5-2: Two sets of ghost-image reconstructions for the example scenes in Fig. 4-1. For both sets the bucket detector is three times the size for the transmitter ($\beta = 3$). In the top set the sampling ratio is 1 (the number of measurements N equals the number of pixels M), while for the bottom set the sampling ratio is 2 ($N = 2M$).

5.3 Bayesian Cramér-Rao Bound

The Bayesian Cramér-Rao bound (BCRB) is a fairly straightforward extension of the traditional CRB. It is still the inverse of the Fisher information, but we replace

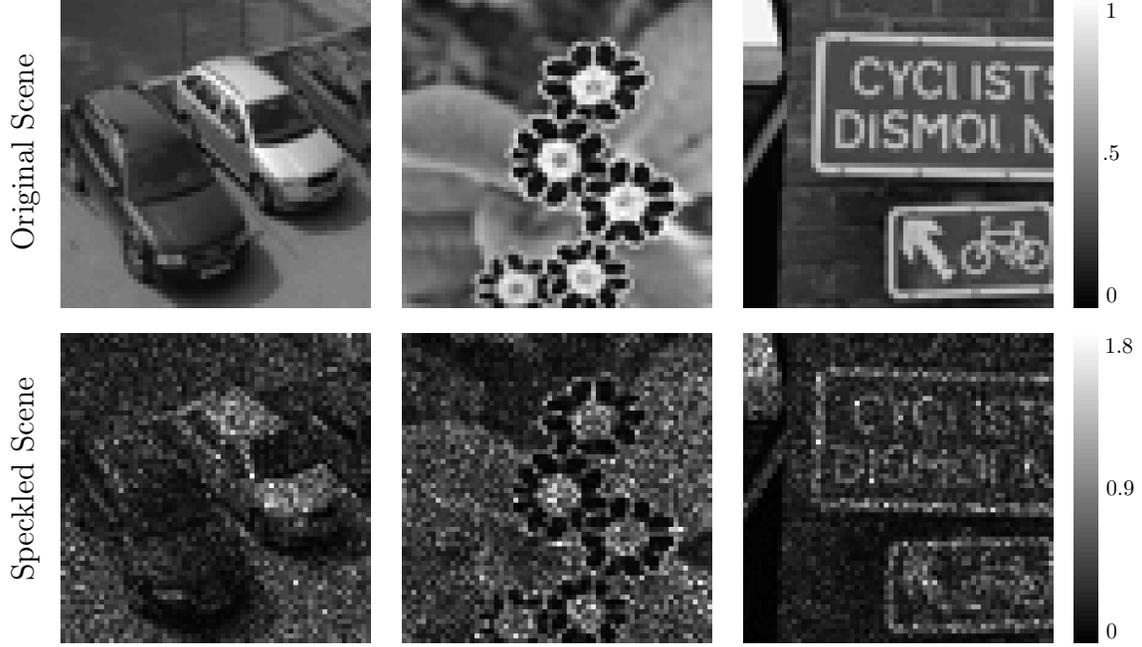


Figure 5-3: The original scenes are presented in the top row, while the speckled reflectivity patterns seen by the detectors (for $\beta = 3$) are presented in the bottom row.

$p_{\mathbf{z}}(\mathbf{z}; \mathbf{x})$, the parameterized distribution of \mathbf{z} , with $p_{\mathbf{z},\mathbf{x}}(\mathbf{z}, \mathbf{x})$, the joint distribution over both \mathbf{z} and \mathbf{x} . The Fisher information is now

$$(\mathbf{J}_{\mathbf{z},\mathbf{x}})_{mk} = - \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \log p_{\mathbf{z},\mathbf{x}}(\mathbf{z}, \mathbf{x}) \right\rangle_{\mathbf{z},\mathbf{x}} \quad (5.41)$$

$$= - \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \log p_{\mathbf{z}|\mathbf{x}}(\mathbf{z}|\mathbf{x}) \right\rangle_{\mathbf{z},\mathbf{x}} - \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \log p_{\mathbf{x}}(\mathbf{x}) \right\rangle_{\mathbf{x}}. \quad (5.42)$$

As before, we can bound the normalized MSE achievable by any Bayesian estimator as

$$\frac{1}{M} \text{MSE}(\hat{\mathbf{x}}) \geq \mathcal{C}_{\mathbf{B}}(\mathbf{x}). \quad (5.43)$$

where the new bound is the normalized trace of the inverse Bayesian Fisher information,

$$\mathcal{C}_{\mathbf{B}}(\mathbf{x}) = \frac{1}{M} \text{tr}(\mathbf{J}_{\mathbf{z},\mathbf{x}}^{-1}), \quad (5.44)$$

The distributions $p_{\mathbf{z}}(\mathbf{z}; \mathbf{x})$ and $p_{\mathbf{z}|\mathbf{x}}(\mathbf{z}|\mathbf{x})$ are functionally the same, except the unknown vector \mathbf{x} has been replaced with the random vector \mathbf{x} , so we define the conditional Fisher information

$$(\mathbf{J}_{\mathbf{z}|\mathbf{x}})_{mk} = - \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \log p_{\mathbf{z}}(\mathbf{z}|\mathbf{x}) \middle| \mathbf{x} \right\rangle_{\mathbf{z}}, \quad (5.45)$$

where the expectation is over the distribution $p_{\mathbf{z}|\mathbf{x}}(\mathbf{z}|\mathbf{x})$. We also have the Fisher information of the prior,

$$(\mathbf{J}_{\mathbf{x}})_{mk} = - \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \log p_{\mathbf{x}}(\mathbf{x}) \right\rangle_{\mathbf{x}}, \quad (5.46)$$

so that using iterated expectations the Bayesian Fisher information is

$$\mathbf{J}_{\mathbf{z},\mathbf{x}} = \langle \mathbf{J}_{\mathbf{z}|\mathbf{x}} \rangle_{\mathbf{x}} + \mathbf{J}_{\mathbf{x}}. \quad (5.47)$$

If we assume the target is composed of pixels that are independently distributed, then the elements of the Fisher information matrix for the prior simplifies to

$$(\mathbf{J}_{\mathbf{x}})_{mk} = - \left\langle \frac{\partial^2}{\partial x_m \partial x_k} \sum_p \log p_{\mathbf{x}}(x_p) \right\rangle_{\mathbf{x}} = -\delta_{mk} \left\langle \frac{\partial^2}{\partial x_m^2} \log p_{\mathbf{x}}(x_m) \right\rangle_{\mathbf{x}}.$$

This is a diagonal matrix whose elements are calculated individually for each pixel. If the target is composed not just of independent pixels, but iid pixels, then each of these calculations is the same, and $\mathbf{J}_{\mathbf{x}}$ is simply a scaled identity matrix.

Choice of Prior Distribution

The choice of prior distribution can greatly affect the resulting BCRB, so we will attempt to match the dataset as closely as possible. We do this by choosing our prior so that its mean and variance match those of the pixels from the dataset, which are $\langle \mathbf{x} \rangle = 0.4633$ and $\mathbf{var}[\mathbf{x}] = 0.0557$. We can quickly rule out several possible distributions based on the constraints of our evaluation. First, we need to satisfy the regularity conditions of the BCRB, which rules out the uniform distribution (deriva-

tive of the log of the distribution is not defined everywhere). In the computation of $\langle \mathbf{J}_{z|x} \rangle_x$ we will need to evaluate $\langle 1/x^2 \rangle$ which eliminates the Gaussian distribution. For the chosen mean and variance, requiring a finite $\langle 1/x^2 \rangle$ also rules out the beta distribution.

The best remaining candidate is the gamma distribution, a strictly positive distribution with a well defined inverse. The gamma distribution is defined by the probability density function

$$p_x(x) = \frac{x^{\varsigma-1} e^{-x/\theta}}{\Gamma(\varsigma)\theta^\varsigma} \quad \text{for } x > 0, \quad (5.48)$$

where ς and θ are parameters that can be calculated from the mean and variance via

$$\varsigma = \frac{\langle x \rangle^2}{\text{var}[x]}, \quad \theta = \frac{\text{var}[x]}{\langle x \rangle}, \quad (5.49)$$

and $\Gamma(\varsigma) = \int_0^\infty dt t^{\varsigma-1} e^{-t}$ is the gamma function. For our mean and variance, $\varsigma = 3.85$ and $\theta = 0.12$.

We model reflectivity as a number from 0 to 1. At 0, no light is reflected, while at 1, all of the light is reflected. The gamma distribution enforces the lower constraint, as its density function is non-zero only for positive numbers, but it does not constrain the reflectivity to be less than our equal to 1. Unfortunately, the distributions we considered that had this constraint were disqualified for other reasons, but for our calculated values of ς and θ , the gamma distribution has 97% of its probability density function between 0 and 1, so we do not expect this lack of constraint to have a significant effect.

The Fisher information for this prior, as calculated from Eq. (5.48), is

$$(\mathbf{J}_x)_{mk} = -\delta_{mk} \left\langle \frac{\partial^2}{\partial x_m^2} \log p_x(x_m) \right\rangle_x = \delta_{mk} \left\langle \frac{\varsigma - 1}{x_m^2} \right\rangle. \quad (5.50)$$

Equation (5.50) requires the computation of $\langle x_m^{-2} \rangle$, which for $\varsigma \geq 2$ (our computed

value is 3.85) is

$$\langle \mathbf{x}_m^{-2} \rangle = \frac{\Gamma(\zeta - 2)}{\theta^2 \Gamma(\zeta)}. \quad (5.51)$$

We are left needing to evaluate $\langle \mathbf{J}_{\mathbf{z}|\mathbf{x}} \rangle_{\mathbf{x}}$ to complete the BCRB. We again split this analysis into the oversampled and undersampled regimes.

5.3.1 Expected Fisher Information when Oversampling

When oversampling, the expected Fisher $\langle (\mathbf{J}_{\mathbf{z}|\mathbf{x}})_{mk} \rangle_{\mathbf{x}}$ is the expectation of Eq. (5.34), which becomes

$$\langle (\mathbf{J}_{\mathbf{z}|\mathbf{x}})_{mk} \rangle_{\mathbf{x}} = \delta_{mk} \left\langle \frac{\Gamma_0^{-1} + 2}{x_m^2} \right\rangle = \delta_{mk} (\Gamma_0^{-1} + 2) \frac{\Gamma(\zeta - 2)}{\theta^2 \Gamma(\zeta)}, \quad (5.52)$$

where the second equality used Eq. (5.51). Combining Eq. (5.52) with the gamma-distribution prior Fisher information from Eq. (5.50) the Bayesian Fisher information is

$$\mathbf{J}_{\mathbf{z},\mathbf{x}} = \delta_{mk} \langle \mathbf{x}_m^{-2} \rangle [\Gamma_0^{-1} + \zeta + 1] = \delta_{mk} \frac{\Gamma(\zeta - 2)}{\theta^2 \Gamma(\zeta)} [\Gamma_0^{-1} + \zeta + 1]. \quad (5.53)$$

This is a diagonal matrix whose inverse is the inverse of the diagonal elements, so the BCRB as defined in Eq. (5.44) is

$$\begin{aligned} \mathcal{C}_B &= \frac{1}{\Gamma_0^{-1} + \zeta + 1} \frac{1}{M} \sum_m \frac{1}{\langle \mathbf{x}_m^{-2} \rangle} \\ &= \frac{1}{\Gamma_0^{-1} + \zeta + 1} \frac{\theta^2 \Gamma(\zeta)}{\Gamma(\zeta - 2)}. \end{aligned} \quad (5.54)$$

5.3.2 Expected Fisher Information when Undersampling

As before, there is no simple solution when undersampling. For the non-Bayesian case we were content to numerically compute the bound in this regime, but now the expectation over \mathbf{x} prevents that. We have made attempts to further lower bound the CRB to obtain analytic results, but these were not useful. Instead, we have elected

to obtain the BCRB via a Monte-Carlo simulation.

Specifically, we numerically simulate the average of the conditional Fisher information from Eq. (5.29),

$$\langle \mathbf{J}_{\mathbf{z}|\mathbf{x}} \rangle_{\mathbf{x}} = \mathbf{A}_m^T \langle \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \rangle_{\mathbf{x}} \mathbf{A}_k + 2\Gamma_0^2 \langle \mathbf{x}_k \mathbf{x}_m \mathbf{A}_k^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \mathbf{A}_m \mathbf{A}_m^T \mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1} \mathbf{A}_k \rangle_{\mathbf{x}}. \quad (5.55)$$

To do this we first generate a vector \mathbf{x} of iid components from our gamma distribution. Then we compute $\mathbf{C}_{\mathbf{z}|\mathbf{x}}^{-1}$, and subsequently $\mathbf{J}_{\mathbf{z}|\mathbf{x}}$. We repeat this process, recording the Fisher information for each iteration. These results are averaged to compute the elements of the expected Fisher information matrix in Eq. (5.55).

However, for a 64×64 scene $\mathbf{J}_{\mathbf{z}|\mathbf{x}}$ is a 4096 by 4096 matrix. Moreover, we will need to multiply and invert matrices of this same order of magnitude for every iteration of the Monte-Carlo simulation. We cannot expect to accurately determine all of the elements in the expected Fisher information matrix from a computationally feasible number of iterations.

Fortunately, we do not need to determine all of the elements of $\langle \mathbf{J}_{\mathbf{z}|\mathbf{x}} \rangle_{\mathbf{x}}$. Rather, we need to compute the BCRB in Eq. (5.44), a low dimensional function of $\langle \mathbf{J}_{\mathbf{z}|\mathbf{x}} \rangle_{\mathbf{x}}$. It turns out this can be numerically approximated with a relatively low number of iterations, and we found the BCRB generally converged in about 100 iterations. Since computation of the BCRB involves inverting $\langle \mathbf{J}_{\mathbf{z}|\mathbf{x}} \rangle_{\mathbf{x}}$, we checked the value of $\mathcal{C}_B(\mathbf{x})$ once every ten iterations of the Monte-Carlo simulation. We considered the bound to have converged when the were three consecutive measurements that were within 0.05% of each other. An example run this convergence can be seen in Fig. 5-4.

Comparison of Bounds

We compare the Bayesian and non-Bayesian Cramér-Rao bounds in Fig. 5-5. The CRB is computed for each scene in our dataset, and the results are averaged over the scenes. The BCRB is computed using the statistical model developed in the preceding sections (directly for the oversampled case, and via Monte-Carlo simulation for the undersampled case). However, it still requires explicit matrices \mathbf{A} , so we again

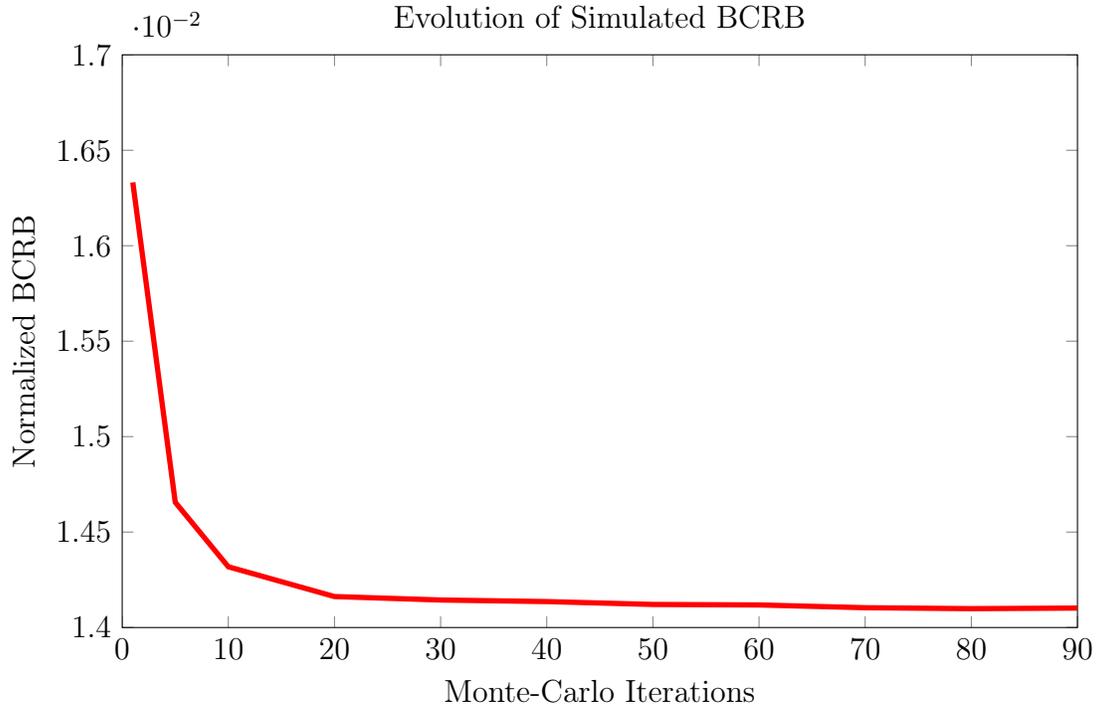


Figure 5-4: A plot of the BCRB, as we accumulate more Monte-Carlo iterations for the computation of $\langle \mathbf{J}_{z|x} \rangle_x$.

computed the bound once for each simulation, and averaged the results. This proved to be unnecessary, as the BCRB calculation showed less an 0.001% difference when computed for different matrices.

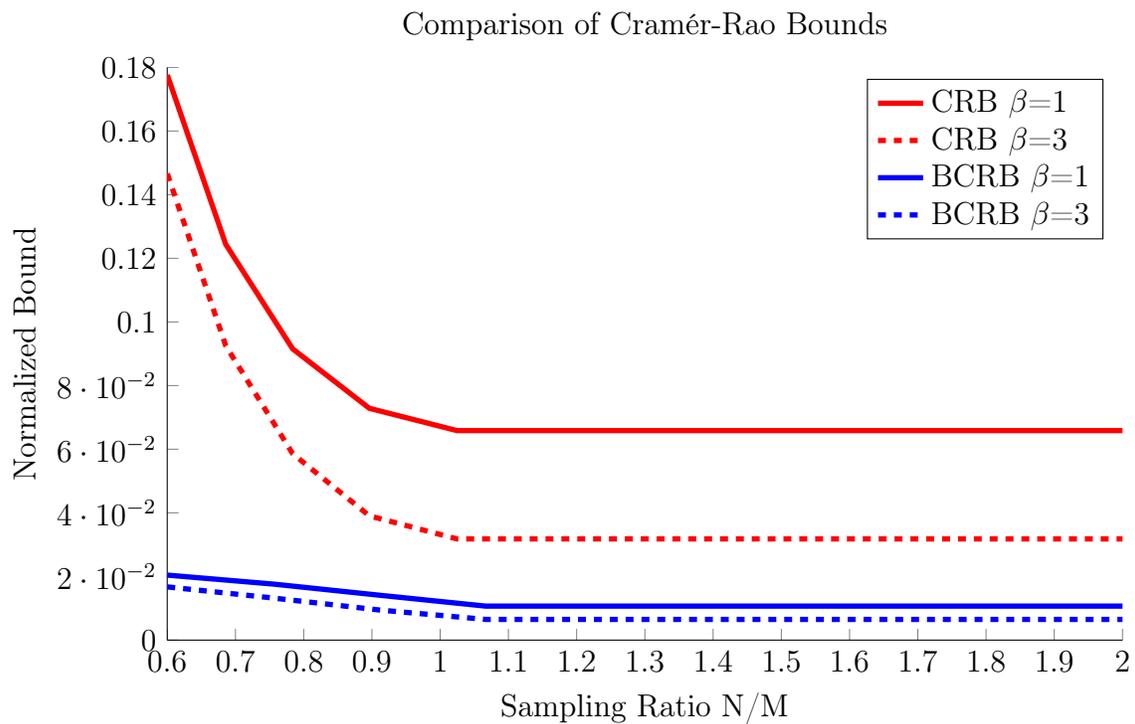


Figure 5-5: Plots versus the sampling rate for the CRB and BCRB, averaged over our 100 scene dataset. Simulation was for a 64×64 scene, for both $\beta = 1$ and $\beta = 3$.

Chapter 6

Linear Minimum Mean-Squared Error Estimator

We began this thesis by building a framework for structured illumination imaging, and then used it to analyze an existing reconstruction method called ghost imaging. We then took the basic structured-illumination framework and discretized it. From this discretized model we derived ultimate performance limits in terms of the mean-squared error (MSE) with the Cramér-Rao bound (CRB), which is our performance benchmark when deriving new estimators.

We begin deriving new estimators now, starting with the linear minimum mean-squared error (LMMSE) estimator [41]. Ghost imaging is a linear estimator, but it was arrived at heuristically. The LMMSE estimator has the minimum expected squared error among all estimators that are linear in the measurements. It is a central tool of estimation theory, and was employed in the development of our discretization model in Ch. 4. As this chapter relies heavily on LMMSE estimation theory, we reproduce the formula for the LMMSE estimate and its error covariance matrix,

$$\hat{\mathbf{x}}^{(L)} = \bar{\mathbf{x}} + \mathbf{C}_{\mathbf{xy}}\mathbf{C}_{\mathbf{y}}^{-1}(\mathbf{y} - \bar{\mathbf{y}}) \quad (6.1)$$

$$\mathbf{C}_{\mathbf{e}(\mathbf{x})} = \mathbf{C}_{\mathbf{x}} - \mathbf{C}_{\mathbf{xy}}\mathbf{C}_{\mathbf{y}}^{-1}\mathbf{C}_{\mathbf{yx}}, \quad (6.2)$$

which were originally presented in Eqs. (4.18) and (4.19). In this formalization we are

estimating the random vector \mathbf{x} from the measurements \mathbf{y} , using their prior means $\bar{\mathbf{y}}$ and $\bar{\mathbf{x}}$, their cross-covariance matrix $\mathbf{C}_{\mathbf{xy}}$, and the measurements' autocovariance matrix $\mathbf{C}_{\mathbf{y}}$. The covariance matrix of the estimator error, $\mathbf{e}(\mathbf{x}) = \mathbf{x} - \widehat{\mathbf{x}}^{(L)}$, also requires the autocovariance matrix $\mathbf{C}_{\mathbf{x}}$. We maintain the notation from Ch. 5 of $x_p \equiv \mathcal{T}_p$ but we now define $y_n = \frac{\tau'_p}{q\eta} i_b(n)$ —where the measurements $i_b(n)$ are defined by Eq. (2.28)—so that

$$y_n = \mathcal{N}_n + \frac{\tau'_p}{q\eta} \Delta i_b(n). \quad (6.3)$$

6.1 Unconditional Mean Values

The mean of the target, $\bar{\mathbf{x}}$, is defined by the statistical model for the target, and does not need to be calculated. To find the unconditional mean of the measurements we use iterated expectation over the number of received photons,

$$\langle \mathbf{y} \rangle = \langle \langle \mathbf{y} | \mathcal{N} \rangle \rangle_{\mathcal{N}} = \langle \mathcal{N} \rangle = \langle \mathbf{z} \rangle + \langle \mathbf{v} \rangle \quad (6.4)$$

In Ch. 4 we showed that \mathbf{v} was comprised of zero-mean random variables, while \mathbf{z} is a jointly-Gaussian random vector with mean $\bar{\mathbf{z}}$, so the measurement mean is

$$\langle \mathbf{y} \rangle = \bar{\mathbf{z}}. \quad (6.5)$$

The elements of this mean vector were originally defined in Eq. (4.36); for clarity we reproduce the result here, substituting¹ $A_{np} = \widehat{\mathcal{I}}^{(L)}(p, p, n)$, viz.,

$$\bar{z}_n = \sum_p A_{np} \bar{x}_p. \quad (6.6)$$

¹This is the same notational convenience we used in Ch. 5

6.2 Target and Measurement Cross-Covariance

We continue developing the linear estimator by now deriving the cross-covariance of the target \mathbf{x} and the measurements \mathbf{y} using the iterated expectation formula from Eq. (4.35), viz.,

$$(\mathbf{C}_{\mathbf{xy}})_{pn} = \mathbf{cov}[x_p, y_n] = \langle \mathbf{cov}[x_p, y_n | x_p, \mathcal{N}_n] \rangle + \mathbf{cov}[\langle x_p | \mathcal{N}_n \rangle, \langle y_n | \mathcal{N}_n \rangle] \quad (6.7)$$

$$= \langle x_p \mathcal{N}_n - x_p \mathcal{N}_n \rangle + \mathbf{cov}[x_p, \mathcal{N}_n] \quad (6.8)$$

$$= \mathbf{cov}[x_p, z_n] + \mathbf{cov}[x_p, v_n]. \quad (6.9)$$

This is simply the sum of the covariances of the target pixel with the signal and noise. From our definition of the noise in Eq. (4.32), we have

$$\mathbf{cov}[x_p, v_n] = 2 \sum_{p'} \sum_{q' < p'} \langle \sqrt{x_{p'} x_{q'}} x_p \rangle \langle \mathcal{I}(p', q', n) \rangle = 0, \quad (6.10)$$

and we find the noise is uncorrelated with the target. The signal \mathbf{z} is correlated with the target, however, which we show using its definition from Eq. (4.31),

$$\mathbf{cov}[x_p, z_n] = \sum_q A_{nq} \langle x_q x_p \rangle - \sum_q A_{nq} \langle x_q \rangle \langle x_p \rangle = A_{np} \mathbf{var}[x_p], \quad (6.11)$$

where, as in our CRB analysis in Ch. 5, we have assumed the $\{x_p\}$ are independent.

The elements of the cross-covariance matrix of the target and measurements are then simply

$$(\mathbf{C}_{\mathbf{xy}})_{pn} = A_{np} \mathbf{var}[x_p], \quad (6.12)$$

which is equivalent to $(\mathbf{C}_{\mathbf{xz}})_{pn}$. To put this in matrix form we note that since we assume an iid prior, the covariance matrix of the target $\mathbf{C}_{\mathbf{x}}$ is simply a diagonal matrix with elements $\mathbf{var}[x_p]$, and thus Eq. (6.12) becomes

$$\mathbf{C}_{\mathbf{xy}} = \mathbf{C}_{\mathbf{xz}} = \mathbf{C}_{\mathbf{x}} \mathbf{A}^T. \quad (6.13)$$

6.3 Measurement Autocovariance

The autocovariance matrix of the measurements \mathbf{C}_y is also found using iterated expectation,

$$\begin{aligned}
(\mathbf{C}_y)_{nm} &= \mathbf{cov}[y_n, y_m] = \langle \mathbf{cov}[y_n, y_m | \mathcal{N}_n, \mathcal{N}_m] \rangle + \mathbf{cov}[\langle y_n | \mathcal{N}_n \rangle, \langle y_m | \mathcal{N}_m \rangle] \\
&= \delta_{nm} \langle \mathbf{var}[y_n | \mathcal{N}_n] \rangle + \mathbf{cov}[\mathcal{N}_n, \mathcal{N}_m] \\
&= \delta_{nm} \frac{\tau_p'^2}{\eta \tau_p''^2} \langle \mathcal{N}_n \rangle + \mathbf{cov}[\mathbf{z}_n + \mathbf{v}_n, \mathbf{z}_m + \mathbf{v}_m] \\
&= \delta_{nm} \left(\frac{\tau_p'^2}{\eta \tau_p''^2} \bar{z}_n + \mathbf{var}[\mathbf{v}_n] \right) + (\mathbf{C}_z)_{nm}. \tag{6.14}
\end{aligned}$$

\mathbf{C}_y is now the sum of a diagonal matrix and \mathbf{C}_z . The diagonal matrix contains the additional variance accrued from the additive noise term and the detection process, which are uncorrelated across the measurements. The remaining covariance is from the randomness inherent in the signal vector \mathbf{z} .

The signal covariance matrix \mathbf{C}_z will appear throughout this thesis, so we take some time to develop and simplify it now. First, using the iterated expectation formula from Eq. (4.35), the elements of \mathbf{C}_z become

$$\begin{aligned}
(\mathbf{C}_z)_{nm} &= \langle (\mathbf{C}_{z|\mathbf{x}})_{nm} \rangle + \mathbf{cov}_{\mathbf{x}}[\langle z_n | \mathbf{x} \rangle, \langle z_m | \mathbf{x} \rangle] \\
&= \langle (\mathbf{C}_{z|\mathbf{x}})_{nm} \rangle + \sum_p \sum_q A_{np} A_{mq} \langle x_p x_q \rangle - \sum_p A_{np} \bar{x}_p \sum_q A_{mq} \bar{x}_q \\
&= \langle (\mathbf{C}_{z|\mathbf{x}})_{nm} \rangle + \sum_p A_{np} A_{mp} \mathbf{var}[x_p]. \tag{6.15}
\end{aligned}$$

To evaluate the first term we use the decomposition of the conditional covariance $\mathbf{C}_{z|\mathbf{x}}$ from Eq. (4.50) so that $\langle \mathbf{C}_{z|\mathbf{x}} \rangle = \langle \tilde{\mathbf{C}}_{z|\mathbf{x}} \rangle + \langle \bar{\mathbf{C}}_{z|\mathbf{x}} \rangle$, where $\tilde{\mathbf{C}}_{z|\mathbf{x}}$ is diagonal and $\bar{\mathbf{C}}_{z|\mathbf{x}}$ is rank-one. These averages become

$$\left\langle (\tilde{\mathbf{C}}_{z|\mathbf{x}})_{nm} \right\rangle = \delta_{nm} \mu_n^2 \left(e^{4(\sigma_s^2 + \sigma_r^2)} (\Gamma_0 + \kappa_1) - \frac{\kappa_\alpha^2}{\kappa_1} \right) \sum_p \epsilon_p^2 (\mathbf{var}[x_p] + \bar{x}_p^2), \tag{6.16}$$

and

$$\begin{aligned} \langle (\bar{\mathbf{C}}_{\mathbf{z}|\mathbf{x}})_{nm} \rangle &= \boldsymbol{\mu}\boldsymbol{\mu}^T e^{4(\sigma_s^2 + \sigma_r^2)} \Gamma_1 \sum_p \epsilon_p^2 (\mathbf{var}[x_p] + \bar{x}_p^2) \\ &\quad + \boldsymbol{\mu}\boldsymbol{\mu}^T \left(e^{4(\sigma_s^2 + \sigma_r^2)} - 1 \right) \left(\sum_p \epsilon_p^2 \mathbf{var}[x_p] + \left[\sum_p \epsilon_p \bar{x}_p \right]^2 \right). \end{aligned} \quad (6.17)$$

The second term in Eq. (6.15) can also be simplified. Because we have defined $A_{np} \equiv \widehat{\mathcal{I}}^{(L)}(p, p, n)$, \mathbf{A} can be decomposed into the sum of a zero-mean iid matrix $\tilde{\mathbf{A}}$ and a rank-one background matrix $\bar{\mathbf{A}}$, where $\bar{\mathbf{A}} = \bar{\mathbf{I}} = \boldsymbol{\mu}\boldsymbol{\epsilon}^T$. Expanding \mathbf{A} in the sum $\sum_p A_{np} A_{mp} \mathbf{var}[x_p]$, we find

$$\sum_p A_{np} A_{mp} \mathbf{var}[x_p] = \sum_p \tilde{A}_{np} \tilde{A}_{mp} \mathbf{var}[x_p] + \mu_n \sum_p \epsilon_p \tilde{A}_{mp} \mathbf{var}[x_p] \quad (6.18)$$

$$+ \mu_m \sum_p \epsilon_p \tilde{A}_{np} \mathbf{var}[x_p] + \mu_n \mu_m \sum_p \epsilon_p^2 \mathbf{var}[x_p]. \quad (6.19)$$

Because $\tilde{\mathbf{A}}$ is composed of iid random variables, the middle two terms become insignificant compared to the final term by the strong law of large numbers. Therefore,

$$\sum_p A_{np} A_{mp} \mathbf{var}[x_p] \xrightarrow{w.p.1} \sum_p \tilde{A}_{np} \tilde{A}_{mp} \mathbf{var}[x_p] + \mu_n \mu_m \sum_p \epsilon_p^2 \mathbf{var}[x_p]. \quad (6.20)$$

where $\xrightarrow{w.p.1}$ means it converges *almost surely*, or with probability one. Recombining Eqs. (6.16), (6.17), and (6.20), the covariance matrix for \mathbf{z} becomes the sum of three matrices,

$$\mathbf{C}_{\mathbf{z}} = \langle \tilde{\mathbf{C}}_{\mathbf{z}|\mathbf{x}} \rangle + \boldsymbol{\mu}\boldsymbol{\mu}^T \Omega + \tilde{\mathbf{A}} \mathbf{C}_{\mathbf{x}} \tilde{\mathbf{A}}^T, \quad (6.21)$$

where the scaling factor for the rank-one matrix, Ω , is

$$\Omega = e^{4(\sigma_s^2 + \sigma_r^2)} \sum_p \epsilon_p^2 (\Gamma_1 (\mathbf{var}[x_p] + \bar{x}_p^2) + \mathbf{var}[x_p]) + \left(e^{4(\sigma_s^2 + \sigma_r^2)} - 1 \right) \left(\sum_p \epsilon_p \bar{x}_p \right)^2. \quad (6.22)$$

Using Eqs. (6.14) and (6.21), we can also specify the covariance matrix for \mathbf{y} as the sum of three matrices,

$$\mathbf{C}_y = \mathcal{D} + \boldsymbol{\mu}\boldsymbol{\mu}^T\Omega + \tilde{\mathbf{A}}\mathbf{C}_x\tilde{\mathbf{A}}^T, \quad (6.23)$$

where the diagonal matrix \mathcal{D} has diagonal elements

$$\mathcal{D}_{nn} = \frac{\tau_p'^2}{\eta\tau_p'^2}\bar{z}_n + \mathbf{var}[v_n] + \left\langle (\tilde{\mathbf{C}}_{\mathbf{z}|\mathbf{x}})_{nn} \right\rangle. \quad (6.24)$$

The sum of a rank-one matrix plus a diagonal matrix can be analytically inverted, but the remaining matrix in \mathbf{C}_y needs to be inverted computationally. Because we have to compute an inverse, it is simplest to compute the inverse of the entire matrix \mathbf{C}_y . We now have all of the necessary matrices to compute the linear estimate and compare its performance with the ghost image and CRB.

6.4 Theoretical and Simulated Performance

6.4.1 Comparison of Theory and Simulation

The theoretical error for the LMMSE estimate is computed in Eq. (6.2). This calculation is based on our model, which presumes that the target pixels follow an iid distribution. This is a poor model for real-world scenes, whose pixels are correlated. Indeed, part of our work in the second half of the thesis involves constructing a better target model to improve estimation performance. This model mismatch means our theoretical mean-squared error becomes less useful, and is more of a guide than an accurate prediction of the performance. To test this presumption, we use the dataset produced by our simulator in Ch. 5 for our CRB and the ghost imaging analysis, and process the results with the LMMSE estimator. A plot of the expected average variance per pixel, as well as the empirical average variance in the reconstruction, are presented in Fig. 6-1. The variances are calculated by computing the MSE for each image, normalizing by the number of pixels, and averaging the results across all of

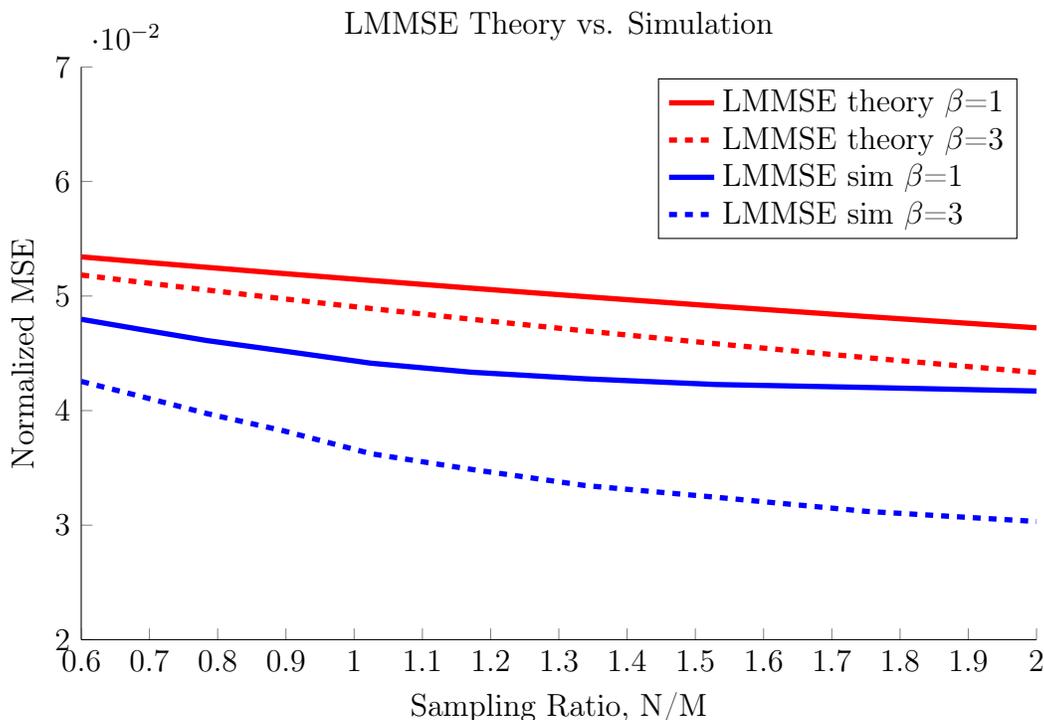


Figure 6-1: Plots versus the sampling ratio of the normalized mean-squared error as predicted by Eq. (6.2), and the empirical mean-squared error from reconstructions calculated with our simulator.

the scenes in the dataset. As expected, the simulation produces a lower variance than predicted by the theory.

6.4.2 Comparison to the Cramér-Rao Bound

The LMMSE estimator is a Bayesian estimator, but is not, in general, the same as the unconstrained MMSE. However, when the measurements \mathbf{y} and variables \mathbf{x} are described by a jointly Gaussian distribution the LMMSE estimator is the unconstrained MMSE estimator. Moreover, it is then *efficient*, i.e., its estimation error reaches the CRB. For a sufficient number of photons the filtered Poisson distribution that drives the measurements approaches a Gaussian distribution, but the target is not accurately described as an independent, identically distributed (iid) Gaussian distribution, so we do not expect LMMSE performance to reach the CRB.

In Fig. 6-2 the Bayesian and non-Bayesian Cramér-Rao bounds are compared

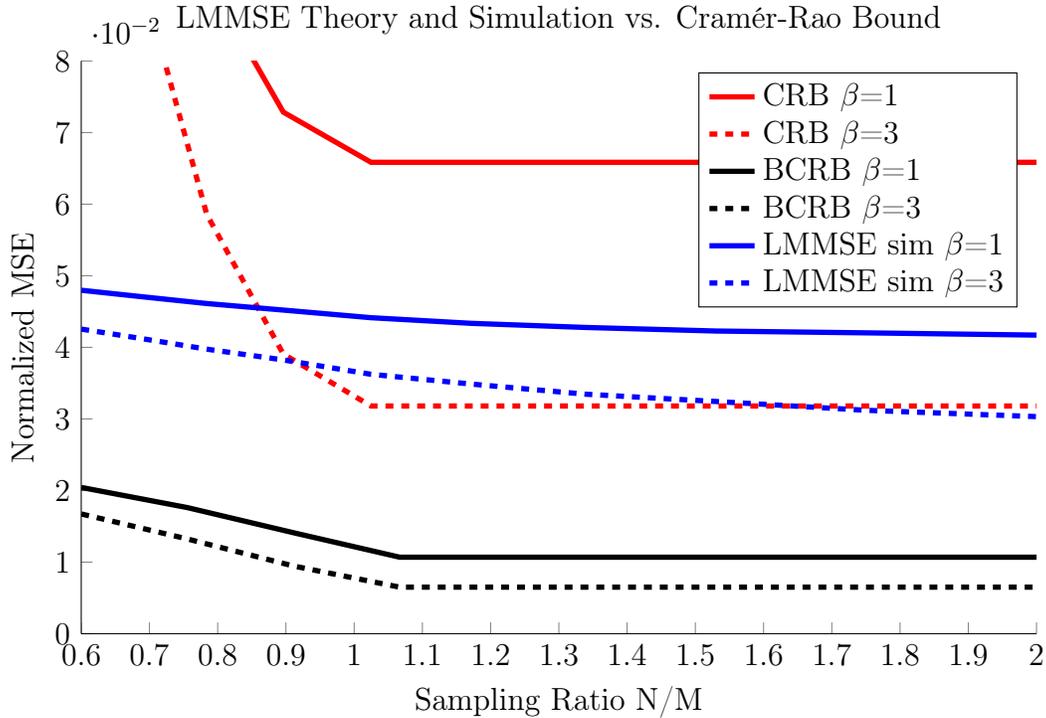


Figure 6-2: Plots versus the sampling ratio of the Cramér-Rao bounds from Eqs. (5.5) and (5.44), and the empirical mean-squared error from reconstructions calculated with our simulator.

to the normalized MSE produced from processing our simulation dataset with the LMMSE estimator. Although the mean-squared error for the LMMSE estimator does fall short of the BCRB, it surpasses the non-Bayesian CRB (and by association the ghost image MSE, as shown in Fig. 5-1). The same simulation data was used to compute all three quantities

Comparison of LMMSE Simulation to Ghost Imaging Performance

The LMMSE estimate has significantly lower MSE than the ghost image, but that is not a fair comparison. MSE is not a fool-proof error metric for image analysis, as human perception is more concerned with relative structure in the image than pixel-wise noise, or global shifts or scalings. These latter two can have profound effects on the MSE of an image without affecting human perception. Indeed, this disparity between subjective and objective image metrics accounts for most of the difference in the MSEs of the LMMSE estimate and the ghost image. We presented

a series of ghost image reconstructions in Fig. 5-2, and for comparison we present the same images reconstructed with the LMMSE estimate in Fig. 6-3. We again see that additional measurements aid the reconstruction, but they are still very noisy. Finally, in Fig. 6-4, we present a side-by-side comparison of the LMMSE estimate and ghost image for two scenes. The two images are close enough that it is hard to say which one is a better reconstruction. This is strong evidence that, although it might produce a lower mean-squared error than the ghost image, the LMMSE estimator falls short of our needs for a high-quality reconstruction. If we want to produce high-quality images, we will need more advanced algorithms. Development of such algorithms begins in Ch. 7.

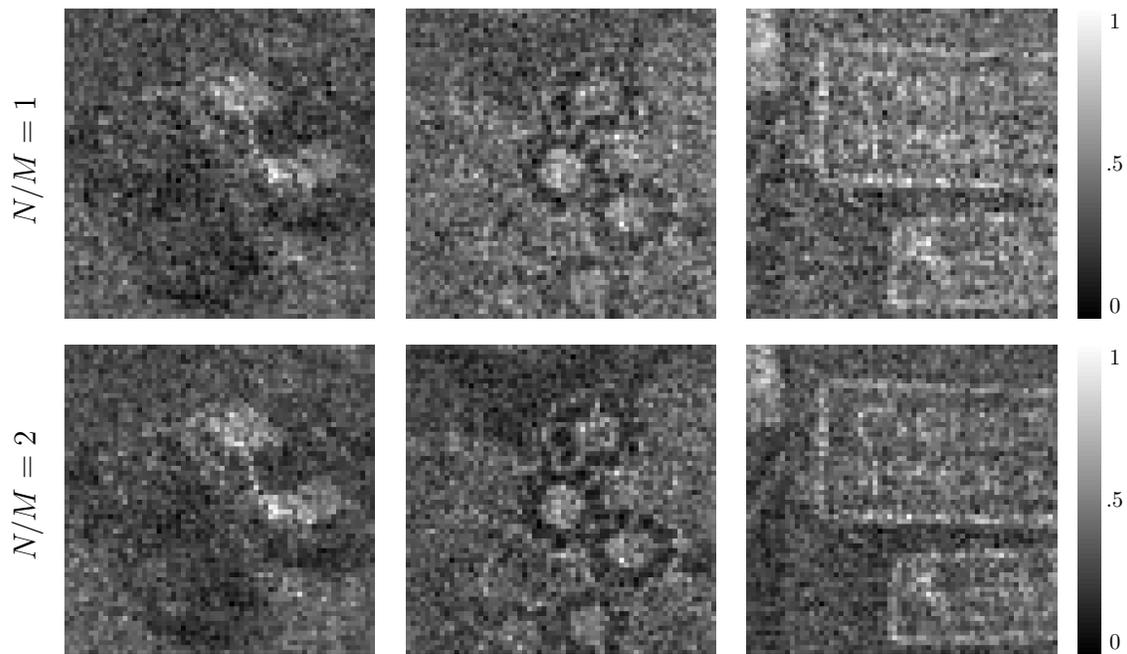


Figure 6-3: Presented are two sets of LMMSE estimates of the example scenes from Fig. 4-1. For both sets the bucket detector is three times the size of the transmitter ($\beta = 3$). In the top set we have a sampling ratio of 1 (as many measurements as pixels), while the bottom set uses twice as many measurements.

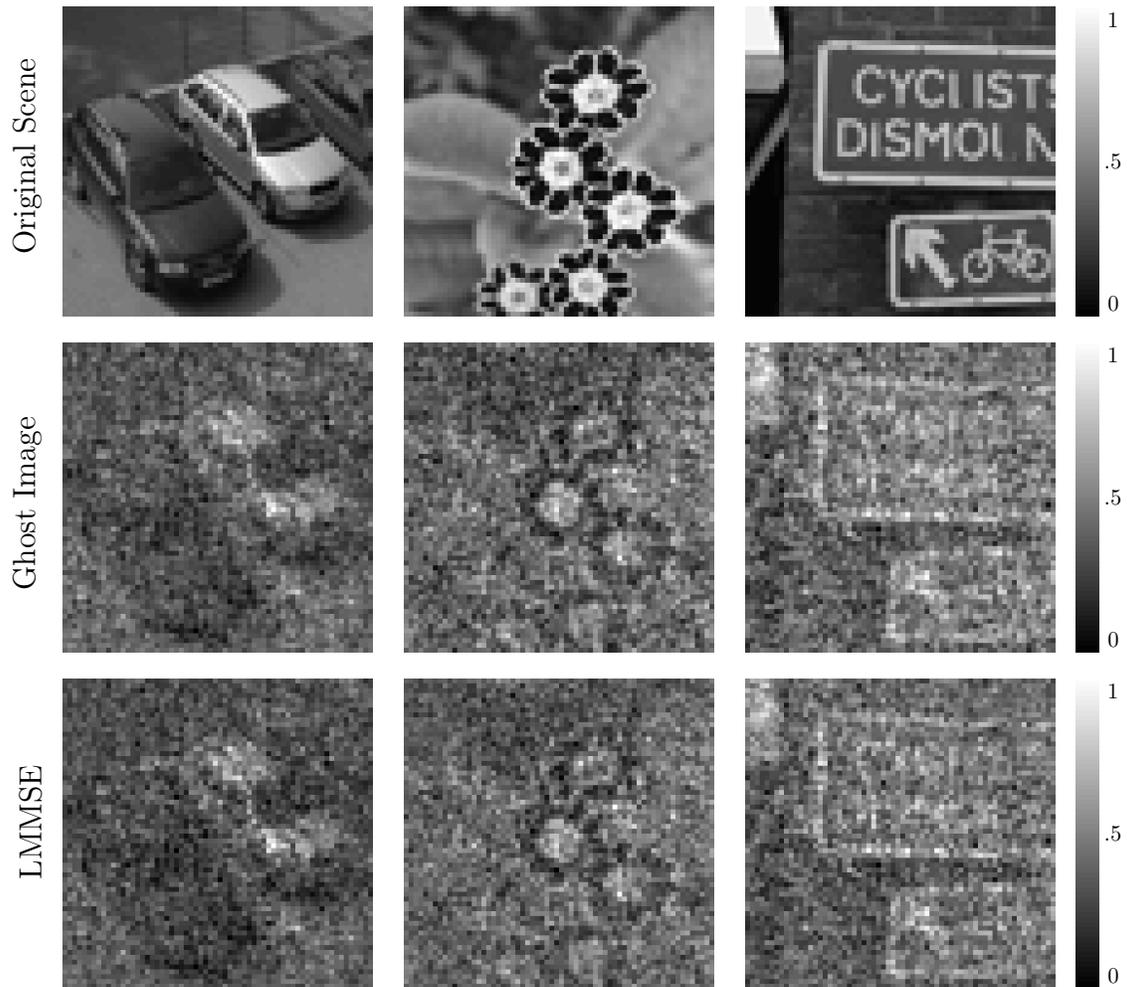


Figure 6-4: A direct comparison of the ghost image and LMMSE reconstructions. The original scene is also presented in the top row. Both image estimates were obtained with a sampling ratio of 1, and for a detector that is three times the size of the source, $\beta = 3$.

Chapter 7

Approximate Inference for Optimal Estimation

We began this thesis in Ch. 2 by laying out the physics of structured-illumination standoff imaging. Then, in Ch. 3, we used that framework to analyze ghost imaging—a particular type of structured-illumination imaging—and compared its performance to a floodlight LADAR system. While this analysis provided insight into the physics of structured-illumination imaging using pseudo-random fields, to design more advanced algorithms we needed to discretize our framework. That task was performed in Ch. 4.

Although we were then ready to start designing new inference algorithms for structured-illumination imaging, we first used Ch. 5 to derive the Cramér-Rao bound (CRB) for our framework. The CRB established the ultimate performance limit of any possible unbiased estimator, providing a goal to design against. Since the ghost image is a linear estimator, we began our investigation of new algorithms in Ch. 6 with the linear minimum mean-squared error (LMMSE) estimator, which is the estimator with the lowest MSE of all estimators that are a linear function of the measurements. Unfortunately, the LMMSE estimator’s performance was only a slight improvement on the ghost imager’s performance, and fell far short of the MSE goal set by the CRB. If we hope to achieve MSE approaching the CRB, we will need to explore nonlinear estimators.

The thesis bifurcates at this point. Whereas the first half required an under-

standing of optical physics and the basics of estimation theory, the second half builds on modern principles of probabilistic inference, primarily graphical modeling, belief propagation, and the Expectation-Maximization (E-M) algorithm. Unfortunately there are some notational clashes between the two halves, as some symbols have different accepted uses in the two different fields. The glossary of terms at the end of the thesis is the ultimate reference on notation, but we pause here for a limited summary of some important notation that will be used in what follows.

When describing statistical quantities, we adhere to the following principles. First, an overbar will denote the mean of a variable computed from its prior distribution, i.e., $\bar{x} = \langle \mathbf{x} \rangle$ is the mean of \mathbf{x} . A caret, however will denote an estimate of the variable. It is often used for a posterior mean, i.e., $\hat{x} = \langle \mathbf{x} | y \rangle$ is the posterior mean of \mathbf{x} given observation y , but it can denote other estimates¹. Finally, ν is used primarily to represent a variance of a random variable, i.e., $\nu^x = \mathbf{var}[\mathbf{x}]$. However, ν is also used to represent the uncertainty of an estimate. In this case the exact meaning will be context dependent, but it will always represent some type of uncertainty measure. Whenever a logarithm is used, it is in base e unless otherwise specified, i.e., $\log x \equiv \log_e x \equiv \ln x$. Finally, θ will be used exclusively for unknown parameters.

In the remainder of this chapter we introduce the fundamentals of estimation theory, followed by an overview of graphical modeling and loopy belief propagation. These are general concepts that have been used in a variety of fields to efficiently compute optimal estimates. This chapter does not focus on original work, but rather is provided to help the reader follow the work appearing in subsequent chapters, including original contributions made in this thesis. In the first half of Ch. 8 we present the generalized approximate message-passing (GAMP) algorithm [17], which approximates the belief propagation presented in this chapter (under certain circumstances). However, GAMP is defined for an ideal framework and cannot be immediately applied to our imaging scenario. Therefore, in the second half of Ch. 8, we modify and extend the existing GAMP algorithm to make it applicable to the physics of our imaging scenario. We then present our modified algorithm and preliminary results

¹The other primary usage will be for a MAP estimate.

when employing a simplistic independent, identically distributed (iid) Gaussian prior for each pixel in the target.

The MSE of the estimate produced by this modified version of GAMP is only marginally better than LMMSE estimator's MSE, but this is purely a consequence of our choice of target prior. We start seeing real performance gains in Ch. 9, when we introduce a prior based on scenes having a sparse representation in a wavelet basis. We also apply a standard extension of GAMP, in which unknown system parameters can be estimated during the running of GAMP. Combined, these two changes yield significantly improved images in comparison to all previous methods. We then go one step further and use our knowledge of imaging physics to jointly estimate the target and the speckle, resulting in even better reconstructions, both quantitatively (MSE) and qualitatively (subjective viewing of the aforementioned reconstructions).

However, before proceeding with algorithm development, we need to establish exactly what it means for one estimation scheme to be better than the next. We therefore present a brief discussion of optimal estimation, after which we give a high-level overview of graphical modeling, leading to the loopy belief propagation (LBP) algorithm. We develop LBP in detail, presenting three different versions of the algorithm, so that the reader might understand the approximations made in Ch. 8, when we review the derivation of the GAMP algorithm.

7.1 Optimal Estimation

When discussing optimal estimation one invariably needs to choose a criterion of optimality (some way of defining what is best). While technically anything could be chosen for the criterion, most do not have benefit for the user. There are two major schools of thought when it comes to estimation: Bayesian and frequentist. A frequentist assumes no prior distribution on the quantity to be estimated, treating it as deterministic but unknown. The Cramér-Rao bound derived in Ch. 5 was for this situation. Moreover, when the bound can be achieved, it is achieved by the *Maximum*

Likelihood (ML) estimator

$$\hat{\mathbf{x}}^{(\text{ML})} = \underset{\mathbf{x}}{\operatorname{argmax}} p_{\mathbf{y}}(\mathbf{y}; \mathbf{x}), \quad (7.1)$$

where the variable \mathbf{x} is being estimated from the measurement \mathbf{y} .

In many situations, however, we have some prior knowledge about what we are trying to estimate, and the Bayesian approach exploits this knowledge. Bayesian estimation is often phrased as minimizing a cost function $C(\mathbf{x}, \mathbf{y})$ that is a function of the joint distribution of \mathbf{x} and \mathbf{y} . There are many ways of choosing the cost function, but it is often selected to be a statistical measure of the expected error or likelihood of the measurements. Similar to the ML estimate, we define the *Maximum a Posteriori Probability* (MAP) estimator

$$\hat{\mathbf{x}}^{(\text{MAP})} = \underset{\mathbf{x}}{\operatorname{argmax}} p_{\mathbf{x}}(\mathbf{x}|\mathbf{y}) \quad (7.2)$$

as the most likely vector in the posterior distribution². The other standard Bayesian estimator is to define the cost function as the expected value of the p -norm of the error,

$$\hat{\mathbf{x}}^{(p\text{-NORM})} = \underset{\mathbf{f}}{\operatorname{argmin}} \langle \|\mathbf{x} - \mathbf{f}(\mathbf{y})\|_p \rangle = \underset{\mathbf{f}}{\operatorname{argmin}} \left\langle \left(\sum_m |x_m - f_m(\mathbf{y})|^p \right)^{\frac{1}{p}} \right\rangle. \quad (7.3)$$

The predominate p -norm cost function uses $p = 2$, leading to the minimum mean-squared error (MMSE) estimator,

$$\hat{\mathbf{x}}^{(\text{MMSE})} = \underset{\mathbf{f}}{\operatorname{argmin}} \langle \|\mathbf{x} - \mathbf{f}(\mathbf{y})\|_2 \rangle. \quad (7.4)$$

Solving the minimization problem, the MMSE estimate reduces to the posterior expectation,

$$\hat{\mathbf{x}}^{(\text{MMSE})} = \langle \mathbf{x} | \mathbf{y} \rangle. \quad (7.5)$$

²A posterior distribution is one formed after making observation.

The expectation is achieved by finding the posterior distribution $p_x(x_m|\mathbf{y})$ for each component of the signal vector.

Computational Complexity

The ML, MAP, and MMSE estimators have been defined for vector measurements. For the purposes of discussion we assume there are M components to the vector \mathbf{x} we wish to estimate, and N to the measurement \mathbf{y} . We will also assume that x_m is a discrete random variable that can take on P possible values. The ML and MAP estimates are then searching for the optimal vector over a space of size P^M . Because this space grows exponentially with the number of components in \mathbf{x} , directly solving this problem quickly becomes impossible if there is no additional information to guide the search. However, if the objective function to be optimized over is convex—or better yet linear—in terms of \mathbf{x} , then there are computationally feasible solutions that only grow polynomially with the number of variables.

A similar problem occurs when solving for the MMSE estimator. We need to find each marginal posterior distribution $p_{x|\mathbf{y}}(x_m|\mathbf{y})$, which requires marginalizing each of the other variables. This thus occurs M times, and in the general case each marginalization takes $O(P^M)$ calculations.

Because the preceding computations grow quickly in complexity with the number of variables, and images can have thousands to millions of pixels, a direct computation of an optimal estimate is not feasible. Our exploration of the LMMSE estimator in Ch. 6 was driven by the desire to avoid undue computational burden, but it offered minimal improvement over conventional ghost imaging. However, there is a rich field of approximate estimation techniques that use iterative methods to search for the optimal estimate, or at least locally-optimal estimates. For inference over a high-dimensional probability distribution, one of most widely used algorithms is LBP, which we build upon in this thesis. As a prelude to LBP, we now provide a short tutorial on graphical models and message-passing algorithms, two useful tools that we will rely on heavily in the rest of the thesis.

7.2 Graphical Modeling

For our purposes, graphical modeling can be thought of as the mapping of a probability distribution to a set of nodes and the edges that connect those nodes. Graphical modeling is a large field and it has many different formalizations. Chief among these are directed graphs, undirected graphs, and factor graphs [44].

In this thesis, we are primarily concerned with factor graphs. These are formed by finding a suitable factorization of the joint distribution of the random variables $\mathbf{x} = [x_1, \dots, x_M]^T$ we are estimating,

$$p_{\mathbf{x}}(\mathbf{x}) = \prod_i f_i(\mathbf{x}_{\mathcal{F}_i}), \quad (7.6)$$

where \mathcal{F}_i is a set of indices that indicates the components of \mathbf{x} that are contained in factor $f_i(\cdot)$, making $\mathbf{x}_{\mathcal{F}_i}$ a sub-vector of \mathbf{x} . If there are any measurements being used to inform the inference, they are implicit in this formalization, i.e., $p_{\mathbf{x}}(\mathbf{x})$ would be the posterior distribution $p_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y})$.

From this factorization we form the graph by assigning a variable node for each variable x_j in the joint distribution, and a factor node for each factor f_i . As seen in Fig. 7-1 these nodes form a bipartite³ graph, with each factor node f_i connected to its neighboring variable nodes $\{x_j : j \in \mathcal{F}_i\}$. Likewise, for each variable node there is a set of neighboring factor nodes $\{f_i : i \in \mathcal{X}_j\}$, where \mathcal{X}_j defines the indices of the factors that depend on x_j . Although each factor node can connect to multiple variable nodes, and each variable node can connect to multiple factor nodes, there are no direct connections among the factor nodes, nor are there direct connections among the variable nodes.

Having abstracted our probability distribution to a graph, we can now design algorithms that run on that graph. These fall into a large class of message-passing algorithms in which computations are performed at each node, and then messages are sent along the edges to adjacent nodes. That is, for iteration t , we compute a

³The nodes in a bipartite graph form two disjoint sets. In each set, there are no intra-connections among the nodes. However, connections are allowed if the two nodes are in different sets.

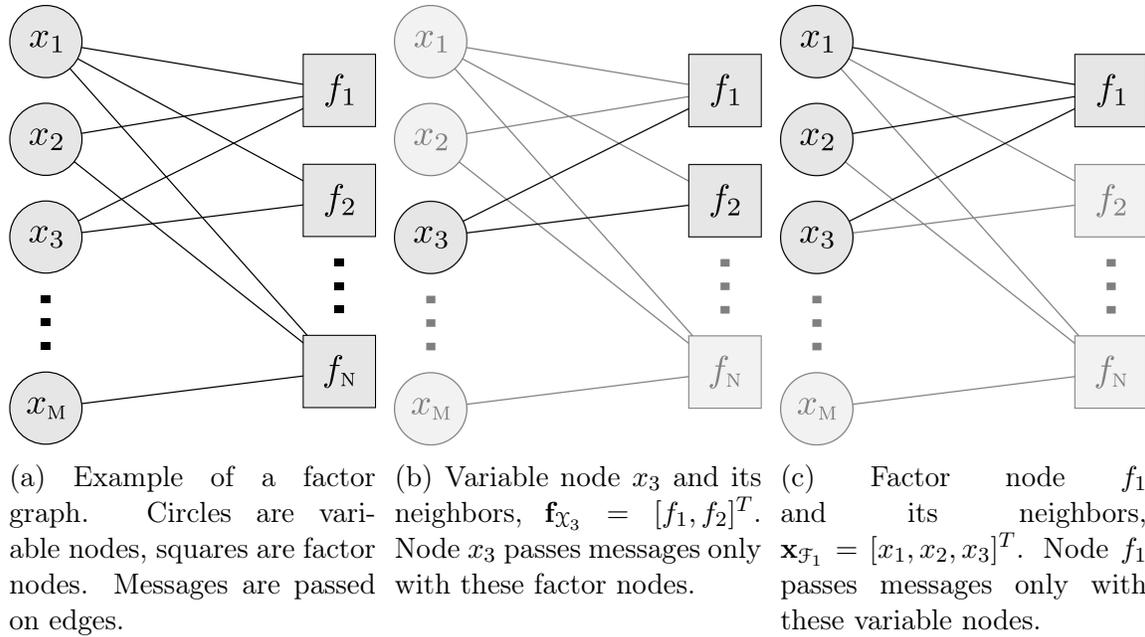


Figure 7-1: An example of a factor graph. The vector \mathbf{x} has M elements, and the distribution factors into N potential factors.

set of messages $\{\Delta_{j \rightarrow i}^t(x_j)\}$ at each variable node j that is passed to each connected factor node $i \in \mathcal{X}_j$. Then, computations are performed on these messages at each factor node i , which forms the new messages $\{\Delta_{j \leftarrow i}^t(x_j)\}$ that are sent back to each connected variable node $j \in \mathcal{F}_i$. This procedure repeats until the algorithm converges. The form of these messages, as well as the schedule at which they are sent, can be varied to form different algorithms that serve different purposes. We are concerned with the sum-product and max-sum algorithms, which are collectively known as *belief propagation* [45]; sum-product computes the MMSE estimate of a random vector, while max-sum computes its MAP estimate.

7.3 Loopy Belief Propagation

The sum-product and max-sum algorithms were originally for tree-like graphs, in which there are no loops (there is only one path between each pair of nodes). In such cases the algorithms are guaranteed to converge in a deterministic number steps. An important example of such graphs are Hidden Markov Models (HMM), and belief

propagation on an HMM is equivalent to the forward-backward algorithm [41].

Belief propagation was later extended to graphs with cycles [45], and although there are no longer rigorous proofs of convergence as there are for loop-free graphs, it has been shown that many useful algorithms can be reduced to LBP [44]. LBP has also been applied to estimation on jointly Gaussian distributions. Here, the messages reduce to updating means and covariance matrices, and the algorithm has been shown to converge in most cases [46]. It has also been shown that Gaussian belief propagation on HMMs is equivalent to the classic Kalman filtering algorithm [41]. LBP, in general, is an approximate inference technique for loopy graphs.

We herein present versions of the sum-product and max-sum algorithms with parallel message scheduling that is appropriate for LBP. In both of these cases, when computing a message to pass along edge (i, j) , we do not use the incoming message. That is, $\Delta_{j \rightarrow i}^{t+1}(x_j)$, the message that variable node x_j passes to factor node f_i at time $t + 1$, does not involve $\Delta_{j \leftarrow i}^t(x_j)$, the message that factor node f_i passed to variable node x_j at time t . Likewise $\Delta_{j \leftarrow i}^{t+1}(x_j)$ does not depend on $\Delta_{j \rightarrow i}^{t+1}(x_j)$. This message-passing structure is shown in Fig. 7-2. For LBP with discrete random variables the messages are vectors that are proportional to probability mass functions for the current beliefs about the variable. For LBP with continuous random variables the messages contain sufficient statistics to characterize the probability density functions of these beliefs.

7.3.1 Sum-Product Algorithm

The sum-product algorithm implements MMSE LBP. The basic algorithm is presented in Alg. 1 for discrete distributions⁴. In this algorithm we see the historical origin of the name: the outgoing messages from the factor nodes are sums over all but one of the incoming messages, while the outgoing messages from the variables nodes are the products of all but one of their incoming messages. However our focus will be on continuous variables, so the sum becomes an integration. Moreover, to simplify

⁴The messages are proportional to a probability mass function in a discrete formalization. If x_j can take P values, $\Delta_{j \rightarrow i}^t(x_j)$ and $\Delta_{j \leftarrow i}^t(x_j)$ are length P vectors.

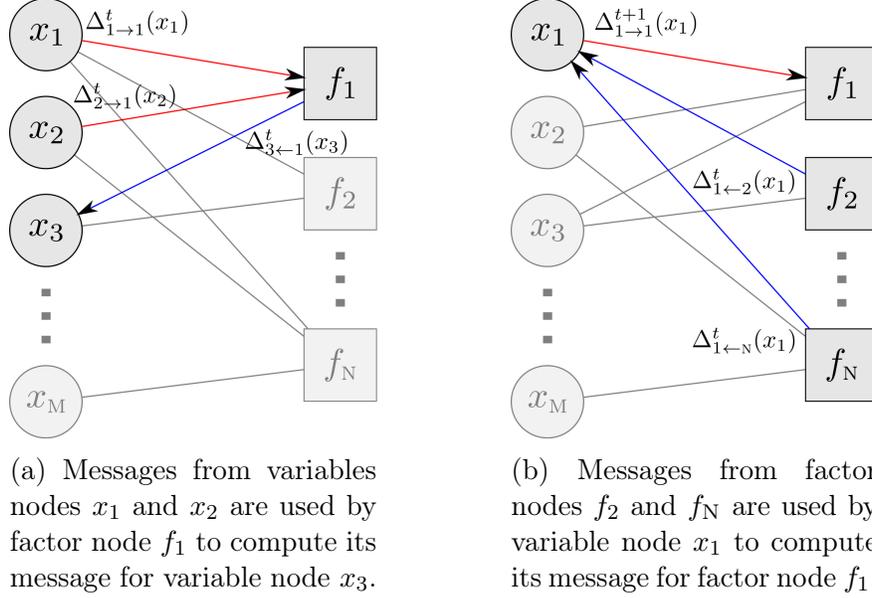


Figure 7-2: An example of messages being passed and computed on a graph for belief propagation.

the later connection to GAMP, we change our messages to be the logarithms of the messages in Alg. 1. This form of the algorithm, which we approximate later in GAMP, is shown in Alg. 2.

Algorithm 1 Sum-Product algorithm for discrete distributions

- 1: **Initialize:** $\Delta_{j \leftarrow i}^0(x_j) = 1$, for $t = 0$, and all connected nodes (j, i) .
 - 2: **repeat**
 - 3:
$$\Delta_{j \rightarrow i}^{t+1}(x_j) = \prod_{k \in \mathcal{X}_j \setminus i} \Delta_{j \leftarrow k}^t(x_j)$$
 - 4:
$$\Delta_{j \leftarrow i}^{t+1}(x_j) = \sum_{\mathbf{x}_{\mathcal{F}_i \setminus j}} f_i(\mathbf{x}_{\mathcal{F}_i}) \prod_{\ell \in \mathcal{F}_i \setminus j} \Delta_{\ell \rightarrow i}^{t+1}(x_\ell)$$
 - 5: **until** convergence condition is satisfied
- return** $p_x^{(t)}(x_j) \propto \prod_{i \in \mathcal{X}_j} \Delta_{j \leftarrow i}^t(x_j)$ ▷ Compute expectation from $p_x^{(t)}(x_j)$
-

The main differences between Alg. 1 and Alg. 2 are as follows: we replaced the summations in Alg. 1 with integrations; the new messages in Alg. 2 are the logarithms of the Alg. 1 messages; and the Alg. 2 factor potentials are the logarithms of the Alg. 1 potentials, i.e., we now have $\log p_{\mathbf{x}}(\mathbf{x}) = \sum_i f_i(\mathbf{x}_{\mathcal{F}_i})$.

The sum-product algorithm can be thought of as repeating two steps. First, at each variable node we compute a message that is essentially the current estimate

Algorithm 2 Sum-Product algorithm for continuous distributions, log-messages

- 1: **Initialize:** $\Delta_{j \leftarrow i}^0(x_j) = 0$, for $t = 0$, and all connected nodes (j, i) .
 - 2: **repeat**
 - 3: $\Delta_{j \rightarrow i}^{t+1}(x_j) = \sum_{k \in \mathcal{X}_j \setminus i} \Delta_{j \leftarrow k}^t(x_j)$
 - 4: $\Delta_{j \leftarrow i}^{t+1}(x_j) = \log \int d\mathbf{x}_{\mathcal{F}_i \setminus j} \exp \left(f_i(\mathbf{x}_{\mathcal{F}_i}) + \sum_{\ell \in \mathcal{F}_i \setminus j} \Delta_{\ell \rightarrow i}^{t+1}(x_\ell) \right)$
 - 5: **until** convergence condition is satisfied
- return** $p_{\mathbf{x}}^{(t)}(x_j) \propto \exp \left(\sum_{i \in \mathcal{X}_j} \Delta_{j \leftarrow i}^t(x_j) \right)$ \triangleright Compute expectation from $p_{\mathbf{x}}^{(t)}(x_j)$
-

(also called the *belief*) of the distribution of that variable. Then, at each factor node we compute the likelihood of that factor given x_j , where the other variables are marginalized using their current messages. This process repeats, as the factor nodes send their likelihoods back to each variable node, which use them to update their beliefs. At the end, this produces marginal posterior distributions, from which we can calculate the marginal posterior expectations, which are the components of the MMSE estimate.

Exact cases for convergence of the sum-product algorithm on loopy graphs are difficult to ascertain, but in some sense we need the combinations of the likelihoods (calculated at the factors) to be a reasonable approximation for the joint likelihood (which is too difficult to directly calculate).

7.3.2 Max-Sum Algorithm

The max-sum algorithm computes the MAP estimate of the random vector \mathbf{x} . Much like the sum-product algorithm, it iterates between computations at the variable and factor nodes. However, instead of computing likelihoods and new distributions, this algorithm computes local maxima. The max-sum algorithm is detailed in Alg. 3. Like Alg. 2, it uses log-based messages, but it is valid for both discrete and continuous distributions.

The max-sum algorithm can be thought of as implementing the following two

Algorithm 3 Max-Sum algorithm for continuous distributions, log-messages

- 1: **Initialize:** $\Delta_{j \leftarrow i}^0(x_j) = 0$, for $t = 0$, and all connected nodes (j, i) .
 - 2: **repeat**
 - 3: $\Delta_{j \rightarrow i}^{t+1}(x_j) = \sum_{k \in \mathcal{X}_j \setminus i} \Delta_{j \leftarrow k}^t(x_j)$
 - 4: $\Delta_{j \leftarrow i}^{t+1}(x_j) = \max_{\mathbf{x}_{\mathcal{F}_i \setminus j}} \left[f_i(\mathbf{x}_{\mathcal{F}_i}) + \sum_{\ell \in \mathcal{F}_i \setminus j} \Delta_{\ell \rightarrow i}^{t+1}(x_\ell) \right]$
 - 5: **until** convergence condition is satisfied
- return** $p_{\mathbf{x}}^{(t)}(x_j) \propto \exp \left(\sum_{i \in \mathcal{X}_j} \Delta_{j \leftarrow i}^t(x_j) \right)$ ▷ Compute maximum from $\{p_{\mathbf{x}}^{(t)}(x_j)\}$, backtracking if needed
-

steps. First, at the variable nodes we combine the incoming messages to form a belief for x_j . Unlike the sum-product algorithm's belief, this is not a posterior distribution, but rather a distribution of the estimate of the most likely value of x_j . At the factor node, the combination of the potential function and incoming messages are maximized, as a function of x_j . These marginal distributions are then sent back to the variable nodes. In general, the MAP estimate is found from the final marginal distributions at each variable node by backtracking through the graph. However, for the situations we are concerned with (namely the GAMP algorithm), the MAP estimate can be computed as the component-wise maximum of each of the $\{p_{\mathbf{x}}^{(t)}(x_j)\}$.

Chapter 8

Adapting GAMP for Standoff Optical Imaging

In Ch. 7 we developed our concept of optimal estimation, and explored approximating these estimates with loopy belief propagation (LBP). Unfortunately, for the distributions arising from our structured-illumination scenario, in which every measurement is dependent on every component of the signal vector, LBP is not computationally efficient. However, as we will show later in this chapter, our imaging problem can be framed as a linear transformation of the target, with component-wise noise on the resulting vector. That is, for a target \mathbf{x} and a measurements matrix \mathbf{A} ,

$$\mathbf{z} = \mathbf{A}\mathbf{x}, \quad y_i \sim p_{y|z}(y_i|z_i), \quad (8.1)$$

so that: (1) each of the output components $\{z_i\}$ is a linear combination of the target components $\{x_j\}$; and (2) each measurement $y_i = y_i$ is generated from the conditional probability distribution $p_{y|z}(y_i|z_i)$ that is only dependent on the scalar random variable z_i .

Optical propagation and reflection are linear processes, so the field at the detector is the product of the on-target field (a Gaussian random process with coherence length ρ_L) and the target, projected through a Fourier matrix. From the discretized framework we developed in Ch. 4, the field at the detector is well approximated as

a discrete linear transformation of the discretized version of the target. If a heterodyne detector is used, then the component of this field that is aligned with the local oscillator’s spatial mode is extracted by the measurement, albeit embedded in additive Gaussian noise. The heterodyne measurement thus fits the desired framework in Eq. (8.1), i.e., a discrete linear transformation of the target, followed by noise.

Starting with [38–40] the preceding framework has been the basis for the field of *compressed sensing*, in which ℓ_1 -norm minimization schemes allow for undersampling—making fewer measurements than the number of target pixels—when the signal is known to be sparse. The original solutions to this problem were based on convex optimization, but then, in [16], Donoho et al. developed the *Approximate Message-Passing* (AMP) algorithm. The AMP algorithm could approximate loopy belief propagation under certain conditions, and if a sparsifying prior is chosen (a heavy tail distribution, e.g., exponential, Laplacian), the algorithm performs efficient compressed sensing. In [17] Rangan extended Donoho et al.’s work to an arbitrary independent, identically distributed (iid) prior on the target and an arbitrary iid measurement distribution with what is known as the *Generalized Approximate Message-Passing* (GAMP) algorithm.

Although GAMP is one of the most efficient algorithms for compressed sensing, just as importantly for us it is based on graphical modeling and can therefore be embedded in a larger graph to aid inference on an extended system [47]. Because our physical system is far from the ideal described in these algorithms, this flexibility will be key to adapting GAMP to our standoff-sensing application. The rest of the chapter is broken down into three main parts. First, we develop the GAMP algorithm, following the derivation in [17], so that the reader can get an idea of the approximations used, and their limitations. We then modify GAMP to match it to our model of the physical imaging systems from Ch. 2 and Ch. 4. Finally, we look at the more challenging non-idealities of our imaging framework, and develop ways to handle them.

8.1 Developing GAMP

For a full, general development of the GAMP algorithm, see [17]. In this section we work through its development when approximating LBP for MMSE estimation; i.e., we approximate the sum-product algorithm shown in Alg. 2. This is the primary case we will use later in the thesis, and so its derivation is presented to provide insight to the reader that is relevant to the later extensions of GAMP.

The signal we are estimating is the random vector \mathbf{x} , whose elements we assume are real-valued iid random variables with distribution $p_{\mathbf{x}}(x_j; \theta)$ that may depend on a parameter θ . The measurements \mathbf{y} (also real-valued) are generated from a component-wise conditional distribution $p_{\mathbf{y}|\mathbf{z}}(y_i|z_i)$, where $\mathbf{z} = \mathbf{A}\mathbf{x}$. The matrix \mathbf{A} is assumed to be composed of known realizations of zero-mean, real-valued iid random variables. Under these conditions the joint distribution factors as,

$$p_{\mathbf{x},\mathbf{y}}(\mathbf{x}, \mathbf{y}) = \prod_i p_{\mathbf{y}|\mathbf{x}}(y_i|\mathbf{x}) \prod_j p_{\mathbf{x}}(x_j; \theta), \quad (8.2)$$

and we immediately see that the factor potentials in the GAMP factor graphs, shown in Fig. 8-1, arise from the prior distributions $p_{\mathbf{x}}(x_j; \theta)$ and the conditional measurement distributions $p_{\mathbf{y}|\mathbf{x}}(y_i|\mathbf{x})$.

The factor potentials thus split into input potentials $\{f_{\text{in}}^{(j)}\}$, and output potentials $\{f_{\text{out}}^{(i)}\}$. Using the log-potentials formalization from Alg. 2, these are

$$f_{\text{in}}^{(j)}(x_j, \theta) = \log p_{\mathbf{x}}(x_j; \theta) \quad (8.3)$$

$$f_{\text{out}}^{(i)}(y_i, z_i) = \log p_{\mathbf{y}|\mathbf{z}}(y_i|z_i), \quad (8.4)$$

where, since $\mathbf{z} = \mathbf{A}\mathbf{x}$, we have substituted the scalar potential function $p_{\mathbf{y}|\mathbf{z}}(y_i|z_i) \equiv p_{\mathbf{y}|\mathbf{x}}(y_i|\mathbf{x})$, making the dependence of \mathbf{z}_i on \mathbf{x} implicit. We note that because the $\{x_j\}$ are drawn iid, when the argument x_j is provided the superscript on the potential is superfluous and thus dropped, i.e., $f_{\text{in}}(x_j, \theta) \equiv f_{\text{in}}^{(j)}(x_j, \theta)$. Likewise, the measurements are also generated by identical processes¹, so when the arguments are present

¹This is a requirement for GAMP, but also true for our filtered photocurrent measurements

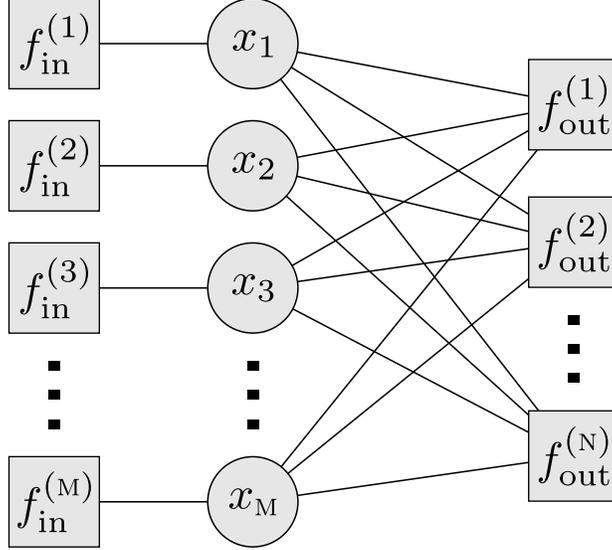


Figure 8-1: The factor graph for GAMP. There are M variables in the vector \mathbf{x} , each with an independent prior defined by the potential $f_{\text{in}}^{(j)}(x_j, \theta) = \log p_{\mathbf{x}}(x_j; \theta)$. The N measurements $\{y_i\}$ are encapsulated in the factor potentials $f_{\text{out}}^{(i)}(y_i, z_i) = \log p_{\mathbf{y}|\mathbf{z}}(y_i|z_i)$.

we can drop the potentials' superscripts and use $f_{\text{out}}(y_i, z_i) \equiv f_{\text{out}}^{(i)}(y_i, z_i)$. The graph for this factorization is shown in Fig. 8-1 where, because each $\{z_i\}$ depends on all of the $\{x_j\}$, each variable node is connected to all of the output factor nodes.

In LBP for MMSE estimation, as presented in Alg. 2, at time t we generate a set of messages $\{\Delta_{j \rightarrow i}^t(x_j)\}$ at each variable node j , for each factor node i . The factor nodes collect the incoming messages, then calculate the return messages for each variable node, $\Delta_{j \leftarrow i}^t(x_j)$. Each variable node is connected to a single input factor node (for its prior distribution), but the variable nodes are fully connected with the output factor nodes $\{f_{\text{out}}\}$. For M variable nodes and N output factor nodes, LBP requires the computation of M messages at each factor node, and each computation requires marginalizing over the messages from the other $M - 1$ variables. Each iteration thus requires $O(NM^2)$ computations at the factor nodes. The combination and distribution of the messages at the variable node takes $O(N^2M)$ computations. The beauty of GAMP is that it reduces this process to a single set of computations at each variable and factor node, and their transformations through a size $N \times M$ matrix. This

described in Ch. 2.

simplification is enabled by Central Limit Theorem (CLT) approximations, combined with Taylor expansions of the messages, as explained below.

8.1.1 Approximations

Before going through the derivation of GAMP, we address the three main approximation that we will use. The first is a CLT approximation, in which \mathbf{z} is assumed to be composed of Gaussian random variables by virtue of their construction as the weighted sums of many iid random variables. This lets us simplify the computations at each factor node to a scalar computation over z_i , whereas normally it is over the entire \mathbf{x} vector. Rewriting the factor-node computation in line 4 of the sum-product algorithm in Alg. 2 with the GAMP notation, we have

$$\Delta_{j \leftarrow i}^t(x_j) = \log \int d\mathbf{x}_{\setminus j} \exp \left(f_{\text{out}}(y_i, \mathbf{x}) + \sum_{\ell \neq j} \Delta_{\ell \rightarrow i}^t(x_\ell) \right), \quad (8.5)$$

where the notation $\mathbf{x}_{\setminus j}$ indicates the vector \mathbf{x} with element j removed. From the CLT approximation this becomes

$$\Delta_{j \leftarrow i}^t(x_j) \approx \log \int dz_i p_{\mathbf{z}|\mathbf{x}}^{(t)}(z_i|x_j) e^{f_{\text{out}}(y_i, z_i)}, \quad (8.6)$$

where $p_{\mathbf{z}|\mathbf{x}}^{(t)}(z_i|x_j)$ is a Gaussian distribution whose mean and variance are determined by the means and variances of the $\{x_j\}$ for iteration t through the $\mathbf{z} = \mathbf{A}\mathbf{x}$ transformation. Therefore, instead of passing information about an entire distribution, the message from variable node x_j only needs to pass the mean and variance of its current belief, i.e., the messages simplify to $\Delta_{j \rightarrow i}^t(x_j) \rightarrow (\hat{x}_{j \rightarrow i}(t), \nu_{j \rightarrow i}^x(t))$.

The two remaining approximations come from Taylor expansions. Instead of computing an estimate $\hat{x}_{j \rightarrow i}(t)$ for the message from variable node j to each factor node i , we compute a single expectation $\hat{x}_j(t)$ at node j . We then approximate each estimate $\hat{x}_{j \rightarrow i}(t)$ with a first-order Taylor expansion, which is shown later in the derivation in Eq. (8.29). Similarly, to allow for a single computation at each factor node we make a two-term Taylor expansion of the return messages $\Delta_{j \leftarrow i}^t(x_j)$, which is presented in

Eq. (8.15). The specifics of these approximation, their justification, and the resulting GAMP algorithm are presented in the rest of this section.

8.1.2 Factor-Node Computations

Combine Incoming Messages

We start by using our CLT approximation to specify our belief of $z_i = \sum_j A_{ij}x_j$ at time step t as the Gaussian distribution

$$p_z^{(t)}(z_i) = \frac{e^{-(z_i - \hat{p}_i(t))^2 / 2\nu_i^p(t)}}{\sqrt{2\pi\nu_i^p(t)}}, \quad (8.7)$$

where the mean and variance,

$$\hat{p}_i(t) = \sum_j A_{ij}\hat{x}_{j \rightarrow i}(t) \quad (8.8)$$

$$\nu_i^p(t) = \sum_j A_{ij}^2\nu_{j \rightarrow i}^x(t), \quad (8.9)$$

are functions of the means and variances of the incoming messages, $\{\hat{x}_{j \rightarrow i}(t), \nu_{j \rightarrow i}^x(t)\}$. However, each factor-node computation uses the distribution of z_i *conditioned* on x_j , because the incoming message from x_j is excluded from the calculation of the returning message $\Delta_{j \leftarrow i}^t(x_j)$. That conditional distribution is the Gaussian

$$p_{z|x}^{(t)}(z_i|x_j) = \frac{\exp\left(-\frac{(z_i - (\hat{p}_{j \rightarrow i}(t) + A_{ij}x_j))^2}{2\nu_{j \rightarrow i}^p(t)}\right)}{\sqrt{2\pi\nu_{j \rightarrow i}^p(t)}}, \quad (8.10)$$

where

$$\hat{p}_{j \rightarrow i}(t) = \hat{p}_i(t) - A_{ij}\hat{x}_{j \rightarrow i}(t) \quad (8.11)$$

$$\nu_{j \rightarrow i}^p(t) = \nu_i^p(t) - A_{ij}^2\nu_{j \rightarrow i}^x(t). \quad (8.12)$$

Following [17] we ignore terms of $O(A_{ij}^2)$ and greater², and then re-parameterize

²This approximation is generally used to simplify variance calculations, e.g., $\nu_{j \rightarrow i}^p(t) \approx \nu_i^p(t)$.

our return message from Eq. (8.6) as a function of \widehat{p}_i , viz.,

$$\Delta_{j \leftarrow i}^t(x_j) \approx \Delta(\widehat{p}_i(t) + A_{ij}(x_j - \widehat{x}_j(t)), \nu_i^p(t), y_i). \quad (8.13)$$

Expanding the exponent of $p_{z|x}^{(t)}(z_i|x_j)$ from Eq. (8.10) and simplifying, $\Delta(\cdot)$ can be written generally as

$$\Delta(\widehat{p}, \nu^p, y) = \log \frac{\int dz \exp\left(-\frac{z^2}{2\nu^p} + \frac{z\widehat{p}}{\nu^p} + f_{\text{out}}(y, z)\right)}{\int dz \exp\left(-\frac{z^2}{2\nu^p} + \frac{z\widehat{p}}{\nu^p}\right)}. \quad (8.14)$$

Approximate Return Message

The return message, $\Delta_{j \leftarrow i}^t(x_j)$, is still difficult to calculate, but we can approximate it with a two-term Taylor expansion. It is natural to take the expansion around the last estimate, i.e. $x_j = \widehat{x}_j(t)$, which is equivalent to expanding $\Delta(\widehat{p}_i(t) + A_{ij}(x_j - \widehat{x}_j(t)), \nu_i^p(t), y_i)$ around its first argument equaling $\widehat{p}_i(t)$. This approach, taken in [17], then yields

$$\begin{aligned} \Delta(\widehat{p}_i(t) + A_{ij}(x_j - \widehat{x}_j(t)), \nu_i^p(t), y_i) &\approx \Delta(\widehat{p}_i(t), \nu_i^p(t), y_i) \\ &+ \Delta'(\widehat{p}_i(t), \nu_i^p(t), y_i) A_{ij}(x_j - \widehat{x}_j(t)) + \frac{1}{2} \Delta''(\widehat{p}_i(t), \nu_i^p(t), y_i) A_{ij}^2(x_j - \widehat{x}_j(t))^2, \end{aligned} \quad (8.15)$$

where $\Delta'(\widehat{p}_i(t), \nu_i^p(t), y_i)$ and $\Delta''(\widehat{p}_i(t), \nu_i^p(t), y_i)$ are the first and second partial derivatives with respect to the first argument, evaluated at $\widehat{p}_i(t)$. To simplify the computation of these derivatives, we note that $\Delta(\widehat{p}, \nu^p, y)$ is the logarithm of the ratio of the partition functions of exponential families with natural statistic \widehat{p}/ν^p . As shown in App. B.1, the first two derivatives of the log-partition function of an exponential family—with respect to its natural statistic—are the mean and variance under that distribution.

However, the Taylor expansion of $\widehat{x}_{j \rightarrow i}(t)$ calculated in Eq. (8.29) is $\widehat{x}_{j \rightarrow i}(t) \approx \widehat{x}_j(t) - \nu_j^x(t) A_{ij} \widehat{s}_i(t)$, where $\widehat{s}_i(t)$ is computed in Eq. (8.17). Therefore $\widehat{p}_{j \rightarrow i}(t) + A_{ij} x_j = \widehat{p}_i(t) + A_{ij}(x_j - \widehat{x}_j(t)) + O(A_{ij}^2)$, and we drop the $O(A_{ij}^2)$ for Eq. (8.13).

At this point we follow the approach of [17], and specify an estimation function $G_y(\cdot)$ that concisely captures the computations required at the factor node. Since we only need functions of the derivatives of the message, we define

$$G_y(\hat{p}, \nu^p, y) = \Delta'(\hat{p}, \nu^p, y). \quad (8.16)$$

To complete the our approximation of the return message we therefore only have to define two final terms based on this function,

$$\hat{s}_i(t) = G_y(\hat{p}_i(t), \nu_i^p(t), y_i) = \frac{1}{\nu_i^p(t)} [\langle z_i | \hat{p}_i(t), \nu_i^p(t), y_i \rangle - \hat{p}_i(t)] \quad (8.17)$$

$$\nu_i^s(t) = -G'_y(\hat{p}_i(t), \nu_i^p(t), y_i) = \frac{1}{\nu_i^p(t)} \left[1 - \frac{\mathbf{var}[z_i | \hat{p}_i(t), \nu_i^p(t), y_i]}{\nu_i^p(t)} \right], \quad (8.18)$$

where the new mean and variance are taken with respect to the posterior distribution

$$p_z^{(t)}(z_i | y_i) \propto p_y(y_i | z_i) p_z^{(t)}(z_i) \propto e^{f_{\text{out}}(z_i, y_i) - \frac{(z_i - \hat{p}_i)^2}{2\nu_i^p}} \quad (8.19)$$

The return message is then approximately

$$\Delta_{j \leftarrow i}^t(x_j) \approx \Delta_{j \leftarrow i}^t(\hat{x}_j) + \hat{s}_i(t) A_{ij} (x_j - \hat{x}_j(t)) - \frac{\nu_i^s(t)}{2} A_{ij}^2 (x_j - \hat{x}_j(t))^2. \quad (8.20)$$

For this approximation we only need two scalar computations $(\hat{s}_i(t), \nu_i^s(t))$ at each factor node to fully define all of the returning messages to the variable nodes.

8.1.3 Variable-Node Computations

We have reached the halfway point of the GAMP derivation. So far we have shown that: (1) the messages from the variable nodes to the factor nodes can be approximated with their means and variances, i.e., $\Delta_{j \rightarrow i}^t(x_j) \rightarrow (\hat{x}_{j \rightarrow i}(t), \nu_{j \rightarrow i}^x(t))$; (2) each factor node is a function of only linear combinations of these means and variances, $\hat{p}_i(t)$ and ν_i^p ; and (3), employing a Taylor approximation of the return messages allows for a single set of scalar computations at each factor node, i.e., Eqs. (8.17) and

(8.18). All that remains is to combine the new messages at each variable node, and use them to compute the means and variances of the variable node messages for the next iteration, $\widehat{x}_{j \rightarrow i}(t+1)$ and $\nu_{j \rightarrow i}^x(t+1)$.

Combine Incoming Messages

Each variable node computes its return message for factor node i from a sum of the incoming messages (except the one from i). In the GAMP notation, this is

$$\Delta_{j \rightarrow i}^t(x_j) = f_{\text{in}}(x_j; \theta) + \sum_{k \neq i} \Delta_{j \leftarrow k}^t(x_j). \quad (8.21)$$

The potential $f_{\text{in}}(x_j; \theta)$ does not change during the run of the algorithm, but we need to characterize the summation of messages from the factor nodes. We start by noting that, from their definition in Eq. (8.6), the message from factor node i at time t approximates the log-likelihood³ of x_j given y_i , i.e., $\Delta_{j \leftarrow i}^t(x_j) \approx \log p_{y|x}^{(t)}(y_i|x_j)$. The combination of these messages is then an approximate likelihood of \mathbf{y} , given x_j , for iteration t . From our CLT approximations this likelihood for the measurements becomes the likelihood of a scalar Gaussian random variable r_j , with mean x_j . Therefore, the combination of returning messages form the likelihood distribution

$$p_{r|x}^{(t)}(\widehat{r}_j(t)|x_j) = \frac{\exp\left(-\frac{(\widehat{r}_j(t)-x_j)^2}{2\nu_j^r(t)}\right)}{\sqrt{2\pi\nu_j^r(t)}} \propto \exp\left(\sum_i \Delta_{j \leftarrow i}^t(x_j)\right), \quad (8.22)$$

where the new effective measurement \widehat{r}_j and variance ν_j^r are found from pattern matching the summation of the messages from Eq. (8.20) with the prescribed Gaussian form, yielding

$$\nu_j^r(t) = \left[\sum_i A_{ij}^2 \nu_i^s(t) \right]^{-1} \quad (8.23)$$

$$\widehat{r}_j(t) = \widehat{x}_j + \nu_j^r(t) \sum_i A_{ij} \widehat{s}_i(t). \quad (8.24)$$

³The other variables are integrated out using the messages they sent to factor node i at time t .

Now, as specified in Eq.(8.21), the return message to factor node i does not include the incoming message from that node, so to compute $\Delta_{j \rightarrow i}^t(x_j)$ we use the summation $\sum_{k \neq i} \Delta_{j \leftarrow k}^t(x_j)$ which is the Gaussian distribution with mean $\widehat{r}_{j \leftarrow i}(t)$ and variance $\nu_{j \leftarrow i}^r(t)$. As we have done throughout the derivation, we can approximate the variance for the specific message from i to j with the general variance for node j , i.e., $\nu_{j \leftarrow i}^r(t) \approx \nu_j^r(t)$, while the estimate for this message is

$$\widehat{r}_{j \leftarrow i}(t) = \widehat{r}_j(t) - \nu_j^r(t) A_{ij} \widehat{s}_i(t). \quad (8.25)$$

The return message $\Delta_{j \rightarrow i}^{t+1}(x_j)$ in Eq. (8.21) is now the sum of the input factor potential and the exponent of a Gaussian distribution,

$$\Delta_{j \rightarrow i}^{t+1}(x_j) = f_{\text{in}}(x_j; \theta) - \frac{(\widehat{r}_{j \leftarrow i}(t) - x_j)^2}{2\nu_j^r(t)}. \quad (8.26)$$

All that is left is to compute the mean and variance of the new belief of each \mathbf{x}_j ,

$$\widehat{x}_{j \rightarrow i}(t+1) = \langle \mathbf{x}_j | \widehat{r}_{j \leftarrow i}(t), \nu_j^r(t); \theta \rangle \quad (8.27)$$

$$\nu_{j \rightarrow i}^x(t+1) = \mathbf{var}[\mathbf{x}_j | \widehat{r}_{j \leftarrow i}(t), \nu_j^r(t); \theta], \quad (8.28)$$

using the new distribution $\exp(\Delta_{j \rightarrow i}^t(x_j))$.

Return Message Approximation

As with the factor node, we only want to perform a single set of computations at each variable node, and use a Taylor expansion around this calculation to make adjustments for the specific messages going to each neighboring node. We therefore make a first-order expansion of each posterior estimate $\widehat{x}_{j \rightarrow i}(t+1)$ at $\widehat{r}_{j \leftarrow i}(t) = \widehat{r}_j(t)$, i.e.

$$\widehat{x}_{j \rightarrow i}(t+1) \approx \langle x_j | \widehat{r}_j(t), \nu_j^r(t); \theta \rangle + \left(\frac{\partial}{\partial \widehat{r}_j(t)} \langle x_j | \widehat{r}_j(t), \nu_j^r(t); \theta \rangle \right) (\widehat{r}_{j \leftarrow i}(t) - \widehat{r}_j(t)), \quad (8.29)$$

and approximate the variance for each individual message with the variance of the current belief, i.e., $\nu_{j \rightarrow i}^x(t+1) \approx \nu_j^x(t+1) = \mathbf{var}[x_j | \hat{r}_j(t), \nu_j^r(t); \theta]$. All expectations are taken with respect to the posterior distribution

$$p_{x|r}^{(t+1)}(x_j | \hat{r}_j(t)) \propto \exp \left(f_{\text{in}}(x_j, \theta) - \frac{(x_j - \hat{r}_j(t))^2}{2\nu_j^r(t)} \right). \quad (8.30)$$

As we did for the factor node computation, we define a variable node estimation function

$$G_x(\hat{r}, \nu^r, \theta) = \langle x | \hat{r}, \nu^r; \theta \rangle \quad (8.31)$$

from which we can compute the mean and variance of the new belief as

$$\hat{x}_j(t+1) = G_x(\hat{r}_j(t), \nu_j^r(t), \theta) = \langle x_j | \hat{r}_j(t), \nu_j^r(t); \theta \rangle \quad (8.32)$$

$$\nu_j^x(t+1) = \nu^r(t) G'_x(\hat{r}_j(t), \nu_j^r(t), \theta) = \mathbf{var}[x_j | \hat{r}_j(t), \nu_j^r(t); \theta]. \quad (8.33)$$

The estimate in Eq. (8.29) now simplifies to

$$\hat{x}_{j \rightarrow i}(t+1) \approx \hat{x}_j(t) - \nu_j^x(t) A_{ij} \hat{s}_i(t), \quad (8.34)$$

and we see there is a simple, linear adjustment to $\hat{x}_j(t)$ when computing specific messages to each factor node i . However, if we take this one step further and consider the computation of $\hat{p}_i(t+1)$ from Eq. (8.8), it becomes

$$\hat{p}_i(t+1) = \sum_j A_{ij} \hat{x}_j(t+1) - \hat{s}_i(t) \sum_j A_{ij}^2 \nu_j^x(t+1) \quad (8.35)$$

$$= \hat{z}_i(t+1) - \nu_i^p(t+1) \hat{s}_i(t). \quad (8.36)$$

Therefore, instead of making an adjustment to the message from each variable node to each factor node (this is $O(NM)$ extra computations), we simply have to subtract a single, linear correction term at each factor node ($O(N)$ extra computations) to the general mean estimate $\hat{z}_i(t+1) = \sum_j A_{ij} \hat{x}_j(t+1)$.

8.1.4 GAMP Algorithm

Our derivation has been for the MMSE variant of LBP, and real-valued \mathbf{x} , \mathbf{A} , and \mathbf{y} . This is the situation of primary interest to us, and the derivation was provided to give the reader context when extending it. However, the GAMP algorithm we present in Alg. 4 admits complex-valued quantities, and can also be used to compute the MAP estimate, i.e., it approximates the max-sum algorithm in Eq. (3). Where Alg. 4 is a generalization of our derivation, we provide specific examples in Table 8.1.

Algorithm 4 GAMP algorithm

- 1: Set, $t = 0$, $\widehat{s}_i(-1) = 0$, initialize $\widehat{x}_j(t)$ and $\nu_j^x(t)$ for $j \in \{1, M\}$, $i \in \{1, N\}$
 - 2: **repeat**
 - 3: $\widehat{z}_i(t) = \sum_j A_{ij} \widehat{x}_j(t)$
 - 4: $\nu_i^p(t) = \sum_j |A_{ij}|^2 \nu_j^x(t)$ ▷ factor-node input combination
 - 5: $\widehat{p}_i(t) = \widehat{z}_i(t) - \nu_i^p(t) \widehat{s}_i(t-1)$
 - 6: $\widehat{s}_i(t) = G_y(\widehat{p}_i(t), \nu_i^p(t), y_i)$ ▷ factor-node message approximation
 - 7: $\nu_i^s(t) = -G'_y(\widehat{p}_i(t), \nu_i^p(t), y_i)$
 - 8: $\nu_j^r(t) = [\sum_i |A_{ij}|^2 \nu_i^s(t)]^{-1}$ ▷ variable-node input combination
 - 9: $\widehat{r}_j(t) = \widehat{x}_j(t) + \nu_j^r(t) \sum_i A_{ij}^* \widehat{s}_i(t)$
 - 10: $\widehat{x}(t+1) = G_x(\widehat{r}_j(t), \nu_j^r(t), q_j)$ ▷ variable-node message approximation
 - 11: $\nu_j^x(t+1) = \nu_j^r(t) G'_x(\widehat{r}_j(t), \nu_j^r(t), q_j)$
 - 12: **until** Estimates converge, $\|\widehat{\mathbf{x}}(t+1) - \widehat{\mathbf{x}}(t)\| \leq \text{tolerance}$
-

8.2 Fitting GAMP to the Physics-of-Imaging Framework

The GAMP algorithm as presented in Alg. 4 is for an ideal situation in which a discrete vector is passed through a known matrix (composed of iid zero-mean random variables) before each component is measured. Our imaging system is much more complicated, as we are dealing with an optical field propagating through the turbulent atmosphere. Still, the core of the algorithm seems to match our problem for coherent imaging, where we assume knowledge of both the magnitude and phase of the on-

Function	MMSE	MAP
$G_x(\hat{r}, \nu^r, \theta)$	$\langle \mathbf{x} \hat{r}, \nu^r, q \rangle$	\hat{x}^0
$\nu^r G'_x(\hat{r}, \nu^r, \theta)$	$\mathbf{var}[\mathbf{x} \hat{r}, \nu^r, q]$	$\frac{\nu^r}{1 - \nu^r f''_{\text{in}}(\hat{x}^0, \theta)}$
$G_y(\hat{p}, \nu^p, y)$	$\frac{\langle \mathbf{z} \hat{p}, \nu^p, y \rangle - \hat{p}}{\nu^p}$	$\frac{\hat{z}^0 - \hat{p}}{\nu^p}$
$-G'_y(\hat{p}, \nu^p, y)$	$\frac{1}{\nu^p} \left[1 - \frac{\mathbf{var}[\mathbf{z} \hat{p}, \nu^p, y]}{\nu^p} \right]$	$\frac{f''_{\text{out}}(\hat{z}^0, y)}{\nu^p f''_{\text{out}}(\hat{z}^0, y) - 1}$
where $\hat{z}^0 = \operatorname{argmax}_z \left[f_{\text{out}}(z, y) - \frac{(z - \hat{p})^2}{2\nu^p} \right]$, $\hat{x}^0 = \operatorname{argmax}_x \left[f_{\text{in}}(x, \theta) - \frac{(x - \hat{r})^2}{2\nu^r} \right]$		

Table 8.1: The input and output estimation functions for the Sum-Product and Max-Sum versions of GAMP. The MAP calculation has been expanded by separately defining the argument maximum of the input and output posterior distributions, \hat{x}^0 and \hat{z}^0 .

target field, and make coherent (heterodyne-detection) measurements. In terms of the framework developed in Ch. 4, the field pattern at detector-plane location $\boldsymbol{\rho}$ for the i -th transmitted pulse is

$$\xi_b(\boldsymbol{\rho}) \propto \sum_j \sqrt{\mathcal{T}_j} \Phi_{ij}(\boldsymbol{\rho}) \quad (8.37)$$

where $\Phi_{ij}(\boldsymbol{\rho})$ is the detector-plane field produced by the scattering of the transmitted field from pixel j 's rough surface. This field pattern is a linear projection of the target's reflectivity pattern, but we still have two problems. First, coherent measurements are made by mixing the field with a local oscillator, and in order to make a good measurement the field must be constant over this mixing area. Since the returning field is speckled, we will need to employ a coherent detector array, which is technically challenging. Secondly, it will be almost impossible to know the on-target complex field pattern with enough accuracy to use with GAMP. We are propagating light through turbulent atmosphere over kilometers. So, even if the intensity profile on each pattern is not severely affected by the turbulence, the relative phase pixel-to-pixel will be. Moreover, any change to the optical path during the imaging process, including micron-scale movements of the target, will also corrupt estimates of the

phase. This type of imager could be simulated or made for a table-top experiment, but we do not expect it to be useful in the field.

Therefore, while we have made an initial connection between a coherent imaging system and GAMP, we now turn our focus to a more robust system. First, we will only assume knowledge of the on-target intensity pattern. We could still construct a heterodyne detector array and measure the full field of the returning light, but if we do not know the on-target phase pattern, the detected field’s pattern cannot be used by GAMP. To avoid the complication of needing multiple local oscillators and a detector array, we eschew coherent measurements in favor of direct detection, as is done with ghost imaging. Direct detection does not need an array, as it measures the *integrated intensity* across the bucket detector’s surface, and as the aperture increases in size it measures a stronger signal. This allows direct detection to perform aperture averaging, which mitigates target speckle [21] (for more on this phenomenon see the ghost imaging section in Ch. 3).

8.2.1 GAMP for Structured-Intensity Imaging

The connection between the physical model for structured-illumination intensity imaging with direct detection⁴ and GAMP initially seems daunting, but much of the required work was done in Ch. 4, when we discretized the physics framework for the Cramér-Rao bound (CRB) and linear minimum mean-squared error (LMMSE) estimator. In that chapter we showed that the measurement y_i was a Poisson shot-noise process whose rate parameter was the integrated photon flux across the detector surface. This measurement, in turn, could be broken into a signal term \mathbf{z}_i that was a linear transformation of the target, and a zero-mean additive noise term \mathbf{v}_i . Because the noise terms $\{\mathbf{v}_i\}$ are uncorrelated between pulses (and we presented evidence in Sec. 4.4 that they are independent), our measurements are therefore component-wise measurements of a linear transformation of the target.

⁴From this point forward we will be concerned exclusively with structured illumination imaging using an intensity reference. All future references to structured illumination will refer to this specific modality.

At this point we can identify three roadblocks to using the out-of-the-box GAMP algorithm on our imaging problem. The first is the inclusion of the additive zero-mean noise terms $\{\mathbf{v}_n\}$. From Ch. 4 we also know that $\mathcal{I}(p, p, n)$ is stochastic, so our transform matrix can only be estimated, not completely known. Finally, the GAMP algorithm assumes a zero-mean transformation matrix, whereas being a matrix of intensity values, \mathbf{A} will not have zero-mean elements. The consequences of these roadblocks are investigated, and in the sections that follow we derive solutions to them that preserve the GAMP framework.

8.2.2 Additive-Noise Term

The noise vector \mathbf{v} —arising from the nonlinear inter-pixel interference pattern—is simple to handle, given our approximation of its components as iid Gaussian random variables. Our measurements y_i are samples of the bucket detector’s photocurrent, and are random variables driven by the number of received photons, \mathcal{N}_i . The received photons are the sum of the contributions from the signal and noise terms, $\mathcal{N}_i = \mathbf{z}_i + \mathbf{v}_i$, where \mathbf{z}_i and \mathbf{v}_i are independent Gaussian random variables. Therefore \mathcal{N}_i is a Gaussian random variable with mean $\langle \mathbf{z}_i \rangle$ and variance $\mathbf{var}[\mathbf{z}_i] + \mathbf{var}[\mathbf{v}_i]$. Following the derivation in Sec. 8.1, the return message $\Delta_{j \leftarrow i}^t(x_j)$ is now approximated as

$$\Delta_{j \leftarrow i}^t(x_j) \approx \log \int d\mathcal{N}_i p_{\mathcal{N}}^t(\mathcal{N}_i | x_j) e^{f_{\text{out}}(y_i, \mathcal{N}_i)}. \quad (8.38)$$

Following through with the rest of the derivation, the only change is the additional noise on the forward message from j to i . Our modifications to GAMP in Alg. 4 are then to first replace the variance calculation of ν_i^p in line 4 of with the two lines

$$\nu_i^z(t) = \sum_j A_{ij}^2 \nu_j^x(t) \quad (8.39)$$

$$\nu_i^p(t) = \nu_i^z(t) + \nu_i^v(t). \quad (8.40)$$

The new variance is calculated from Eq. (4.39) as

$$\nu_i^v(t) = 2e^{4(\sigma_s^2 + \sigma_r^2)} \sum_j \sum_{\ell < j} \bar{I}_{ij} \bar{I}_{i\ell} \Gamma_0(j, \ell) \hat{x}_j(t) \hat{x}_\ell(t). \quad (8.41)$$

Since this is simply a noise term, we will approximate it by replacing the estimates $\hat{x}_j(t)$ with the prior mean \bar{x}_j , allowing it to be pre-calculated. Using the rank-one decomposition of the background matrix from Eq. (4.9), $\bar{\mathbf{I}} = \boldsymbol{\mu} \boldsymbol{\epsilon}^T$, Eq. (8.41) reduces to

$$\nu_i^v = 2e^{4(\sigma_s^2 + \sigma_r^2)} \mu_i^2 \sum_j \epsilon_j \bar{x}_j \sum_{\ell < j} \Gamma_0(j, \ell) \epsilon_\ell \bar{x}_\ell. \quad (8.42)$$

8.2.3 Matrix Uncertainty

Altering the GAMP algorithm so that it can accommodate a stochastic matrix has been investigated [48]. Our CLT approximation, when \mathbf{A} is deterministic and dense is that $\mathbf{z} = \mathbf{A}\mathbf{x}$ is a Gaussian random vector because \mathbf{x} is an iid random vector. If \mathbf{A} is composed of iid random variables as well, this is still true. Whereas before the mean and variance of \mathbf{z}_i could be directly calculated with Eqs. (8.8) and (8.9), we now need to use the iterated expectation formulas in Eqs. (4.34) and (4.35). In particular, we calculate means and variances conditioned on knowledge of the \mathbf{A} matrix, and then average over the uncertainty in the matrix, obtaining

$$\langle \mathbf{z}_i \rangle = \langle \langle \mathbf{z}_i | \mathbf{A} \rangle \rangle_{\mathbf{A}} = \sum_j \langle A_{ij} \rangle \hat{x}_j \quad (8.43)$$

$$\mathbf{var}[\mathbf{z}_i] = \langle \mathbf{var}[\mathbf{z}_i | \mathbf{A}] \rangle_{\mathbf{A}} + \mathbf{var}[\langle \mathbf{z}_i | \mathbf{A} \rangle] = \sum_j \langle A_{ij}^2 \rangle \nu_j^x + \sum_j \mathbf{var}[A_{ij}] \hat{x}_j^2 \quad (8.44)$$

We see that the mean is only modified by replacing the previously deterministic A_{ij} with the expected value of the new random matrix, which we will denote $\hat{A}_{ij} \equiv \langle A_{ij} \rangle$. The variance is a little more complicated, but defining $\nu_{ij}^A \equiv \mathbf{var}[A_{ij}]$ this can be

written as

$$\mathbf{var}[\mathbf{z}_i] = \sum_j \widehat{A}_{ij}^2 \nu_j^x + \sum_j \nu_{ij}^A (\nu_j^x + \widehat{x}_j^2). \quad (8.45)$$

The variance of \mathbf{z}_i is now the sum of two terms. The first is the noise term ν_i^z from Eq. (8.39), with $\widehat{\mathbf{A}}$ replacing \mathbf{A} , while the second is a new term resulting from the matrix uncertainty. In Sec. 8.2.2 we modified GAMP with Eqs. (8.39) and (8.40).

We now replace those same lines with

$$\nu_i^z(t) = \sum_j \widehat{A}_{ij}^2 \nu_j^x(t) \quad (8.46)$$

$$\nu_i^p(t) = \nu_i^z(t) + \sum_j \nu_{ij}^A (\nu_j^x(t) + \widehat{x}_j^2(t)) + \nu_i^v, \quad (8.47)$$

where ν_i^v is the additional variance from inter-pixel interference, defined in Eq. (8.42).

Propagating this change through our derivation in Sec. 8.1, we note that all remaining instances of A_{ij} come from the Taylor expansion of $\Delta_{j \leftarrow i}^t(x_j)$, and are associated with the mean of \mathbf{z}_i ; therefore, they will all now be \widehat{A}_{ij} . As a consequence, when computing $\widehat{x}_{j \rightarrow i}$ in Eq. (8.34) the correction term is now $-\nu_j^x \widehat{A}_{ij} \widehat{s}_i$. Going one step further, line 5 of Alg. 4 is computed using $\nu_i^z(t)$ instead of $\nu_i^p(t)$, viz.,

$$\widehat{p}_i(t) = \widehat{z}_i(t) - \nu_i^z(t) \widehat{s}_i(t-1). \quad (8.48)$$

For our particular scenario, the mean and variance of the measurement matrix are

$$\widehat{A}_{ij} = \left[\frac{\kappa_\alpha}{\kappa_1} I_{ij} + \left(1 - \frac{\kappa_\alpha}{\kappa_1} \right) \bar{I}_{ij} \right] \quad (8.49)$$

$$\nu_{ij}^A = \bar{I}_{np}^2 \left[e^{4(\sigma_s^2 + \sigma_r^2)} (\Gamma_0 + \Gamma_1 + \kappa_1 + 1) - \left(\frac{\kappa_\alpha^2}{\kappa_1} + 1 \right) \right], \quad (8.50)$$

which are derived from Eqs. (4.24) and (4.28).

8.2.4 Non-Centered Matrix

The primary difficulty in adapting GAMP for use in structured-illumination imaging using only intensity information is that the measurement matrix is now non-centered. To clarify this discussion we will assume there is no measurement noise, so that $\mathbf{y} = \mathbf{z}$, understanding that the non-centered matrix does not affect our ability to estimate z_i from y_i ; the trouble lies in our estimation of x_j from \mathbf{z} .

Given x_j , the vector \mathbf{z} has the Gaussian distribution

$$p_{\mathbf{z}|\mathbf{x}}(\mathbf{z}|x_j) \propto \exp\left(-\frac{1}{2}(\mathbf{z} - \widehat{\mathbf{z}}^j)^T \mathbf{C}_{\mathbf{z}|x_j}^{-1} (\mathbf{z} - \widehat{\mathbf{z}}^j)\right). \quad (8.51)$$

where $\widehat{\mathbf{z}}^j$ is the conditional mean vector

$$\widehat{z}_i^j = \sum_{\ell \neq j} \widehat{A}_{i\ell} \widehat{x}_\ell + \widehat{A}_{ij} x_j = \widehat{z}_i + \widehat{A}_{ij} (x_j - \widehat{x}_j). \quad (8.52)$$

Because $\mathbf{C}_{\mathbf{z}|x_j}$ does not depend on x_j , we approximate $\mathbf{C}_{\mathbf{z}|x_j} \approx \mathbf{C}_{\mathbf{z}}$, which is analogous to the approximations made in the GAMP derivation for the $\{\nu_i^p\}$. Now, consider the basic GAMP framework, in which the matrix \mathbf{A} is deterministic and composed of zero-mean, iid random variables. The covariance matrix elements are then

$$(\mathbf{C}_{\mathbf{z}})_{nm} = \sum_j A_{nj} \nu_j^x A_{mj}. \quad (8.53)$$

When $m \neq n$, this is the sum of zero-mean random variables, and so all off-diagonal elements of $\mathbf{C}_{\mathbf{z}}$ are zero-mean. The diagonal elements are $\sum_j A_{nj}^2 \nu_j^x$, and as the size of \mathbf{x} grows $\mathbf{C}_{\mathbf{z}}$ becomes strongly diagonal. The distribution of \mathbf{z} is then well approximated by the factored distribution $p_{\mathbf{z}}(\mathbf{z}) \approx \prod_i p_{\mathbf{z}}(z_i)$, because the cross-correlations among elements of \mathbf{z} are random and become relatively weak as the number of variables grows. This is why the GAMP algorithm works: the factored distribution for \mathbf{z} is a good approximation to the true joint distribution, and the joint log-likelihood calculation of x_j given \mathbf{z} is well approximated by a combination of the marginal log-

likelihood calculations that comprise the factor node messages, viz.,

$$\log p_{\mathbf{z}|\mathbf{x}}^t(\mathbf{z}|x_j) \approx \sum_i \Delta_{j \leftarrow i}^t(x_j). \quad (8.54)$$

Consequently, each iteration of the algorithm will be close to calculating the true likelihood, and the errors should be random and unbiased (since part of the covariance matrix that is dropped is zero-mean and iid). In essence, we are approximating \mathbf{C}_z with a diagonal matrix whose diagonal elements are $\sum_j A_{nj}^2 \nu_\ell^x$.

However, the story is different when \mathbf{A} is non-centered and stochastic. From our framework in Ch. 4 we can decompose the matrix as $\mathbf{A} = \tilde{\mathbf{A}} + \bar{\mathbf{A}}$ where $\tilde{\mathbf{A}}$ is composed of zero-mean, iid random variables, and $\bar{\mathbf{A}} = \boldsymbol{\mu}\boldsymbol{\epsilon}^T$ is a rank-one matrix. Our measurements vector can then be written

$$\mathbf{z} = \tilde{\mathbf{A}}\mathbf{x} + \bar{\mathbf{A}}\mathbf{x}. \quad (8.55)$$

The covariance matrix for this situation was derived in Eq. (6.21) as

$$\mathbf{C}_z = \tilde{\mathbf{A}}\mathbf{C}_x\tilde{\mathbf{A}}^T + \langle \tilde{\mathbf{C}}_{z|\mathbf{x}} \rangle + \boldsymbol{\mu}\boldsymbol{\mu}^T\Omega, \quad (8.56)$$

where $\langle \tilde{\mathbf{C}}_{z|\mathbf{x}} \rangle$ is a diagonal matrix, and scaling factor Ω is defined in Eq. (6.22). The first term on the right side of the equality is what we had in Eq. (8.53), when \mathbf{A} had no background component, and $\langle \tilde{\mathbf{C}}_{z|\mathbf{x}} \rangle$ is purely a result of matrix uncertainty. If the covariance matrix was just these first two components, it would be strongly diagonal and we could approximate the joint distribution of \mathbf{z} with its factored form. However, the rank-one component of the background, $\boldsymbol{\mu}\boldsymbol{\mu}^T\Omega$, means the components of \mathbf{z} are now all correlated. Thus the joint distribution of \mathbf{z} is not well approximated by the factored form, and we should not expect GAMP to work.

To circumvent the preceding problem, we note that if we knew the value of the vector $\bar{\mathbf{A}}\mathbf{x}$, then this rank-one component of the covariance matrix would disappear, and the distribution for \mathbf{z} would be well approximated by its factorized form. Fur-

thermore, since $\bar{\mathbf{A}}$ is rank-one, we only need to know a single auxiliary variable,

$$\mathbf{x}_+ = \boldsymbol{\epsilon}^T \mathbf{x} = \sum_{j=1}^M \epsilon_j x_j, \quad (8.57)$$

from which we could deterministically compute $\bar{\mathbf{A}}\mathbf{x} = \boldsymbol{\mu}\mathbf{x}_+$.

Our approach to dealing with a non-centered measurement matrix is thus to explicitly include \mathbf{x}_+ as a variable in our distribution, and extend the graph accordingly. We then evaluate the new graph with the GAMP assumptions, derive a modified algorithm, and show that in this graph the posterior likelihood of x_j , given \mathbf{z} , is well approximated by the combination of messages as in Eq. (8.54).

Extending the Graph

To define a new graph, we need to determine the factor potentials of the new joint distribution,

$$\begin{aligned} p_{\mathbf{y}, \mathbf{x}_+, \mathbf{x}}(\mathbf{y}, x_+, \mathbf{x}) &= p_{\mathbf{y}|\mathbf{x}_+, \mathbf{x}}(\mathbf{y}|x_+, \mathbf{x}) p_{x_+|\mathbf{x}}(x_+|\mathbf{x}) p_{\mathbf{x}}(\mathbf{x}) \\ &= p_{x_+|\mathbf{x}}(x_+|\mathbf{x}) \prod_i p_{y_i|\mathbf{x}_+, \mathbf{x}}(y_i|x_+, \mathbf{x}) \prod_j p_{\mathbf{x}}(x_j). \end{aligned}$$

The factorization is mostly the same as before: the prior distributions $p_{\mathbf{x}}(x_j)$ still define input potential functions, and the conditional measurement distributions define output potential functions. However, the measurement distributions are now functions of our new variable \mathbf{x}_+ , and so each output factor node will be connected to the new variable node. We also add a new factor node $f_+(x_+, \mathbf{x})$, to account for the new factor

$$p_{x_+|\mathbf{x}}(x_+|\mathbf{x}) = \delta(x_+ - \boldsymbol{\epsilon}^T \mathbf{x}), \quad (8.58)$$

which enforces the constraint that \mathbf{x}_+ is the weighted mean of the variables \mathbf{x} . As a result, this factor node is also connected to every variable node. The resulting augmented graph is shown in Fig. 8-2.

The original factor nodes can still be written as scalar functions of a linear transform of the target variables, i.e., $p_{y|x_+, \mathbf{x}}(y_i|x_+, \mathbf{x}) \equiv p_{y|z}(y_i|z_i)$ but each z_i is now computed as

$$\mathbf{z}_i = \sum_j \tilde{A}_{ij} \mathbf{x}_j + \mathbf{x}_+ \mu_i. \quad (8.59)$$

Moreover, because $p_{x_+|x}(x_+|x)$ is also function of a linear transform of the target variables, we define an additional transformation variable

$$\mathbf{z}_+ = \mathbf{x}_+ - \boldsymbol{\epsilon}^T \mathbf{x}, \quad (8.60)$$

so we can again write all of the temporary variables at the factor nodes as linear transformations of all of the target variables, i.e., we define the augmented vectors and matrix

$$\mathbf{x} \leftarrow \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_+ \end{bmatrix} \quad \mathbf{z} \leftarrow \begin{bmatrix} \mathbf{z} \\ \mathbf{z}_+ \end{bmatrix} \quad \underline{\mathbf{A}} = \begin{bmatrix} \tilde{\mathbf{A}} & \boldsymbol{\mu} \\ \boldsymbol{\epsilon}^T & -1 \end{bmatrix}, \quad (8.61)$$

where \mathbf{x} is now indexed from 1 to $M + 1$, \mathbf{z} from 1 to $N + 1$, $\underline{\mathbf{A}}$ is an $(N + 1)$ by $(M + 1)$ matrix, and

$$\mathbf{z} = \underline{\mathbf{A}} \mathbf{x}. \quad (8.62)$$

Because we are merely appending elements to the previous \mathbf{x} and \mathbf{z} vectors, there is no ambiguity when referring to the elements of the augmented vectors, so we have maintained the previous notation. However, there can be some ambiguity regarding the measurement matrix because the original elements are modified. That is why we have elected to define a new augmented matrix $\underline{\mathbf{A}}$. Our new factor's potential⁵ is $f_{\text{out}}^{(N+1)}(z_{N+1}) = \log \delta(z_{N+1})$. If we assume a measurement of $y_+ = y_{N+1} = 0$, this can

⁵We define the potential $f_{\text{out}}^{(N+1)}(z_{N+1})$ this way to maintain consistency with our previous notation, even though the log of a delta function is poorly defined. However, we will only every use $e^{f_{\text{out}}^{(N+1)}(z_{N+1})} = \delta(z_{N+1})$ in computations, so in practice this is not an issue.

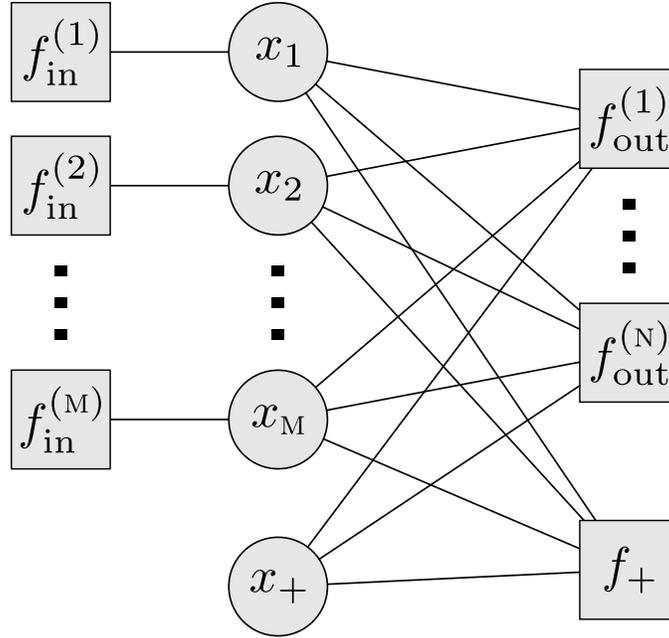


Figure 8-2: The factor graph for the GAMP algorithm when there is a non-centered matrix. An additional variable node is introduced to estimate x_+ , the weighted mean of the target vector \mathbf{x} . A new factor node f_+ is also introduced, which enforces the consistency of this variable with the empirical mean of \mathbf{x} during each iteration of the algorithm.

be viewed as a measurements channel viz., $p_y(y_{N+1}|z_{N+1}) = \delta(y_{N+1} - z_{N+1})$, where the measurement is always $y_{N+1} = 0$.

8.2.5 Reconciling with the GAMP Algorithm

Having changed the underlying graph, we need to re-derive an approximate algorithm. Fortunately, this turns out to be a slightly modified version of GAMP. For this derivation we assume that the message from the new variable node x_{M+1} to each factor node f_i is a Gaussian distribution, which we will justify when calculating that message at the end of the section. The rest of this section breaks down as follows. First, we look at the assumptions and computations at the factor nodes. This is done separately for the original factor nodes $\{f_i : 1 \leq i \leq N\}$, and the new factor node node f_{N+1} . We then look at the recombination of messages at the variable nodes. Again, we treat the original nodes $\{x_j : 1 \leq j \leq M\}$ differently than the new node x_{M+1} . Finally, we compare behavior of this combination of the messages at each variable node to a

true computation of the log-likelihood of x_j , and show that our approach is a good approximation. We thus have remedied the root cause that prevented GAMP from being applied to systems with non-centered matrices.

Factor-Node Computations

We start with the basic CLT assumption that \mathbf{z} is Gaussian distributed. Consider the original components $\{\mathbf{z}_i : 1 \leq i \leq N\}$, for which $\mathbf{z}_i = \sum_{j=1}^M \tilde{A}_{ij} \mathbf{x}_j + \mu_i \mathbf{x}_{M+1}$. Both $\sum_{j=1}^M \tilde{A}_{ij} \mathbf{x}_j$ and $\mu_i \mathbf{x}_{M+1}$ are Gaussian random variables by the CLT, but we can also show they are jointly Gaussian random variables with almost no cross-correlation⁶. Therefore, their sum, $\{\mathbf{z}_i : 1 \leq i \leq N\}$ is still a Gaussian random variable. For the new factor node the story is different, but we still have that the $\sum_{j=1}^M \epsilon_j \mathbf{x}_j$ and \mathbf{x}_{M+1} are jointly Gaussian because $\sum_{j=1}^M \epsilon_j \mathbf{x}_j$ and \mathbf{x}_+ equals that sum. It follows that we can treat the elements of \mathbf{z} as Gaussian random variables. Therefore, at each factor node we can combine incoming messages into a mean and variance as with GAMP, where.

$$\hat{p}_i(t) = \sum_{j=1}^{M+1} \hat{A}_{ij} \hat{x}_{j \rightarrow i}(t) \quad (8.63)$$

$$\nu_i^z(t) = \sum_{j=1}^{M+1} \hat{A}_{ij}^2 \nu_j^x(t) \quad (8.64)$$

$$\nu_i^p(t) = \nu_i^z(t) + \sum_{j=1}^{M+1} \nu_{ij}^A(\nu_j^x(t) + \hat{x}_j^2(t)) + \nu_i^v. \quad (8.65)$$

for all $1 \leq i \leq N$.

Since we have maintained the form of the messages coming into the factor nodes, the computations there are unchanged. We take the Taylor expansion, and make the same computation of \hat{s}_i and ν_i^s . Since the potential $f_{\text{out}}^{(N+1)}$ is driven by the delta

⁶They are jointly Gaussian because arbitrary linear combinations are also Gaussian random variables. The individual variances are $\mathbf{var}[\sum_j \tilde{A}_{ij} \mathbf{x}_j] = \sum_j \tilde{A}_{ij}^2 \nu_j^x$ and $\mathbf{var}[\mu_i \mathbf{x}_{M+1}] = \mu_i^2 \sum_j \epsilon_j^2 \nu_j^x$, both of which are sums of positive quantities. The covariance is $\mathbf{cov}[\sum_j \tilde{A}_{ij} \mathbf{x}_j, \mu_i \mathbf{x}_{M+1}] = \mu_i \sum_j \tilde{A}_{ij} \epsilon_j \nu_j^x$ which is the sum of zero-mean quantities, and by the strong law of large numbers becomes much smaller than the two variances.

function $\delta(\mathbf{z}_{N+1})$ we can simplify the computations at the new node to

$$\widehat{\mathbf{s}}_{N+1}(t) = -\frac{\widehat{p}_{N+1}(t)}{\nu_{N+1}^p(t)} \quad (8.66)$$

$$\nu_{N+1}^s = \frac{1}{\nu_{N+1}^p(t)}. \quad (8.67)$$

Variable-Node Computations

At the original variable nodes we compute the estimates and variances $\{(\widehat{r}_j, \nu_j^r) : j \leq 1 \leq M\}$ in the same manner as before, only we now use the augmented vectors and measurement matrix, $\widehat{\mathbf{s}}, \boldsymbol{\nu}^s$, and $\underline{\mathbf{A}}$. The new variable node \mathbf{x}_{M+1} is not attached to a factor node for a prior distribution, so it generates its outgoing messages as

$$\Delta_{M+1 \rightarrow i}^{t+1}(x_{m+1}) = \sum_{k \neq i} \Delta_{M+1 \leftarrow k}^t(x_{M+1}). \quad (8.68)$$

We still assume the summation of incoming measurements is well approximated by a Gaussian distribution, so that $\sum_{k \neq i} \Delta_{M+1 \leftarrow k}^t(x_{M+1})$ becomes a scalar log-likelihood function defined by the estimate $\widehat{r}_{M+1 \leftarrow i}(t)$ and variance $\nu_{M+1 \leftarrow i}^r(t)$. We then immediately have the mean and variance of the return messages,

$$(\widehat{x}_{M+1 \rightarrow i}(t+1), \nu_{M+1 \rightarrow i}^x(t+1)) = (\widehat{r}_{M+1 \leftarrow i}(t), \nu_{M+1 \leftarrow i}^r(t)). \quad (8.69)$$

However, Eq. (8.69) does not perfectly fit with the standard GAMP messages. To make that happen we employ our standard variance approximations of $\nu_{M+1 \leftarrow i}^r(t) \approx \nu_{M+1}^r(t)$ and $\nu_{M+1 \rightarrow i}^x(t+1) \approx \nu_{M+1}^x(t+1)$, so that the estimate $\widehat{r}_{M+1 \leftarrow i}(t)$ becomes

$$\widehat{r}_{M+1 \leftarrow i}(t) = \widehat{r}_{M+1}(t) - \nu_{M+1}^r(t) \underline{\widehat{A}}_{i, M+1} \widehat{\mathbf{s}}_i(t) \quad (8.70)$$

where

$$\nu_{M+1}^r(t) = \left[\sum_i \hat{A}_{iM+1}^2 \nu_i^s(t) \right]^{-1} \quad (8.71)$$

$$\hat{r}_{M+1}(t) = \hat{x}_{M+1}(t) + \nu_{M+1}^r(t) \sum_i \hat{A}_{iM+1} \hat{s}_i(t). \quad (8.72)$$

We have now recovered the GAMP form, and the mean and variance of the new messages are

$$\hat{x}_{M+1}(t+1) = \hat{r}_{M+1}(t) \quad (8.73)$$

$$\nu_{M+1}^x(t+1) = \nu_{M+1}^r(t) \quad (8.74)$$

and the estimates for the message from $M+1$ to i are found with the correction

$$\hat{x}_{M+1 \rightarrow i}(t+1) = \hat{x}_{M+1}(t+1) - \nu_{M+1}^x(t) \hat{A}_{iM+1} \hat{s}_i(t). \quad (8.75)$$

The messages being sent from \mathbf{x}_{M+1} are indeed Gaussian distributed, as was required for our derivation. We also note that for the original variable nodes, this correction term came from a first-order approximation, whereas at this node it is an exact calculation (after making the variance simplification).

Verify Correctness of Messages

Our motivation for modifying the GAMP algorithm was that when the matrix \mathbf{A} was non-centered, the combination of messages at variable node j was no longer a good approximation for the log-likelihood of \mathbf{x}_j . We showed this to be so for a simplified framework in which there was no measurement noise, and the measurements were just the transformed variable, i.e., $\mathbf{y} = \mathbf{z}$.

To verify that our modifications will yield useful messages, we need to show that the combination of messages at variable node j is now a good approximation of the log-likelihood of \mathbf{x}_j , given measurements \mathbf{z} , i.e., the logarithm of the distribution in Eq. (8.51). Although this is indeed true, the proof is somewhat tedious, so we have

elected to present it in App. C. Herein we will simply outline the approach.

We denote the j -th column of matrix \mathbf{A} as \mathbf{A}_j so that the conditional mean vector $\widehat{\mathbf{z}}^j$ (whose components are given in Eq. (8.52)) can be written in vector form as

$$\widehat{\mathbf{z}}^j = \widehat{\mathbf{z}} + \mathbf{A}_j(x_j - \widehat{x}_j). \quad (8.76)$$

Then, the logarithm of Eq. (8.51) can be expanded into a quadratic function of x_j ,

$$\begin{aligned} \log p_{\mathbf{z}|\mathbf{x}}(\mathbf{z}|x_j) &= -\frac{1}{2}(\mathbf{z} - \widehat{\mathbf{z}}^j)^T \mathbf{C}_{\mathbf{z}}^{-1}(\mathbf{z} - \widehat{\mathbf{z}}^j) + \text{const} \\ &= -\frac{1}{2}(\mathbf{z} - \widehat{\mathbf{z}} - \mathbf{A}_j(x_j - \widehat{x}_j))^T \mathbf{C}_{\mathbf{z}}^{-1}(\mathbf{z} - \widehat{\mathbf{z}} - \mathbf{A}_j(x_j - \widehat{x}_j)) + \text{const} \\ &= -\frac{x_j^2}{2} \mathbf{A}_j^T \mathbf{C}_{\mathbf{z}}^{-1} \mathbf{A}_j + x_j \mathbf{A}_j^T \mathbf{C}_{\mathbf{z}}^{-1} \mathbf{A}_j \widehat{x}_j + x_j \mathbf{A}_j^T \mathbf{C}_{\mathbf{z}}^{-1}(\mathbf{z} - \widehat{\mathbf{z}}) + \text{const}. \end{aligned} \quad (8.77)$$

where the constant term changes line-to-line but does not depend on x_j . This function is quadratic in x_j , so it can be rearranged into the exponent of a Gaussian distribution, $-(\widehat{r}_j - x_j)^2/2\nu_j^r$. The returning messages in GAMP also form a Gaussian distribution. Our goal is then to show that the values of \widehat{r}_j and ν_j^r resulting from the decomposition of the log-likelihood distribution in Eq. (8.77) match those from GAMP,

$$\widehat{r}_j = \widehat{x}_j + \nu_j^r \sum_{i=1}^{N+1} \widehat{A}_{ij} \widehat{s}_i \quad (8.78)$$

$$\nu_j^r = \left[\sum_{i=1}^{N+1} \widehat{A}_{ij} \nu_i^s \right]^{-1}. \quad (8.79)$$

In App. C we show that in a large-number approximation⁷ these two computations converge. Therefore, for a sufficiently large number of variables and measurements our GAMP modification should compute accurate estimates.

⁷We increase both the number of pixels M and measurements N , but keep their ratio fixed.

Initializing GAMP for Structured-Intensity Imaging

In adapting GAMP to our imaging scenario we overcame three disconnects. Handling the additive noise \mathbf{v} and the uncertainty in the transformation matrix were fairly straightforward, and fit with the CLT approximations already being made. Compensating for the non-centered matrix was more complicated, but we were able to augment the graph to form useful messages. There is still, however, one remaining unknown parameter of the system.

Our model for rough-surface reflection predicts that the total light it returns is proportional to its coherence area, $\pi\varrho_T^2$, which our analysis has assumed to satisfy $\pi\varrho_T^2 = \lambda_0^2$. This assumption is fine for assessing the imager’s spatial resolution and its SNR, but it is deficient for our modified version of GAMP. Because this term scales the strength of the measurement, misjudging it causes our prior to be mismatched, which can result in GAMP failing to converge.

In Ch. 9 we explore methods to learn parameters during the running of GAMP, but we still need to initialize the algorithm accurately. Our approach to initialization is as follows. First, we calculate an initial Fresnel number product \mathcal{F}_0 from Eq. (4.5), assuming $\varrho_T^2 = \lambda_0^2/\pi$. We then presume an iid prior on our target pixels, with mean \bar{x} . This prior implies that our predicted mean for measurement i is $\bar{y}_i = \bar{x} \sum_j \hat{A}_{ij}$. If our model is correct, the empirical average of the $\{\bar{y}_i\}$ should be the same as the empirical average of the measurements $\{y_i\}$. If this is not the case, we can correct the Fresnel number product using

$$\mathcal{F} = \mathcal{F}_0 \frac{\sum_i y_i}{\bar{x} \sum_{i,j} \hat{A}_{ij}}, \quad (8.80)$$

The new version of GAMP, including all of these adjustments, is shown in Alg. 5. It should be noted that the adjustment for the non-centered matrix was derived for the MMSE estimate, but applies just as well for the MAP estimate. Since the messages returned to \mathbf{x}_+ are Gaussian, the mean is the most likely value and the MAP and MMSE estimates are the same. At the new factor node, the measurement distribution is a delta function, and again the MMSE estimate of \mathbf{z}_+ is the same as the

MAP estimate, i.e., both are 0. Therefore, Alg. 5 works equally well for the MMSE and MAP criteria.

Algorithm 5 GAMP algorithm for structured illumination

For sensing matrix \mathbf{A} with mean $\widehat{\mathbf{A}}$ and variance $\boldsymbol{\nu}^A$, and envelope $\boldsymbol{\epsilon}$:

- 1: **Center measurement matrix:** $\mu_i = \sum_j \widehat{A}_{ij}$, $\widetilde{\mathbf{A}} = \widehat{\mathbf{A}} - \boldsymbol{\mu}\boldsymbol{\epsilon}^T$
 - 2: **Augment matrices:** $\widehat{\underline{\mathbf{A}}} \leftarrow \begin{bmatrix} \widetilde{\mathbf{A}} & \boldsymbol{\mu} \\ \boldsymbol{\epsilon}^T & -1 \end{bmatrix}$, $\boldsymbol{\nu}^{\underline{\mathbf{A}}} \leftarrow \begin{bmatrix} \boldsymbol{\nu}^A & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix}$
 - 3: **Augment vectors:** $\mathbf{x} \leftarrow \begin{bmatrix} \mathbf{x} \\ x_+ \end{bmatrix}$, $\mathbf{z} \leftarrow \begin{bmatrix} \mathbf{z} \\ z_+ \end{bmatrix}$, $\mathbf{y} \leftarrow \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix}$
 - 4: **New nodes:** $G_y(\widehat{p}_{N+1}, \nu_{N+1}^p, y_{N+1}) = -\frac{\widehat{p}_{N+1}}{\nu_{N+1}^p}$, $G_x(\widehat{r}_{M+1}, \nu_{M+1}^r, \theta) = \widehat{r}_{M+1}$
 - 5: **Initialize:** $t = 0$, $\widehat{s}_i(-1) = 0$, $\widehat{x}_j(t)$ and $\nu_j^x(t)$ for $j \in \{1, M+1\}$, $i \in \{1, N+1\}$
 - 6: **repeat**
 - 7: $\widehat{z}_i(t) = \sum_j \widehat{A}_{ij} \widehat{x}_j(t)$
 - 8: $\nu_i^z(t) = \sum_j |\widehat{A}_{ij}|^2 \nu_j^x(t)$
 - 9: $\nu_i^p(t) = \nu_i^z(t) + \sum_j \nu_{ij}^A (\nu_j^x(t) + \widehat{x}_j^2(t)) + \nu_i^v$ \triangleright factor-node input combination
 - 10: $\widehat{p}_i(t) = \widehat{z}_i(t) - \nu_i^z(t) \widehat{s}_i(t-1)$
 - 11: $\widehat{s}_i(t) = G_y(\widehat{p}_i(t), \nu_i^p(t), y_i)$ \triangleright factor-node message approximation
 - 12: $\nu_i^s(t) = -G'_y(\widehat{p}_i(t), \nu_i^p(t), y_i)$
 - 13: $\nu_j^r(t) = \left[\sum_i |\widehat{A}_{ij}|^2 \nu_i^s(t) \right]^{-1}$ \triangleright variable-node input combination
 - 14: $\widehat{r}_j(t) = \widehat{x}_j(t) + \nu_j^r(t) \sum_i \widehat{A}_{ij}^* \widehat{s}_i(t)$
 - 15: $\widehat{x}_j(t+1) = G_x(\widehat{r}_j(t), \nu_j^r(t), q)$ \triangleright variable-node message approximation
 - 16: $\nu_j^x(t+1) = \nu_j^r(t) G'_x(\widehat{r}_j(t), \nu_j^r(t), q)$
 - 17: **until** Estimates converge, $\|\widehat{\mathbf{x}}(t+1) - \widehat{\mathbf{x}}(t)\| \leq \text{tolerance}$
-

8.3 Comparison to LMMSE Estimate and CRB

We have taken the original GAMP algorithm in Alg. 4 and extended it in Alg. 5 to account for the noise caused by inter-pixel interference and speckle. We also modified the underlying graphical framework to include an extra variable node and an extra factor node, which allowed us to use GAMP despite the measurement matrix being non-centered. Now that GAMP has been adapted to our imaging scenario, it is time to

see how its performance compares to that of our current best estimator, the LMMSE estimator. This comparison is done using our simulator, which was first presented in Sec. 4.4.

In our initial comparisons in Chs. 5 and 6, we focused on the MSE performance as the number of measurements increased. These results were presented for two different values of β , the ratio of the detector's area to the transmitter's area. From that analysis we reached a few conclusions. First, the reconstructions were not of high quality, and better methods were required. This observation spurred the work in the present chapter. However, we also noticed that there were diminishing returns from additional measurements once you had as many measurements as there were pixels. Moreover, reconstruction quality dramatically improved as the bucket detector size was increased. Therefore, in this chapter and the next, we look at performance as a function of bucket detector size, for two different sampling ratios: first when we have a 0.8 undersampling (the number of measurements N is $0.8M$, where M is the number of pixels), and secondly when $N = M$.

Because we are now using many fewer measurements, but are looking at a larger range of bucket detector sizes, we re-ran our simulator. The remainder of the thesis uses a new dataset for the same scenes, with the resolution increased so that the images are 100×100 instead of 64×64 . The same example scenes for this new resolution are presented in Fig. 8-3 along with their speckled versions. In Ch. 5 we showed that each pixel in the scene is modulated by a speckle coefficient, so if no additional information about the target is used to eliminate the speckle, we cannot hope to perfectly reconstruct the original scene. Our estimators treat the speckle as randomness in the sensing matrix and will suppress some of it, but for a pixel-wise prior we expect the scene reconstructions to more closely resemble the speckled versions of the scenes than the original scenes.

Beyond this, we still have to contend with the inter-pixel interference noise vector \mathbf{v} from Eq. (4.32). When analyzing our simulator in Sec. 4.4, we found the effective signal-to-noise ratio (defined here as the ratio of variance of signal vector \mathbf{z} to noise vector \mathbf{v}) was $\sim \beta$. Therefore, even when the speckle variance is sufficiently suppressed

by increasing β , we still expect the inter-pixel interference to limit the quality of our reconstructions.

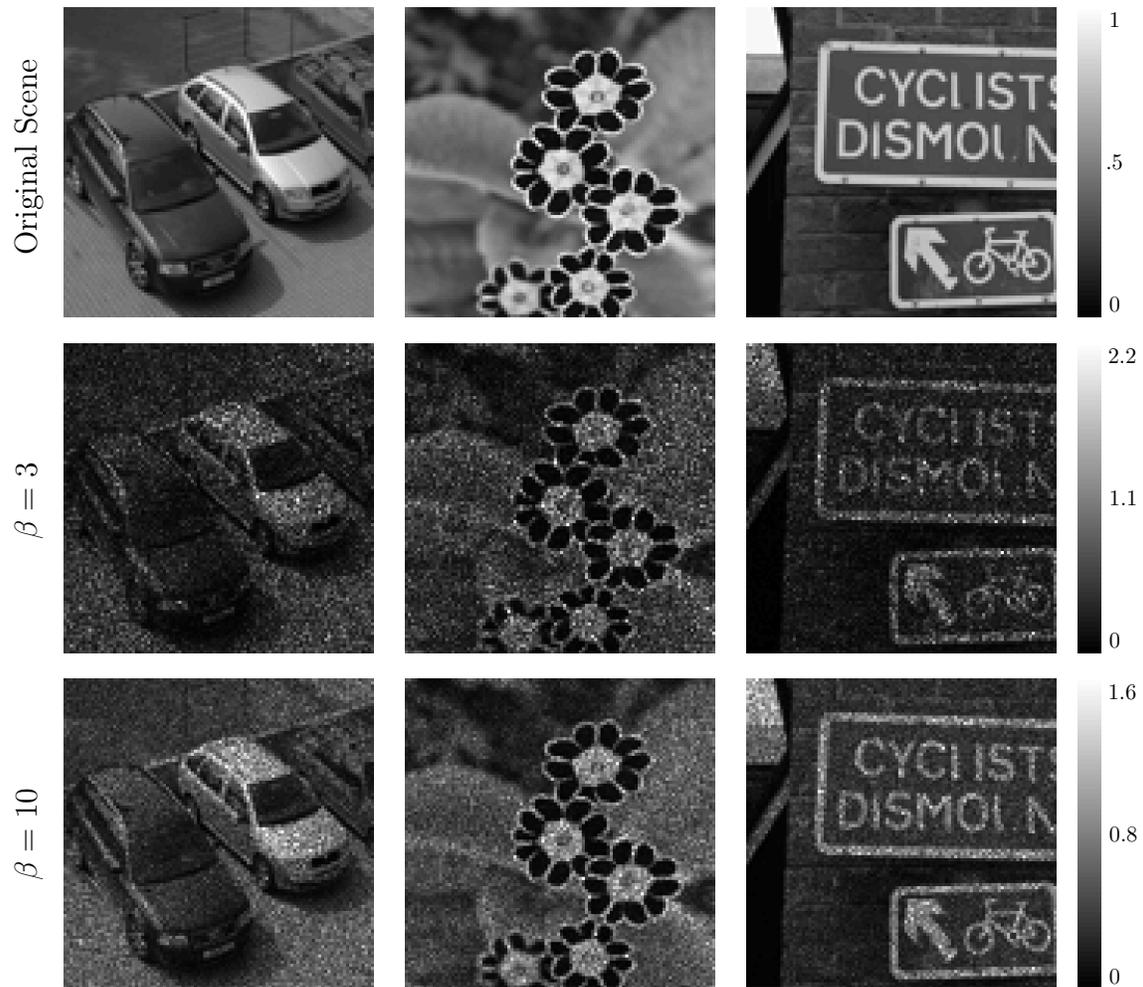


Figure 8-3: Three example scenes at a resolution of 100×100 are shown in the top row. The next two rows are the speckled reflectivity patterns of the scenes for $\beta = 3$ and $\beta = 10$.

We have also chosen to adjust how we compute the MSE. In our analysis of the LMMSE estimator in Ch. 6, the estimator had knowledge of the true mean and variance for each individual scene. This was done to facilitate comparison with the CRB, but is not practical for a real imager. We now only assume knowledge of the mean and variance over the entire dataset, not for particular scenes. However, this implies that the reconstructions will be for the wrong mean. This mismatch isn't a problem for our algorithm because we intelligently initialize the Fresnel number

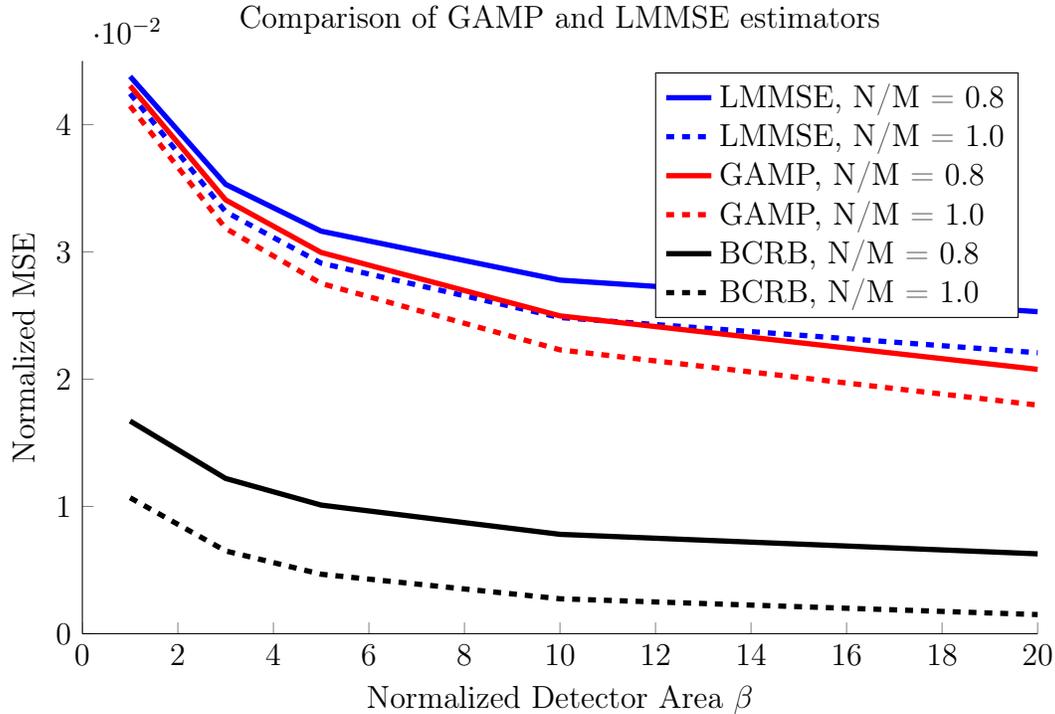


Figure 8-4: Plots versus normalized detector area of the MSE for the LMMSE and GAMP estimators (using and iid Gaussian prior for each pixel), with the BCRB provided for comparison. The MSE is computed for each image, and then averaged over our 100 image dataset.

product \mathcal{F} to correct for scaling issues, but it does imply that the final reconstruction will be incorrectly scaled. This scaling does not change the appearance of the image, but it does affect the MSE. Therefore, before computing the MSE of each image we re-scale it to correct for the discrepancy between the mean value of this particular image and the assumed prior mean used for estimation.

The result of this analysis for our new GAMP estimator is presented in Fig. 8-4. For comparison we have also presented the results for the LMMSE estimator (with the MSE now computed in the same manner), as well as the Bayesian Cramér-Rao bound (BCRB).

In Fig. 8-4 we see very little difference in the MSE of GAMP and the LMMSE estimator. While at first this seems surprising, it is a straightforward consequence of our chosen prior distribution. The LMMSE estimator is the minimum mean-squared error estimator when the target and measurements are jointly Gaussian random variables.

The Poisson noise process approaches a Gaussian distribution for many photons per measurement, so choosing a Gaussian distribution for the prior essentially enforces a jointly Gaussian distribution on the target and measurements at high photon numbers, meaning that GAMP can perform no better than the LMMSE estimator. As further confirmation of this behavior, we present a visual comparison of reconstructions of our test scenes, for different size bucket detectors, in Figs. 8-5 and 8-6.

The pixels in a real scene clearly do not follow an iid Gaussian distribution, and we will need a more accurate target model if we hope to improve our estimation. The LMMSE estimator is not affected by the choice of prior (beyond its mean and variance), but GAMP is. Instead of looking at different iid models for the pixels, in Ch. 9 we introduce a wavelet-basis model for the target, and demonstrate significant improvement in image quality (both visually and in MSE). We then extend the graph to include the image speckle coefficients, and by jointly estimating the wavelet-basis target and the speckle we achieve state-of-the art structured-illumination imaging.

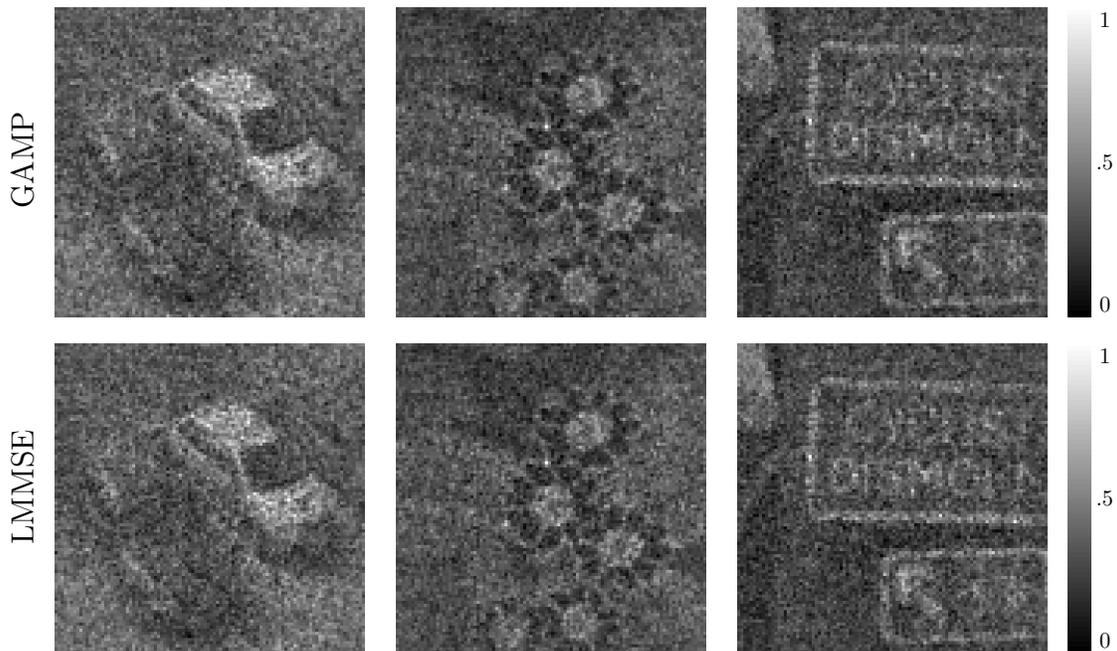


Figure 8-5: Images for the GAMP estimator (with a spatial-basis iid Gaussian prior) and the LMMSE estimator for $\beta = 3$. The top row are reconstructions using the GAMP algorithm, while the bottom row shows the LMMSE estimates.

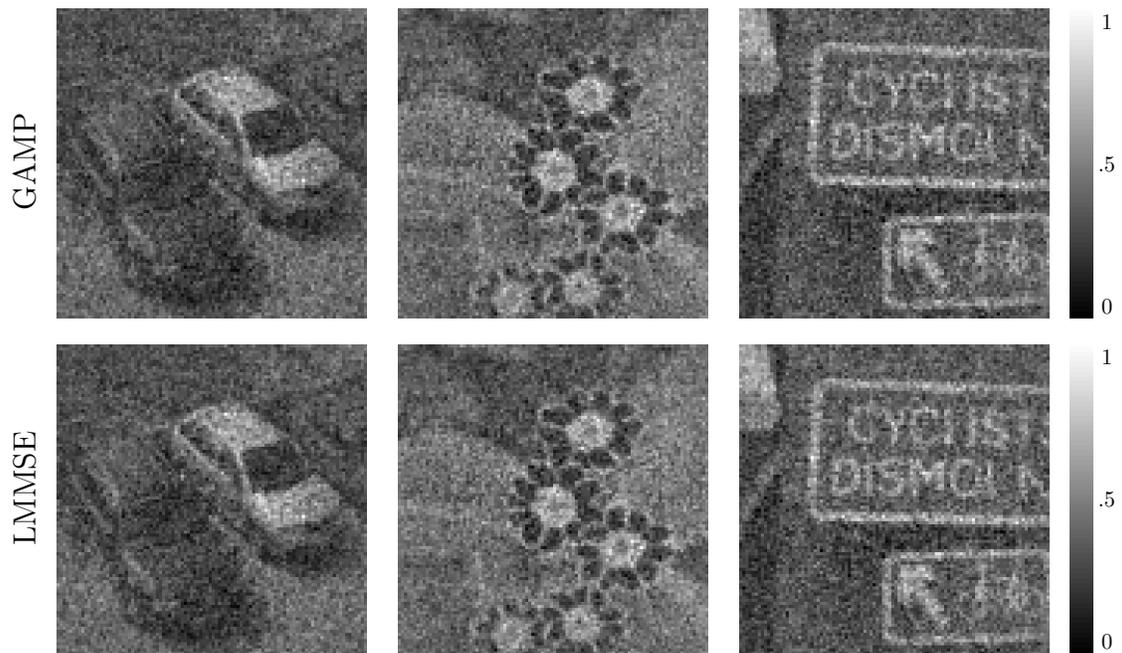


Figure 8-6: Image for the GAMP estimator (with a spatial-basis iid Gaussian prior) and the LMMSE estimator are presented for $\beta = 10$. The top row are reconstructions using the GAMP algorithm, while the bottom row shows the LMMSE estimates.

Chapter 9

Extending GAMP: Parameter Estimation, Wavelet-Basis Prior, and Speckle Mitigation

We have made an initial connection between structured illumination and the GAMP algorithm. Yet we are using a simplistic iid prior on the target pixels, treating the surface speckle as pure noise, and guessing several system parameters. At this point, GAMP’s performance is similar to that of a simple linear estimator.

In this chapter we use the richness of the graphical framework to extend and enhance our basic algorithm. We first use Adaptive GAMP methods [49–51] to learn system parameters as the algorithm runs. We then leverage the inherent structure of natural scenes to aid our reconstruction efforts by modeling the target with sparsifying priors in a wavelet basis. We also develop a model of surface speckle and explicitly expose the speckle contributions as additional variables in our model. Finally, we combine our wavelet-basis model of the target with our speckle model to estimate both quantities together, allowing us to intelligently remove the speckle from the image. This final modification provides state-of-the-art target reconstructions, and is the culmination of this thesis.

9.1 Adaptive GAMP

Despite intelligently initializing the algorithm with our adjustment in Eq. (8.80), we cannot know the correct parameterization for the target a priori. In Ch. 8 we assumed an iid Gaussian prior on each pixel, in which the mean and variance were found from ensemble averaging our dataset of scenes. However, the mean and variance for an individual scene can vary drastically from these values, and this prior mismatch degrades the reconstruction quality.

The EM-GAMP algorithm [49, 50] was proposed as a way of learning parameters of the prior distribution or output channel by continuously updating estimates of the parameters with calculations derived from the Expectation-Maximization (E-M) algorithm. We develop the E-M algorithm in App. B.5, but the core result is that we can estimate a hidden parameter using a set of related measurements via an iterative series of expectations and maximizations.

For our purposes we want to learn a parameter θ of the prior distribution on the target pixels, $p_{\mathbf{x}}(\mathbf{x}; \theta)$, from the measurements \mathbf{y} . The expectation and maximization can be compacted into a single line, so that the new estimate of the parameter, $\theta^{(t+1)}$, is

$$\theta^{(t+1)} = \operatorname{argmax}_{\theta} \langle \log p(\mathbf{x}; \theta) | \mathbf{y}; \theta^{(t)} \rangle_{\mathbf{x}}. \quad (9.1)$$

where $\theta^{(t)}$ is the previous estimate, and the expectation is taken with respect to the posterior distribution $p_{\mathbf{x}|\mathbf{y}}(x_j | \mathbf{y}; \theta^{(t)})$. The MMSE version of GAMP will produce this distribution at the end of its run, but for the update to be useful we would have to run GAMP many times. The key to the EM-GAMP method is that this posterior distribution is approximated during every iteration of the algorithm, allowing for the parameter to be updated *during* the running of GAMP.

In Sec. 8.2.4, we argued that GAMP works primarily because the combination of messages at the variable nodes $\{x_j\}$ is a good approximation for the log-likelihood of x_j , given the measurements. In GAMP the combination of messages at variable node

x_j approximates this likelihood with the Gaussian distribution

$$p_{r|x}(\hat{r}_j|x_j; \nu_j^r) = \frac{e^{-(\hat{r}_j-x_j)^2/2\nu_j^r}}{\sqrt{2\pi\nu_j^r}}, \quad (9.2)$$

and $p_{y|x}(\mathbf{y}|x_j; \theta^{(t)}) \approx p_{r|x}(\hat{r}_j|x_j; \nu_j^r)$. The posterior distribution of x_j for this iteration of the E-M algorithm is then

$$p_{x|y}(x_j|\mathbf{y}; \theta^{(t)}) \approx p_{x|r}(x_j|\hat{r}_j(t); \nu_j^r(t), \theta^{(t)}) = \frac{p_{r|x}(\hat{r}_j(t)|x_j; \nu_j^r(t))p_x(x_j; \theta^{(t)})}{p_r(\hat{r}_j; \nu_j^r, \theta^{(t)})}, \quad (9.3)$$

and the E-M update for the parameter θ is

$$\theta^{(t+1)} = \operatorname{argmax}_{\theta} \langle \log p(\mathbf{x}; \theta) | \hat{r}_j; \nu_j^r, \theta^{(t)} \rangle_{\mathbf{x}}. \quad (9.4)$$

This approach has been extended with the Adaptive GAMP algorithm [51], whose performance is predicted by a set of scalar state evolutions. The analysis therein showed that Adaptive GAMP produced asymptotically consistent estimates equivalent to an oracle algorithm which knew the parameters beforehand. Instead of using the E-M update, this variant finds the new parameter estimate by directly maximizing the log-likelihood of the parameter as

$$\theta^{(t+1)} = \operatorname{argmax}_{\theta} \frac{1}{M} \sum_{j=1}^M \log p_r(\hat{r}_j; \nu_j^r, \theta^{(t)}). \quad (9.5)$$

The analysis in [51] also showed asymptotic consistency with the EM-GAMP updates, so either can be used. At this point we note that the Adaptive GAMP method has analogous updates to learn parameters of the measurement channel, but in this thesis we are only concerned with learning parameters of the prior distribution, and so we do not incorporate measurement-channel parameter estimates into our algorithm.

To include the parameter updates in the GAMP algorithm we define a general

adaptation function, $H_x(\cdot)$, for the prior distribution, so that

$$\theta^{(t)} = H_x(\hat{\mathbf{r}}(t), \boldsymbol{\nu}^r(t)). \quad (9.6)$$

The adaptation function can implement either the EM-GAMP estimation function in Eq. (9.4), or the Adaptive GAMP estimation function in Eq. (9.5). This adaptation step is used to update the parameter estimate before computing the new means and variances $(\hat{\mathbf{x}}(t+1), \boldsymbol{\nu}^x(t+1))$. The adaptive GAMP variant is defined in Alg. 6. As it will be used extensively in the remainder of this thesis, it has also been functionalized to clarify its use. In this thesis we implement $H_x(\cdot)$ using the E-M style updates shown in Eq. (9.4).

9.2 Estimating the Target in a Wavelet Basis

One of the difficulties in a remote imaging scenario is the diversity of possible targets. Although there are several proposed models for real-world targets, one model cannot suffice for all scenarios. Thus, while it would be useful to extend our work to choose among a large class of models, we will restrict our focus to sparse representations in a wavelet basis.

We have previously modeled the target with an iid prior in a spatial basis, but this is a poor model for real-world scenes because they have significant inter-pixel correlations. Natural scenes are typically collections of objects that are generally smooth, but with sharp borders at their intersections. This piece-wise smooth structure is what enables image compression algorithms [52]; there is less information (as measured by entropy) in the scene than an iid pixel model would predict.

The heart of many image compression algorithms is finding an alternative representation that is (on average) sparse¹ for images of natural scenes [52]. The resulting transformed image can then be stored with an encoding that only retains the most significant components of the transformed image.

¹A sparse vector is one in which a significant fraction of the components are zero, or small enough that they can be safely ignored.

Algorithm 6 Adaptive GAMP algorithm for structured illumination

- 1: **procedure** GAMP($G_x, G_y, \hat{\mathbf{A}}, \boldsymbol{\nu}^A, \boldsymbol{\epsilon}, H_x$)
 - 2: **Center measurement matrices:** $\mu_i = \sum_j \hat{A}_{ij}, \tilde{\mathbf{A}} = \hat{\mathbf{A}} - \boldsymbol{\mu}\boldsymbol{\epsilon}^T$
 - 3: **Augment matrices:** $\hat{\underline{\mathbf{A}}} \leftarrow \begin{bmatrix} \tilde{\mathbf{A}} & \boldsymbol{\mu} \\ \boldsymbol{\epsilon}^T & -1 \end{bmatrix}, \boldsymbol{\nu}^{\underline{A}} \leftarrow \begin{bmatrix} \boldsymbol{\nu}^A & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix}$
 - 4: **Augment vectors:** $\mathbf{x} \leftarrow \begin{bmatrix} \mathbf{x} \\ x_+ \end{bmatrix}, \mathbf{z} \leftarrow \begin{bmatrix} \mathbf{z} \\ z_+ \end{bmatrix}, \mathbf{y} \leftarrow \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix}$
 - 5: **New nodes:** $G_y(\hat{p}_{N+1}, \nu_{N+1}^p, y_{N+1}) = -\frac{\hat{p}_{N+1}}{\nu_{N+1}^p}, G_x(\hat{r}_{M+1}, \nu_{M+1}^r, q) = \hat{r}_{M+1}$
 - 6: **Initialize:** $t = 0, \hat{s}_i(-1) = 0, \hat{x}_j(t)$ and $\nu_j^x(t)$ for $j \in \{1, M+1\}, i \in \{1, N+1\}$
 - 7: **repeat**
 - 8: $\hat{z}_i(t) = \sum_j \hat{A}_{ij} \hat{x}_j(t)$
 - 9: $\nu_i^z(t) = \sum_j |\hat{A}_{ij}|^2 \nu_j^x(t)$
 - 10: $\nu_i^p(t) = \nu_i^z(t) + \sum_j \nu_{ij}^A (\nu_j^x(t) + \hat{x}_j^2(t)) + \nu_i^v$ \triangleright factor-node input combination
 - 11: $\hat{p}_i(t) = \hat{z}_i(t) - \nu_i^z(t) \hat{s}_i(t-1)$
 - 12: $\hat{s}_i(t) = G_y(\hat{p}_i(t), \nu_i^p(t), y_i)$ \triangleright factor-node message approximation
 - 13: $\nu_i^s(t) = -G'_y(\hat{p}_i(t), \nu_i^p(t), y_i)$
 - 14: $\nu_j^r(t) = \left[\sum_i |\hat{A}_{ij}|^2 \nu_i^s(t) \right]^{-1}$ \triangleright variable node input combination
 - 15: $\hat{r}_j(t) = \hat{x}_j(t) + \nu_j^r(t) \sum_i \hat{A}_{ij}^* \hat{s}_i(t)$
 - 16: $\boldsymbol{\theta}^{(t)} = H_x(\hat{\mathbf{r}}(t), \boldsymbol{\nu}^r(t))$ \triangleright adaptive step
 - 17: $\hat{x}_j(t+1) = G_x(\hat{r}_j(t), \nu_j^r(t), \boldsymbol{\theta}^{(t)})$ \triangleright variable-node message approximation
 - 18: $\nu_j^x(t+1) = \nu_j^r(t) G'_x(\hat{r}_j(t), \nu_j^r(t), \boldsymbol{\theta}^{(t)})$
 - 19: **until** Estimates converge, $\|\hat{\mathbf{x}}(t+1) - \hat{\mathbf{x}}(t)\| \leq \text{tolerance}$
 - 20: **return** $(\hat{\mathbf{x}}(t+1), \boldsymbol{\nu}^x(t+1))$
 - 21: **end procedure**
-



(a) Original gray scale image.



(b) Three-level wavelet decomposition using a db4 filter.

Figure 9-1: The original image is on the left and its wavelet decomposition is on the right. The wavelet representation is falsely colored with a rotating band of black-gray-white to emphasize edges.

One of the most successful alternative representations comes from a wavelet decomposition of the image [53–55]. This decomposition is a linear transformation that creates a vector of approximation coefficients, and a sequence of descending levels of detail vectors, each of which is a correction to the immediately preceding level. Most of the pixels in the detail vectors are close to zero, and the resulting representation is sparse, as can be seen in Fig. 9-1.

Sparse vectors and random mixing are at the heart of the field of *compressed sensing* [38–40]. In the same way that a known sparse signal can be compressed, if an unknown signal is known to be sparse, then by taking a series of linear projections through a random matrix, i.e. $\mathbf{z} = \mathbf{A}\mathbf{x}$, we only have to make as many measurements as there are significant pixels in the signal. That is, if \mathbf{x} is length N but only has $M \ll N$ significant pixels, we only need to make slightly more than M measurements. This undersampling is done by finding the signal with the minimum ℓ_1 -norm that still explains the measurements. When there is no additional noise, these algorithms can *exactly* recover the target [40].

In essence, sparse signal acquisition can be thought of as encoding, or compressing, the vector \mathbf{x} into the vector \mathbf{z} . We then simply need to decompress our signal to

recover it. The initial formalization of approximate message passing was an attempt to more efficiently do this decoding. It was known that belief propagation could infer sparse signals if an appropriate sparsifying prior was used, e.g., a heavy-tail distribution such as the exponential distribution. However, belief propagation is ill suited to the fully-connected graph that results from random linear mixing, such as that of our structured-illumination setup. Hence, the original approximate message-passing algorithm [16] was created to approximate BP for compressed sensing. GAMP has generalized and extended the original approximate message-passing algorithm. Nevertheless, it remains true that employing a sparsifying prior in a wavelet basis should reduce the number of required measurements, and improve image quality. The wavelet basis does not provide for a wholly encompassing prior for our target, but it does capture important concepts about its structure that allow for more efficient decoding of the measurements.

The wavelet representation φ of the intensity reflectance pattern \mathcal{T} is defined by the matrix transformation

$$\mathcal{T} = \mathbf{U}\varphi. \tag{9.7}$$

where \mathbf{U} is a matrix that implements the wavelet reconstruction filters. In the rest of this section we develop our algorithm for a wavelet-basis prior. Besides describing the distribution, we examine the Adaptive GAMP update of parameters of this distribution, and adjust the augmented graph made for the non-centered measurement matrix.

9.2.1 Wavelet-Basis Prior

For each element of the wavelet representation we assume a Laplace prior (double-sided exponential distribution) that has the density function

$$p_\psi(\psi; \theta) = \frac{\theta}{2} e^{-\theta|\psi|}. \tag{9.8}$$

for rate parameter θ . However, because the variance decreases for each successive level of a wavelet decomposition [56], we do not define an iid prior over the wavelet coefficients, but rather an iid prior over the coefficients at each level. In particular, for \mathcal{J}_ℓ being the set of wavelet coefficient indices at level ℓ , our prior is

$$p_\varphi(\varphi_j; \theta_\ell) = \frac{\theta_\ell}{2} e^{-\theta_\ell |\varphi_j|}, \quad \text{for } j \in \mathcal{J}_\ell. \quad (9.9)$$

We therefore can update the prior in this basis once at each level, instead of globally; the update for level- ℓ only operates on the indices in \mathcal{J}_ℓ . Details about the exact computations needed to use this prior model are found in App. C.2.

In practice, updating every level was not the best approach. There is no definitive way to distinguish between the variations caused by speckle and details in the scene. Therefore, when updating the variance estimate for the more detailed levels, speckle is problematic, and the update actually degrades the image quality. This is only a problem for the detail-level coefficients, so our approach was to learn the rate parameter of the approximation level coefficients, and leave the detail levels at their prior distributions. This choice improves both the objective MSE performance, as well as the subjective quality of the images we obtain.

In addressing the adaptive learning of parameters in the wavelet basis, our model has changed, hence we need to re-derive some of the adjustments that we made in Ch. 8 to fit GAMP to our imaging problem. These are primarily associated with the measurement-matrix uncertainty, and adapting to the use of a non-centered matrix.

9.2.2 Measurement-Matrix Centering for Wavelet Basis

When evaluating the target in the wavelet basis φ , the measurement matrix is $\mathbf{A}^\varphi = \mathbf{A}\mathbf{U}$, where \mathbf{A} was the original matrix and \mathbf{U} is the wavelet transformation matrix. This new measurement-matrix might be zero mean, but the elements of the transformed vector $\mathbf{z} = \mathbf{A}^\varphi \varphi$ are still correlated because \mathbf{A} is non-centered. Yet since it is \mathbf{A} that needs centering, and the wavelet transformation is generally well behaved as far as the GAMP algorithm is concerned, our methodology is only slightly different.

Previously, we had shown that the estimate of a non-centered measurement matrix in the spatial basis could be written as $\widehat{\mathbf{A}} = \widetilde{\mathbf{A}} + \boldsymbol{\mu}\boldsymbol{\epsilon}^T$ (where $\widetilde{\mathbf{A}}$ is centered). It follows that the new wavelet-basis measurement matrix can be decomposed as

$$\widehat{\mathbf{A}}^\varphi = \widehat{\mathbf{A}}\mathbf{U} = \widetilde{\mathbf{A}}\mathbf{U} + \boldsymbol{\mu}\boldsymbol{\epsilon}^T\mathbf{U} = \widetilde{\mathbf{A}}^\varphi + \boldsymbol{\mu}\boldsymbol{\epsilon}^{\varphi T}. \quad (9.10)$$

The new variational matrix $\widetilde{\mathbf{A}}^\varphi$ handles the primary transformation of the wavelet coefficients to the measurements, while $\boldsymbol{\mu}\boldsymbol{\epsilon}^{\varphi T}$ accounts for the transfer through the average measurement matrix. Using our previous approach to centering the matrix, we use Eq. (9.10) to define a new variable for the background term, and find that it is *exactly* the same as our previous auxiliary variable, viz.

$$x_+ = \sum_k \epsilon_k^\varphi \varphi_k = \sum_k \sum_j \epsilon_j U_{jk} \varphi_k = \sum_j \epsilon_j x_j. \quad (9.11)$$

To complete the picture we need the variance of the new matrix's elements,

$$\nu_{ij}^{A^\varphi} = \sum_k \mathbf{var}[A_{ik}] U_{kj}^2 = \sum_k \nu_{ij}^A U_{kj}^2. \quad (9.12)$$

Estimation in the wavelet basis can still be done with Alg. 6, with the function call altered to be

$$(\boldsymbol{\varphi}, \boldsymbol{\nu}^\varphi) = \text{GAMP}(G_x, G_y, \widehat{\mathbf{A}}^\varphi, \boldsymbol{\nu}^{A^\varphi}, \boldsymbol{\epsilon}^\varphi, H_x). \quad (9.13)$$

9.2.3 Simulation Results

Our new algorithm—which uses a wavelet-basis prior and learns the prior parameters for the approximation level of the wavelet decomposition as the algorithm runs—greatly outperforms our previous estimators. This is shown quantitatively in Fig. 9-2, which plots the MSE performance of our new algorithm versus our previous spatial-basis GAMP algorithm, as well as the Bayesian Cramér-Rao bound (BCRB).

For qualitative results, we show the images of our test scenes for both our new

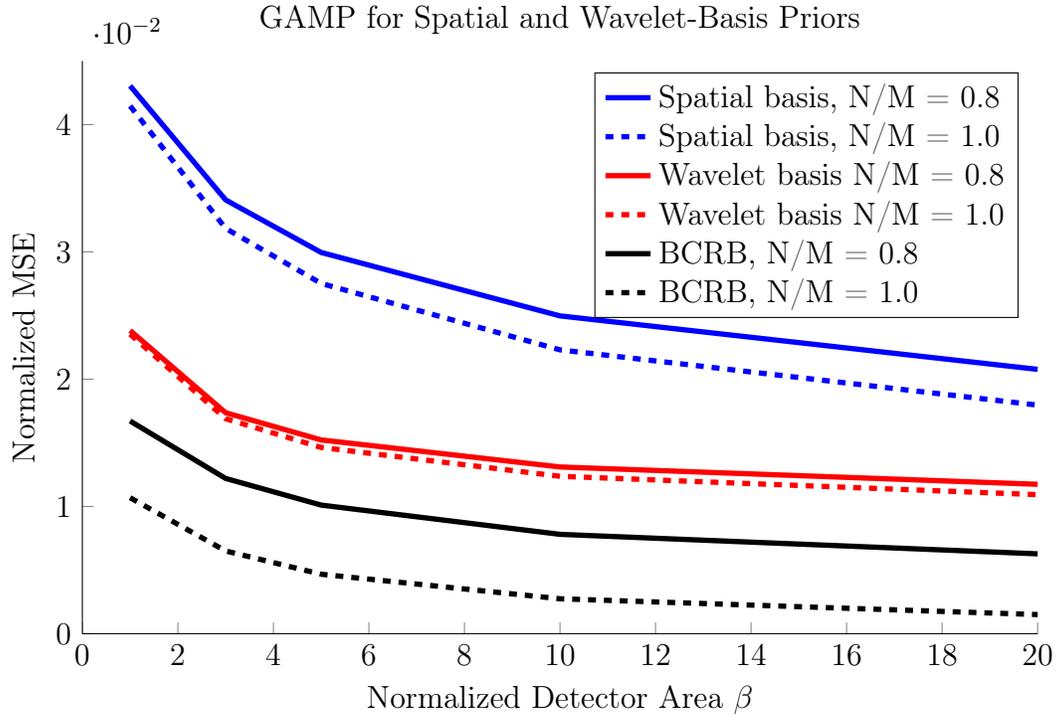


Figure 9-2: Plots versus normalized detector area of the MSE for our two different GAMP-based estimators, with the BCRB provided for comparison. The MSE is computed for each of the 100 scenes in our dataset, and then averaged.

algorithm and the previous spatial-basis GAMP algorithm. These are presented in Fig. 9-3, for $\beta = 3$ (β is the ratio of detector area to transmitter area), and in Fig. 9-4, for $\beta = 10$. In both figures the reconstructions were performed with a unity sampling ratio (the same number of measurements as pixels). To better illustrate the behavior of the new algorithm for changing bucket detector size, we also provide Fig. 9-5. This presents reconstructions of the test scenes for β equaling 1, 5, 10, and 20, with a unity sampling ratio.

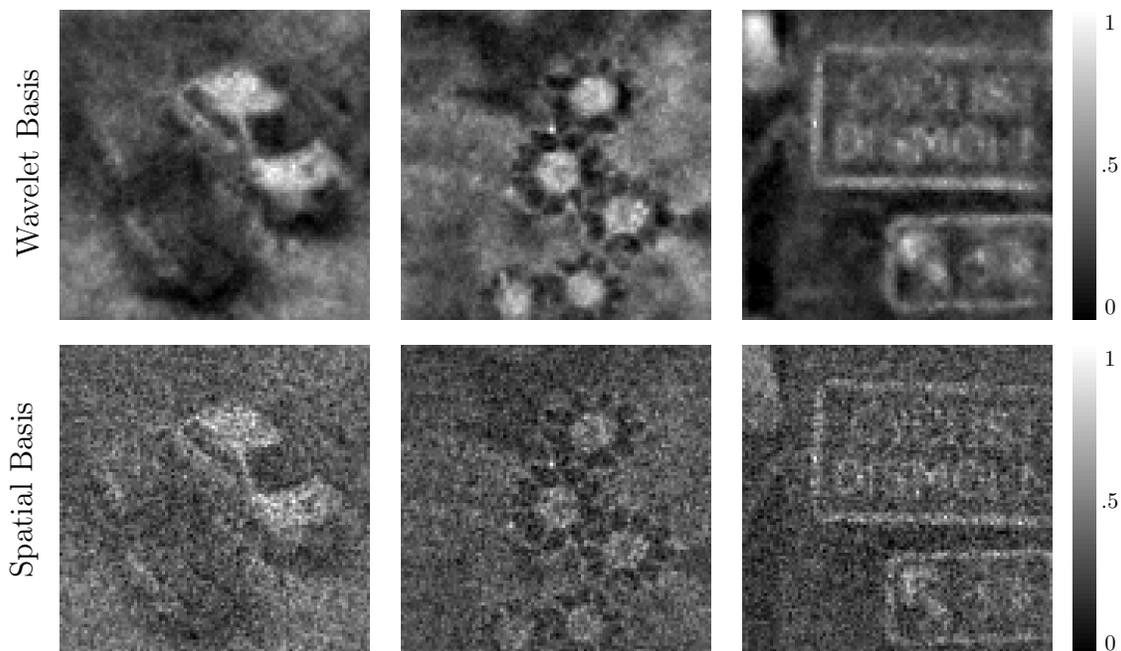


Figure 9-3: Images obtained with the GAMP-based algorithms for $\beta = 3$. The top row contains reconstructions using the adaptive wavelet-basis algorithm, while the bottom row used a spatial-basis iid Gaussian prior.

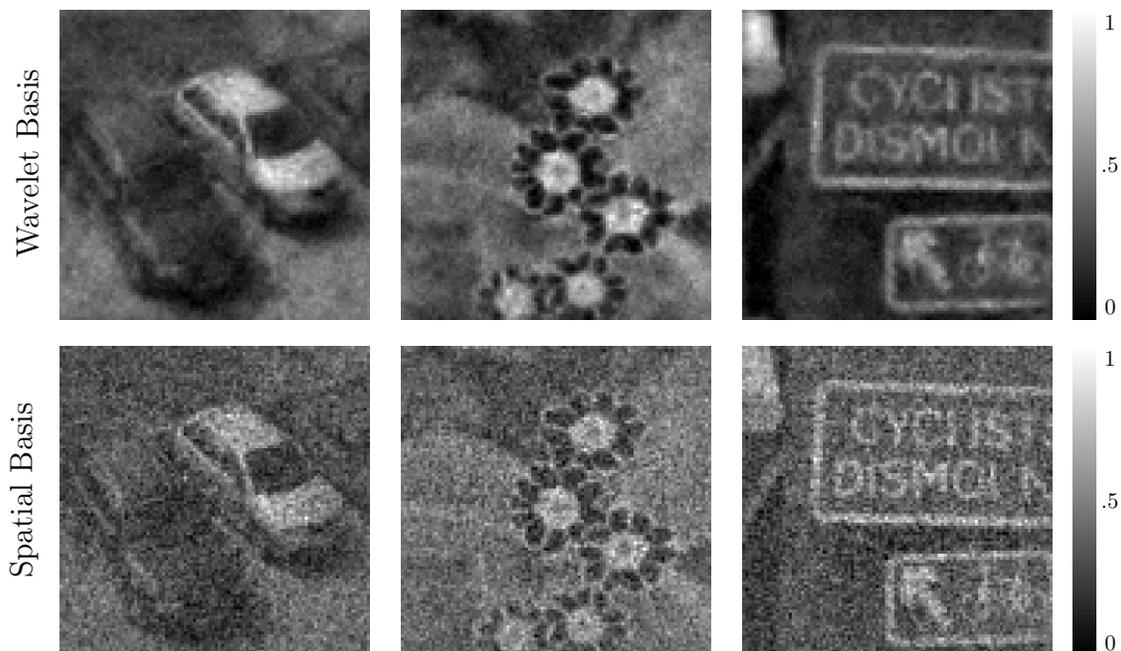


Figure 9-4: Images obtained with the GAMP-based algorithms for $\beta = 10$. The top row contains reconstructions using the adaptive wavelet-basis algorithm, while the bottom row used a spatial-basis iid Gaussian prior.

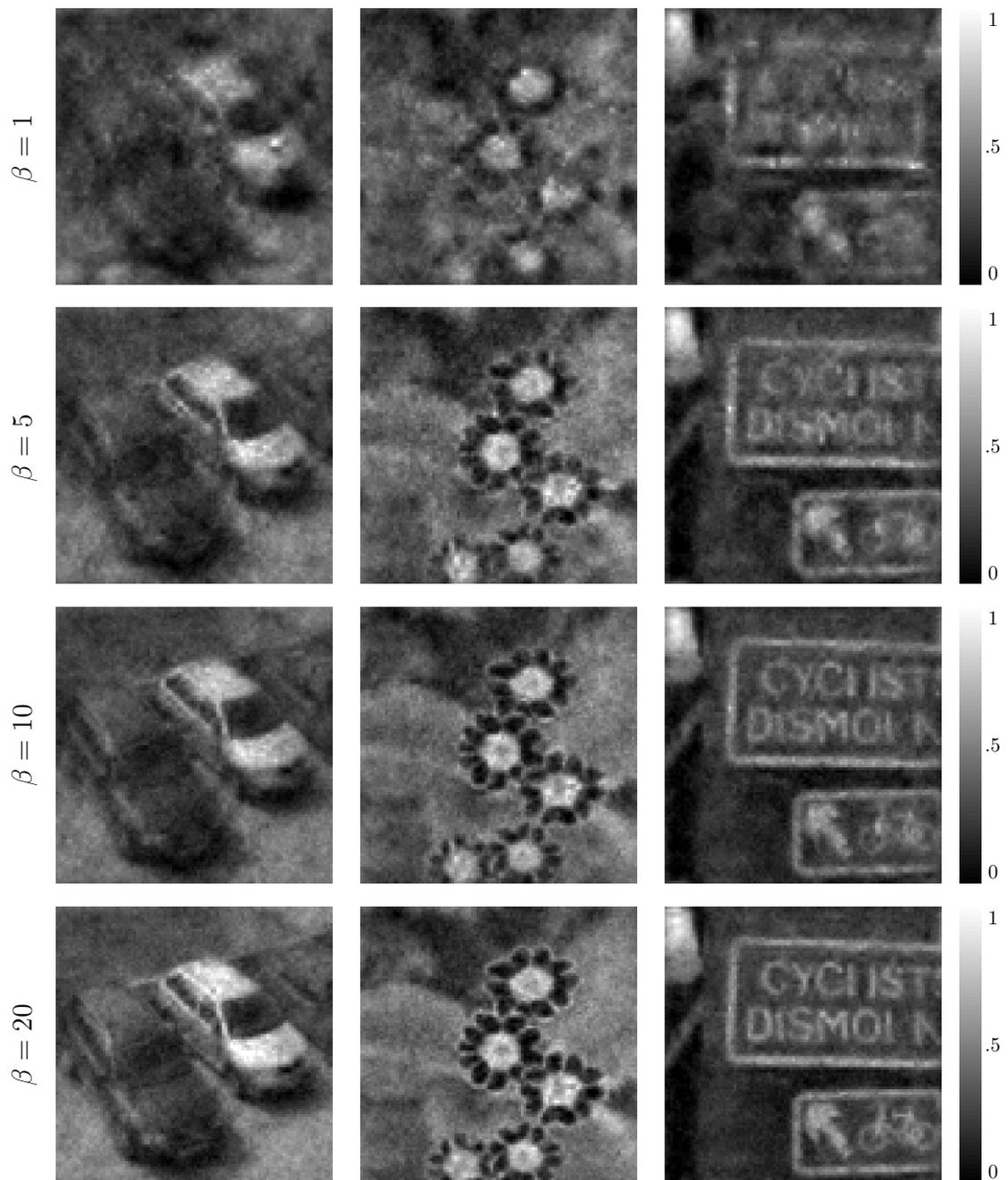


Figure 9-5: Images obtained with the adaptive, wavelet-basis GAMP algorithm. Each row used a different detector size. From top to bottom the β values are 1, 5, 10, and 20.

9.3 Exposing Speckle in the Model

We have extended our original algorithm to make it adaptive to unknown parameters and to exploit the sparsity of natural scenes, but we are still treating the speckle as noise in the measurement matrix. Speckle from optical scattering is a well studied field, and there are accurate statistical models that we can use [21]. In this section we develop this model of speckle, redevelop our imaging framework to include the speckle contributions as variables in the system, and reconcile the changes with the GAMP algorithm we have derived so far.

9.3.1 Model of Integrated Speckle

In Sec. 2.3 we developed our theory of reflectivity from rough surfaces. Building upon that foundation, we now develop a theory of *integrated speckle* [21]. Each pixel creates a speckled field pattern in the detector plane,

$$\Phi'_p(\boldsymbol{\rho}) = \int_{\mathcal{A}_p} d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_n) \vartheta(\boldsymbol{\rho}'), \quad (9.14)$$

where $\vartheta(\cdot)$ is a complex Gaussian random process resulting from the height variations on the rough surface, and $h_r(\cdot)$ is the extended Huygens-Fresnel propagation kernel. This $\Phi'_p(\boldsymbol{\rho})$ expression follows from the $\Phi_p(\boldsymbol{\rho}, n)$ defined in Eq. (4.14) when the on-target field pattern is approximately uniform over each target pixel so that $\Phi_p(\boldsymbol{\rho}, n) = \Phi'_p(\boldsymbol{\rho}) \xi_t(\boldsymbol{\rho}_p, n)$, where $\boldsymbol{\rho}_p$ is the center of \mathcal{A}_p . At every point $\boldsymbol{\rho}$, $\Phi'_p(\boldsymbol{\rho})$ is the weighted integration of the surface fluctuation $\vartheta(\cdot)$ over many of its coherence lengths. In the absence of turbulence, $\Phi'_p(\boldsymbol{\rho})$ is a Gaussian process because it is a linear transformation of $\vartheta(\boldsymbol{\rho}')$. CLT considerations imply that $\Phi'_p(\boldsymbol{\rho})$ still will be Gaussian in the presence of turbulence because the turbulence coherence length is much longer than that of the rough surface.

In the Cramér-Rao bound (CRB) derivation from Ch. 5, we briefly introduced a normalized integrated speckle intensity in Eq. (5.24) that, using Eq. (9.14), is given

by

$$\mathcal{S}_p = \frac{1}{\mathcal{F}A_p} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} |\Phi'_p(\boldsymbol{\rho})|^2. \quad (9.15)$$

The exact distribution of \mathcal{S}_p has been studied, but is not computationally tractable [21]. Instead, a pill-box approximation is frequently used. We divide the detector's surface into M identical rectangles, or pill-boxes, whose width is the coherence length of the speckle field $\Phi'_p(\boldsymbol{\rho})$. We approximate the integration over pill box \mathcal{A}_m (centered at $\boldsymbol{\rho}_m$) as

$$\int_{\mathcal{A}_m} d\boldsymbol{\rho} |\Phi'_p(\boldsymbol{\rho})|^2 \approx \frac{A_b}{M} |\Phi'_p(\boldsymbol{\rho}_m)|^2, \quad (9.16)$$

so that the integrated speckle term becomes

$$\mathcal{S}_p \approx \frac{A_b}{M\mathcal{F}A_p} \sum_{m=1}^M |\Phi'_p(\boldsymbol{\rho}_m)|^2. \quad (9.17)$$

The $|\Phi'_p(\boldsymbol{\rho}_m)|^2$ are then taken to be iid exponential random variables. The sum of iid exponential random variables is a gamma random variable, so the probability density function of \mathcal{S}_p is

$$p_{\mathcal{S}}(\mathcal{S}_p) = \begin{cases} \frac{\mathcal{S}_p^{k-1} e^{-\mathcal{S}_p/\theta_{\mathcal{S}}}}{\theta_{\mathcal{S}}^k \Gamma(k)}, & \mathcal{S}_p > 0, k > 0, \theta_{\mathcal{S}} > 0 \\ 0 & \text{otherwise} \end{cases}, \quad (9.18)$$

for shape and rate parameters k and $\theta_{\mathcal{S}}$, with $\Gamma(\cdot)$ being the standard gamma function. For our coherence-area discretization model of the detector, these parameters are $k = M$, and $\theta_{\mathcal{S}} = \frac{A_b}{M\mathcal{F}A_p} \langle |\Phi'_p(\boldsymbol{\rho}_m)|^2 \rangle$. However, to make the gamma distribution derived by the preceding discretization as accurate as possible, we will match the gamma distribution's mean and variance,

$$\langle \mathcal{S}_p \rangle = k\theta_{\mathcal{S}} \quad \mathbf{var}[\mathcal{S}_p] = k\theta_{\mathcal{S}}^2, \quad (9.19)$$

to the values obtained from the exact evaluations of the true moments of \mathcal{S}_p from its original definition in Eq. (9.15).

Mean and Variance Calculation

From our normalization, the speckle coefficient has a unity mean, which we quickly see from

$$\begin{aligned}
\langle \mathcal{S}_p \rangle &= \frac{1}{\mathcal{F}A_p} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \langle |\Phi_p(\boldsymbol{\rho})|^2 \rangle \\
&= \frac{1}{\mathcal{F}A_p} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \int_{\mathcal{A}_p} d\boldsymbol{\rho}' \int_{\mathcal{A}_p} d\boldsymbol{\rho}'' \langle h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_n) h_r^*(\boldsymbol{\rho}, \boldsymbol{\rho}'', \tau_n) \rangle \langle \vartheta(\boldsymbol{\rho}') \vartheta^*(\boldsymbol{\rho}'') \rangle \\
&= \frac{\pi \varrho_T^2}{\mathcal{F}A_p} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}' \frac{1}{L^2 \lambda_0^2} \\
&= 1.
\end{aligned} \tag{9.20}$$

The variance calculation is more complicated. Since we know the mean we only have to compute the second moment,

$$\langle \mathcal{S}_p^2 \rangle = \frac{1}{\mathcal{F}^2 A_p^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \langle |\Phi_p'(\boldsymbol{\rho}_1)|^2 |\Phi_p'(\boldsymbol{\rho}_2)|^2 \rangle. \tag{9.21}$$

Similar to the ghost imaging evaluations, the steps can become quite tedious when evaluating the fourth-order moments over the turbulence and surface roughness, but the approach is straightforward. We find that $\langle \mathcal{S}_p^2 \rangle = e^{4\sigma_r^2}(1 + \Gamma_0)$, where Γ_0 is given by Eq. (4.26). In fact, because the variance becomes

$$\text{var}[\mathcal{S}_p] = e^{4\sigma_r^2} \Gamma_0 + (e^{4\sigma_r^2} - 1), \tag{9.22}$$

we see that Γ_0 is the turbulence-free variance of the speckle. Combining Eqs. (9.20) and (9.22) with (9.19) yields

$$k = \frac{1}{e^{4\sigma_r^2} \Gamma_0 + (e^{4\sigma_r^2} - 1)}, \quad \theta = \frac{e^{4\sigma_r^2} \Gamma_0 + (e^{4\sigma_r^2} - 1)}{1}. \tag{9.23}$$

9.3.2 Connection of Speckle to Measurements

We have defined the speckle coefficient for each pixel, but now we need to make explicit its connection to the measurements \mathbf{y} . We start by considering the signal term from Eq. (4.31),

$$\mathbf{z}_n = \sum_p \mathcal{I}(p, p, n) \mathcal{T}_p, \quad (9.24)$$

where from Eq. (4.16) we have

$$\mathcal{I}(p, p, n) = \mathbb{N}_s \int_{\mathcal{A}_b^2} d\boldsymbol{\rho} |\Phi_{np}(\boldsymbol{\rho})|^2, \quad (9.25)$$

with \mathbb{N}_s being is the number of transmitted photons per pulse, and

$$\Phi_{np}(\boldsymbol{\rho}) = \int_{\mathcal{A}_p} d\boldsymbol{\rho}' h_r(\boldsymbol{\rho}, \boldsymbol{\rho}', \tau_n) \vartheta(\boldsymbol{\rho}') \xi_t(\boldsymbol{\rho}', n), \quad (9.26)$$

is the field at the detector resulting from the reflection of the field pattern $\xi_t(\boldsymbol{\rho}', n)$ from pixel p for pulse n . Imposing the condition that $\xi_t(\boldsymbol{\rho}', n)$ is constant across the pixel, $\mathcal{I}(p, p, n)$ simplifies to

$$\mathcal{I}(p, p, n) = \mathbb{N}_s \mathcal{F} A_p |\xi_t(\boldsymbol{\rho}_p, n)|^2 \mathcal{S}_p, \quad (9.27)$$

where \mathcal{F} is the Fresnel number product from Eq.(4.5), and A_p is the area of pixel \mathcal{A}_p . Our signal terms are now linear transformations of the target multiplied by the speckle (which we will refer to as the *speckled target*), viz.,

$$\mathbf{z}_n = \mathbb{N}_s \mathcal{F} A_p \sum_p |\xi_t(\boldsymbol{\rho}_p, n)|^2 \mathcal{T}_p \mathcal{S}_p. \quad (9.28)$$

9.3.3 Perturbative Expansion

Unfortunately, the field profile is not constant across each pixel, but we still want to directly expose the speckle coefficient. That is, we want to decouple the problem

into an intensity matrix Ξ and speckle-intensity coefficients \mathcal{S} that together with the target are responsible for the mean signal, with all other pulse-dependent fluctuations being treated as noise. We take a perturbative approach, redefining our signal as the transfer of the speckled target through this new matrix, viz.,

$$\mathbf{z}_n = \sum_p \Xi_{np} \mathcal{S}_p \mathcal{T}_p, \quad (9.29)$$

while all other fluctuations contribute to a new zero-mean noise term

$$\mathbf{w}_n = \sum_p (\mathcal{I}(p, p, n) - \Xi_{np} \mathcal{S}_p) \mathcal{T}_p. \quad (9.30)$$

To maintain consistency with our previous framework, we want the transformations of the means to remain the same, which requires that $\langle \mathcal{I}(p, p, n) \rangle = \langle \Xi_p \rangle \langle \mathcal{S}_p \rangle$. This condition is satisfied by defining the intensity matrix as a scaled version of the total number of photons impinging on each pixel,

$$\Xi_{np} = \mathcal{F} P_s \int_{\mathcal{A}_p} d\boldsymbol{\rho} |\xi_t(\boldsymbol{\rho}, n)|^2. \quad (9.31)$$

Equations (9.29)–(9.31) lead to the number of received photons being expressed as the sum of three random variables, $\mathcal{N}_n = \mathbf{z}_n + \mathbf{v}_n + \mathbf{w}_n$, where \mathbf{v}_n is the inter-pixel noise term from Eq. (4.32).

We again use a linear estimator to approximate the statistics of \mathbf{w}_n , conditioned on knowledge of the reference, and find that it is also conditionally a zero-mean Gaussian random variable with variance

$$\nu_n^w = e^{4(\sigma_s^2 + \sigma_r^2)} \mu_n^2 (\Gamma_1 + \Gamma_0(2 + \kappa_1) - 2(\Gamma_2 + \Gamma_3)) \sum_p \epsilon_p^2 (\bar{\mathcal{T}}_p^2 + \nu_p^{\mathcal{T}}), \quad (9.32)$$

where $\bar{\mathcal{T}}_p$ and $\nu_p^{\mathcal{T}}$ are the prior mean and variance of the target, Γ_0 and Γ_1 are defined

in Eqs. (4.26) and (4.27), and the remaining terms are

$$\Gamma_2 = \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} e^{-\frac{|\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2|^2}{2\rho_L^2}} \quad (9.33)$$

$$\Gamma_3 = \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^3} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_2 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_3 \\ \times e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} e^{-\frac{|\boldsymbol{\rho}'_3 - \boldsymbol{\rho}'_2|^2 + |\boldsymbol{\rho}'_3 - \boldsymbol{\rho}'_1|^2}{2\rho_L^2}}. \quad (9.34)$$

This additional noise term implies a slight modification of GAMP: in Alg. 6, whenever the inter-pixel noise term ν_i^z appears we add the new noise term ν_i^w .

Our algorithmic development in the next section rests on the following observations. Were the speckle known then, by Eq. (9.29), \mathbf{z} can be viewed as a linear transformation of the target \mathcal{T} through a matrix with elements $A_{np} = \Xi_{np} \mathcal{S}_p$, and we can use GAMP to estimate \mathcal{T} . This procedure easily extends to estimating the target in the wavelet basis. On the other hand, were the target known, then \mathbf{z} can be viewed as a linear transformation of the speckle coefficients through a matrix with elements $A_{np} = \Xi_{np} \mathcal{T}_p$, and we can use GAMP to estimate \mathcal{S} . To perform these calculations we just need the mean and variance of the measurement-matrix elements.

Both of these situations require the mean and variance of the elements of Ξ . We again use a linear estimator to approximate these terms, obtaining

$$\widehat{\Xi}_{np}^{(L)} = \left[I_{np} \frac{\kappa_\alpha}{\kappa_1} + \bar{I}_{np} \left(1 - \frac{\kappa_\alpha}{\kappa_1} \right) \right] \quad (9.35)$$

$$\mathbf{var}^{(L)}[\Xi_{np}|I_{np}] = \bar{I}_{np}^2 \left[e^{4\sigma_s^2} (1 + \kappa_1) - \left(1 + \frac{\kappa_\alpha^2}{\kappa_1} \right) \right]. \quad (9.36)$$

The reader will note that the estimate of the linear transform matrix is exactly what was found for the general case in Eq. (8.50) when the speckle was treated as noise. This is because we normalized the speckle, so that $\langle \mathcal{I}(p, p, n) | I_{np} \rangle = \langle \Xi_{np} | I_{np} \rangle$. However, because the speckle is no longer being treated as noise, the variance in Eq. (9.36) is significantly reduced compared to the variance for $\mathcal{I}(p, p, n)$ computed in Eq. (4.28).

9.4 Joint Target and Speckle Estimation

We have extended our model to include better descriptions of the target and speckle. We now seek a way to use this additional information to improve our estimation of the target. We have investigated several methods of approximating belief propagation to estimate both the target wavelet coefficients and the speckle coefficients. The most accurate and robust method we found was an iterative approach based on the E-M algorithm.

We start with the E-M algorithm for learning the speckle \mathcal{S} from the measurements \mathbf{y} ,

$$\widehat{\mathcal{S}}^{(\tau+1)} = \operatorname{argmax}_{\mathcal{S}} \left\langle \log p_{\mathbf{y}, \mathbf{x}, \mathcal{S}}(\mathbf{y}, \mathbf{x}, \mathcal{S}) \middle| \mathbf{y}, \widehat{\mathcal{S}}^{(\tau)} \right\rangle_{\mathbf{x}} \quad (9.37)$$

where we choose the intermediary vector to be the wavelet coefficients of the target, so $\mathbf{x} \equiv \boldsymbol{\varphi}$. Since we use t for iterations within the GAMP algorithm, we are using τ for the E-M iterations. The expectation in Eq. (9.37) is taken with respect to the posterior distribution for \mathbf{x} , computed with the speckle estimate for iteration τ . We denote this new distribution

$$p_{\mathbf{x}}^{(\tau)}(\mathbf{x}) \equiv p_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y}; \mathcal{S}^{(\tau)}). \quad (9.38)$$

The reader will note that this distribution is the result of running the GAMP algorithm to estimate \mathbf{x} , conditioned on knowledge of $\mathcal{S}^{(\tau)}$. If we can compute the maximization in Eq. (9.37) to estimate $\mathcal{S}^{(\tau+1)}$, we can iterate between these two steps to continually compute more accurate estimates of the posterior distribution, $p_{\mathbf{x}}^{\tau}(\mathbf{x})$, using continually improved estimates of the speckle, $\mathcal{S}^{(\tau)}$. However, the maximization in Eq. (9.37) is difficult to compute. In the next section we derive an alternate update step, and show that it is asymptotically equivalent to the E-M update. We therefore can be assured that the algorithm will converge to a local maximum of the distribution, with each step providing a better estimate than its predecessor.

9.4.1 Maximum Likelihood Approximation to Expectation Maximization Update

Our approach is similar to that taken in the Adaptive GAMP algorithm [51]; instead of using an E-M update to learn \mathcal{S} , we directly maximize an updated likelihood function for \mathcal{S} given \mathbf{y} ,

$$\widehat{\mathcal{S}}^{(\tau+1)} = \underset{\mathcal{S}}{\operatorname{argmax}} \log p_{\mathbf{y},\mathcal{S}}^{(\tau)}(\mathbf{y}, \mathcal{S}). \quad (9.39)$$

The likelihood $p_{\mathbf{y},\mathcal{S}}^{(\tau)}(\mathbf{y}, \mathcal{S})$ for each iteration τ is computed by eliminating the wavelet coefficients from their joint distribution,

$$p_{\mathbf{y},\mathcal{S}}^{(\tau)}(\mathbf{y}, \mathcal{S}) = \int d\mathbf{x} p_{\mathbf{y},\mathbf{x},\mathcal{S}}^{(\tau)}(\mathbf{y}, \mathbf{x}, \mathcal{S}). \quad (9.40)$$

This joint distribution is computed by replacing the prior distribution for \mathbf{x} with the updated distribution computed by GAMP in Eq. (9.38), i.e.,

$$p_{\mathbf{y},\mathbf{x},\mathcal{S}}^{(\tau)}(\mathbf{y}, \mathbf{x}, \mathcal{S}) \equiv p_{\mathbf{y},\mathcal{S}|\mathbf{x}}(\mathbf{y}, \mathcal{S}|\mathbf{x}) p_{\mathbf{x}}^{(\tau)}(\mathbf{x}). \quad (9.41)$$

We have taken this approach because the maximization in Eq. (9.39) can be computed using the MAP variant of GAMP. We specify this computation in Sec. 9.4.2, but first we show that the maximum likelihood (ML) update in Eq. (9.39) produces estimates that are equivalent to those obtained from the E-M update in Eq. (9.37). This equivalence occurs asymptotically when the number of pixels, M , and number of measurements, N , both increase without bound, but at a fixed sampling ratio, N/M , that is a function of the target's sparsity.

To show that our ML approach yields a speckle estimate $\widehat{\mathcal{S}}^{(\tau)}$ equivalent to that of the E-M algorithm, we look at the difference in their objective functions,

$$V(\mathcal{S}|\mathcal{S}^{(\tau)}) = \left\langle \log p_{\mathbf{y},\mathbf{x},\mathcal{S}}(\mathbf{y}, \mathbf{x}, \mathcal{S}) \Big|_{\mathbf{x}} \Big|_{\mathbf{y}, \mathcal{S}^{(\tau)}} \right\rangle - \log p_{\mathbf{y},\mathcal{S}}^{(\tau)}(\mathbf{y}, \mathcal{S}). \quad (9.42)$$

We expand the average as an integral over the distribution $p_{\mathbf{x}}^{(\tau)}(\mathbf{x})$, expand the log

distributions, and discard additive terms that will not affect the maximization². Doing so, we obtain

$$V(\mathcal{S}|\mathcal{S}^{(t)}) = \int d\mathbf{x} p_{\mathbf{x}}^{(t)}(\mathbf{x}) \log p_{\mathbf{y}|\mathcal{S},\mathbf{x}}(\mathbf{y}|\mathcal{S},\mathbf{x}) - \log p_{\mathbf{y}|\mathcal{S}}^{(t)}(\mathbf{y}|\mathcal{S}), \quad (9.43)$$

where $p_{\mathbf{y}|\mathcal{S}}^{(t)}(\mathbf{y}|\mathcal{S}) = \int d\mathbf{x} p_{\mathbf{y}|\mathbf{x},\mathcal{S}}(\mathbf{y}|\mathbf{x},\mathcal{S}) p_{\mathbf{x}}^{(t)}(\mathbf{x})$. From Jensen's inequality we see that

$$\int d\mathbf{x} p_{\mathbf{x}}^{(t)}(\mathbf{x}) \log p_{\mathbf{y}|\mathcal{S},\mathbf{x}}(\mathbf{y}|\mathcal{S},\mathbf{x}) \leq \log p_{\mathbf{y}|\mathcal{S}}^{(t)}(\mathbf{y}|\mathcal{S}), \quad (9.44)$$

and therefore the ML maximization function is an upper bound on the E-M function, and so their difference satisfies $V(\mathcal{S}|\mathcal{S}^{(t)}) \leq 0$.

This difference being non-positive is a problem, because the E-M algorithm is predicated on maximizing a lower bound of the true log-likelihood. We are now maximizing the upper bound on that lower bound that, in general, need not yield an update which has the E-M algorithm's performance guarantee. However, we can show that the gap separating these two functions becomes insignificant as M (and N , since the sampling ratio is fixed) increases, implying that the estimates produced by these two methods converge.

We start by presenting the asymptotic equipartition property (AEP). For a vector \mathbf{y} of N iid random variables y_n , the normalized logarithm of their joint distribution converges to the entropy of the marginal distribution, $H(\mathbf{y}) = -\langle \log p_{\mathbf{y}}(\mathbf{y}) \rangle$, as

$$-\lim_{N \rightarrow \infty} \frac{1}{N} \log p_{\mathbf{y}}(\mathbf{y}) = -\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N \log p_{\mathbf{y}}(y_n) = H(\mathbf{y}). \quad (9.45)$$

The measurements \mathbf{y} approach an iid distribution as $M \rightarrow \infty$, so we have

$$\log p_{\mathbf{y}|\mathcal{S},\mathbf{x}}(\mathbf{y}|\mathcal{S},\mathbf{x}) \rightarrow -NH(\mathbf{y}|\mathcal{S},\mathbf{x}) \quad (9.46)$$

$$\log p_{\mathbf{y}|\mathcal{S}}^{(\tau)}(\mathbf{y}|\mathcal{S}) \rightarrow -NH(\mathbf{y}|\mathcal{S}). \quad (9.47)$$

In both cases the measurement y_i can be recast as probabilistic function of a scalar

²Only components of the objective functions that are functions of \mathcal{S} will affect the maximization

random variable; for $p_{\mathbf{y}|\mathcal{S},\mathbf{x}}(y_i|\mathcal{S},\mathbf{x})$ this is z_i , while for $p_{\mathbf{y}|\mathcal{S}}^{(\tau)}(\mathbf{y}|\mathcal{S})$ we denote it $z_i^{(\tau)}$. As the summation of M independent random variables, these are both Gaussian random variables with signal-to-noise ratios that grow linearly with M . Therefore, as $M \rightarrow \infty$ the entropies in Eqs. (9.46) and (9.47) become

$$\mathrm{H}(\mathbf{y}|\mathcal{S},\mathbf{x}) \rightarrow \mathrm{H}(\mathbf{y}|\bar{z}) \quad (9.48)$$

$$\mathrm{H}(\mathbf{y}|\mathcal{S}) \rightarrow \mathrm{H}(\mathbf{y}|\bar{z}^{(\tau)}), \quad (9.49)$$

where $\bar{z}_i = \sum_j \hat{A}_{ij} x_j \mathcal{S}_j \rightarrow \bar{z}$ and $\bar{z}_i^{(\tau)} = \sum_j \hat{A}_{ij} \hat{x}_j^{(\tau)} \mathcal{S}_j \rightarrow \bar{z}^{(\tau)}$ for all i . Incidentally, the expectation of \bar{z} over the distribution $p_{\mathbf{x}}^{(t)}(\mathbf{x})$ is $\langle \bar{z} \rangle_{\mathbf{x}} = \bar{z}^{(\tau)}$. This means that, since \bar{z}_i is still the sum of M random variables, its signal to noise ratio is

$$\frac{\langle \bar{z} \rangle^2}{\mathrm{var}[\bar{z}]} = \frac{(\bar{z}^{(\tau)})^2}{\mathrm{var}[\bar{z}]} \propto M \gg 1. \quad (9.50)$$

The entropies in Eqs. (9.48) and (9.49) are the entropies of the Poisson detection process, which for a large value of the rate parameter z is

$$\mathrm{H}(\mathbf{y}|z) \approx \frac{1}{2} \log(2\pi z). \quad (9.51)$$

With these entropies in hand we find that the difference function obeys

$$V(\mathcal{S}|\mathcal{S}^{(t)}) = -N \int d\mathbf{x} p_{\mathbf{x}}^{(t)}(\mathbf{x}) \mathrm{H}(\mathbf{y}|\mathcal{S},\mathbf{x}) + \mathrm{H}(\mathbf{y}|\mathcal{S}) \quad (9.52)$$

$$= -N \int d\mathbf{x} p_{\mathbf{x}}^{(t)}(\mathbf{x}) [\mathrm{H}(\mathbf{y}|\bar{z}) - \mathrm{H}(\mathbf{y}|\bar{z}^{(\tau)})] \quad (9.53)$$

$$= -\frac{N}{2} \int d\mathbf{x} p_{\mathbf{x}}^{(t)}(\mathbf{x}) \log \left(\frac{\bar{z}}{\bar{z}^{(\tau)}} \right). \quad (9.54)$$

The random variable $\bar{z}/\bar{z}^{(\tau)}$ has a high signal-to-noise ratio and mean of one, so

employing the logarithm's small number expansion³ we achieve

$$V(\mathcal{S}|\mathcal{S}^{(t)}) \approx -N \int d\mathbf{x} p_{\mathbf{x}}^{(t)}(\mathbf{x}) \left[\left(\frac{\bar{z}}{\bar{z}^{(\tau)}} - 1 \right) - \frac{1}{2} \left(\frac{\bar{z}}{\bar{z}^{(\tau)}} - 1 \right)^2 \right] \quad (9.55)$$

$$= \frac{N \mathbf{var}[\bar{z}]}{4 (\bar{z}^{(\tau)})^2} \quad (9.56)$$

$$\propto \frac{N}{M}. \quad (9.57)$$

Because we are holding the sampling ratio N/M fixed as we increase N and M , $V(\mathcal{S}|\mathcal{S}^{(t)})$ converges to a constant. From our previous discussion we know that the log-likelihood grows linearly with the number of measurements, so despite $V(\mathcal{S}|\mathcal{S}^{(t)})$ being a function of \mathcal{S} , its value becomes insignificant in performing the optimization for $M \gg 1$ and a constant N/M . Thus, our ML estimate will approach the estimate produced by the E-M algorithm.

9.4.2 Speckle Estimation

Having shown that our ML optimization strategy will provide useful estimates for the speckle, we need to show that the maximization in Eq. (9.39) can be accomplished using GAMP. Our first step is to factor the joint distribution of \mathbf{y} and \mathcal{S} ,

$$p_{\mathbf{y}, \mathcal{S}}^{(\tau)}(\mathbf{y}, \mathcal{S}) = \prod_i p_{y_i | \mathcal{S}}^{(\tau)}(y_i | \mathcal{S}) \prod_j p_{\mathcal{S}}(\mathcal{S}_j), \quad (9.58)$$

as required by GAMP. The measurement y_i is a scalar function of the transform variable $z_i^{(\tau)}$ so that

$$p_{y_i | \mathcal{S}}^{(\tau)}(y_i | \mathcal{S}) = \int dz_i p_{z_i | \mathcal{S}}^{(\tau)}(z_i^{(\tau)} | \mathcal{S}) p_{y|z}(y_i | z_i^{(\tau)}). \quad (9.59)$$

We have already noted that $z_i^{(\tau)}$ Gaussian distributed, with mean

$$\langle z_i^{(\tau)} | \mathcal{S} \rangle = \sum_j \hat{A}_{ij} \hat{x}_j^{(\tau)} \mathcal{S}_j. \quad (9.60)$$

³For $|z - 1| \ll 1$ we can approximate $\log z$ by $(z - 1) - \frac{1}{2}(z - 1)^2$.

Its variance can be found from Eq. (4.35) by iterating over the measurement-matrix and target, viz.

$$\begin{aligned}
\mathbf{var}[z_i^{(\tau)}|\mathcal{S}] &= \left\langle \mathbf{var}[z_i^{(\tau)}|\mathbf{A}, \mathcal{S}] \right\rangle_{\mathbf{A}} + \mathbf{var}_{\mathbf{A}} \left[\left\langle z_i^{(\tau)}|\mathbf{A}, \mathcal{S} \right\rangle \right] \\
&= \sum_j \langle A_{ij}^2 \rangle \nu_j^{x(\tau)} \mathcal{S}_j^2 + \sum_j \nu_{ij}^A (\hat{x}_j^{(\tau)})^2 \mathcal{S}_j^2 \\
&= \sum_j \mathcal{S}_j^2 \nu_{ij}^A (\nu_j^{x(\tau)} + (\hat{x}_j^{(\tau)})^2) + \sum_j \mathcal{S}_j^2 \hat{A}_{ij}^2 \nu_j^{x(\tau)}. \tag{9.61}
\end{aligned}$$

In essence, we have defined a new measurement matrix for the speckle using the current estimate $(\hat{\mathbf{x}}^{(\tau)}, \boldsymbol{\nu}^{x(\tau)})$ of the target. The mean and variance of this matrix are then

$$\hat{A}_{ij}^{\mathcal{S}} = \hat{A}_{ij} \hat{x}_j^{(\tau)}, \quad \nu_{ij}^{A\mathcal{S}} = \sum_j \nu_{ij}^A (\nu_j^{x(\tau)} + (\hat{x}_j^{(\tau)})^2) + \sum_j \hat{A}_{ij}^2 \nu_j^{x(\tau)}, \tag{9.62}$$

while the new envelope vector is $\boldsymbol{\epsilon}_j^{\mathcal{S}} = \epsilon_j \hat{x}_j^{(\tau)} / \sum_j \epsilon_j \hat{x}_j^{(\tau)}$. The measurement matrix has a significant background, so we will have to use our adjustment for non-centered matrices derived in Sec. 8.2.4. Then, for G_S^{MAP} and G_y^{MAP} being the MAP input and output estimators from Table 8.1, the new speckle update is found from our modified GAMP algorithm, Alg. 6, as

$$(\hat{\mathcal{S}}^{(\tau+1)}, \cdot) = \text{GAMP}(G_S^{\text{MAP}}, G_y^{\text{MAP}}, \hat{\mathbf{A}}^{\mathcal{S}}, \boldsymbol{\nu}^{A\mathcal{S}}, \boldsymbol{\epsilon}^{\mathcal{S}}, \cdot). \tag{9.63}$$

9.4.3 Final Algorithm

At this point we have shown how to perform the alternating steps of our estimation algorithm. All that remains to combine them. Our methodology is presented in Alg. 7. The imaging results obtained with this algorithm and their comparison to our previous work appears in the next section.

Algorithm 7 Joint estimation of target and speckle

Start with sensing matrix Ξ (with expected value $\hat{\Xi}$ and variance ν^Ξ), wavelet transform matrix \mathbf{U} , wavelet MMSE estimation function G_φ , speckle MAP estimation function G_S^{MAP} , measurement MMSE estimation function G_y , measurement MAP estimation function G_y^{MAP} , and wavelet adaptation function H_φ .

- 1: Initialize estimate of speckle $\hat{\mathcal{S}}_j^{(0)}$.
- 2: **repeat**
- 3: Update mean and variance of matrix, adjust envelope for target estimation,

$$\begin{aligned}\hat{A}_{ij}^\varphi &= \sum_k \hat{\Xi}_{ik} \hat{\mathcal{S}}_k^{(\tau)} U_{ki} \\ \nu_{ij}^{A\varphi} &= \sum_k \nu_{ik}^\Xi (\hat{\mathcal{S}}_k^{(\tau)})^2 U_{ki}^2 \\ \epsilon_j^\varphi &= \frac{\epsilon_j \hat{\mathcal{S}}_j^{(\tau)}}{\sum_\ell \epsilon_\ell \hat{\mathcal{S}}_\ell^{(\tau)}}.\end{aligned}$$

- 4: Find MMSE estimate of target,

$$(\hat{\varphi}^{(\tau)}, \nu^{\varphi^{(\tau)}}) = \text{GAMP}(G_\varphi, G_y, \hat{\mathbf{A}}^\varphi, \nu^{A\varphi}, \epsilon^\varphi, H_\varphi).$$

- 5: Compute mean and variance in spatial basis,

$$\hat{x}_j^{(\tau)} = \sum_k U_{jk} \hat{\varphi}_k^{(\tau)} \tag{9.64}$$

$$\nu_j^{x^{(\tau)}} = \sum_k U_{jk}^2 \nu_k^{\varphi^{(\tau)}}. \tag{9.65}$$

- 6: Update mean and variance of matrix, adjust envelope for speckle estimation,

$$\begin{aligned}\hat{A}_{ij}^{\mathcal{S}} &= \hat{\Xi}_{ij} \hat{x}_j^{(\tau)} \\ \nu_{ij}^{A\mathcal{S}} &= \sum_j \nu_{ij}^\Xi (\nu_j^{x^{(\tau)}} + (\hat{x}_j^{(\tau)})^2) + \sum_j \hat{\Xi}_{ij}^2 \nu_j^{x^{(\tau)}} \\ \epsilon_j^{\mathcal{S}} &= \frac{\epsilon_j \hat{x}_j^{(\tau)}}{\sum_\ell \epsilon_\ell \hat{x}_\ell^{(\tau)}}.\end{aligned}$$

- 7: Find MAP estimate of speckle

$$(\hat{\mathcal{S}}^{(\tau+1)}, \cdot) = \text{GAMP}(G_S, G_y, \hat{\mathbf{A}}^{\mathcal{S}}, \nu^{A\mathcal{S}}, \epsilon^{\mathcal{S}}, \cdot).$$

- 8: **until** estimates converge, $\|\hat{\mathbf{x}}^{(\tau)} - \hat{\mathbf{x}}^{(\tau-1)}\| \leq \text{tolerance}$.
-

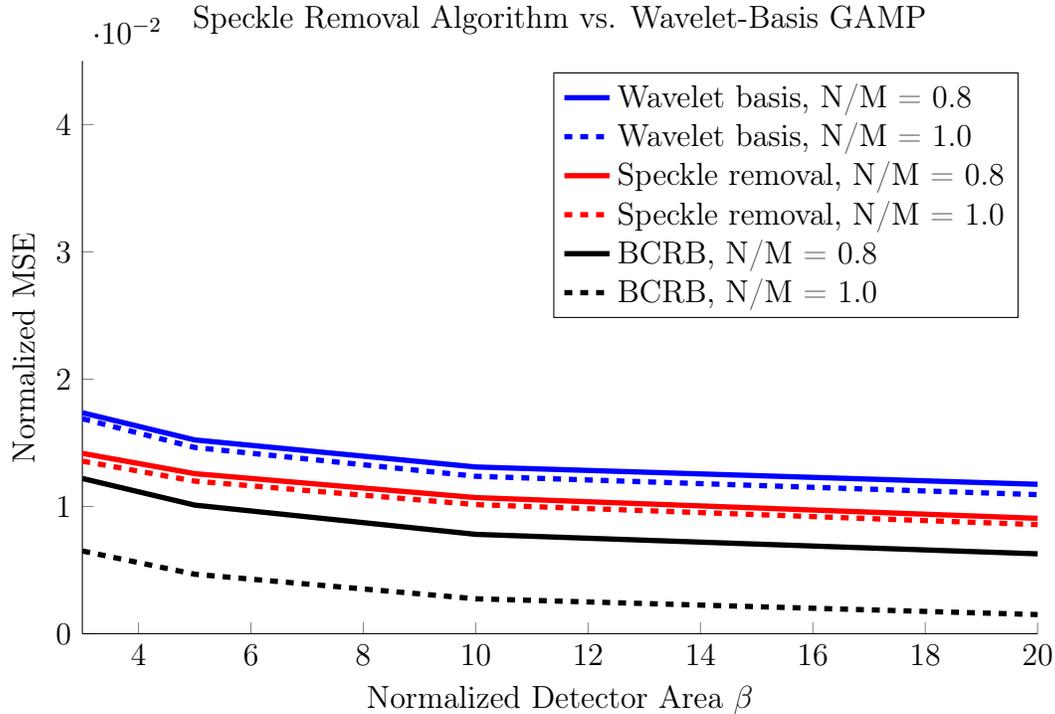
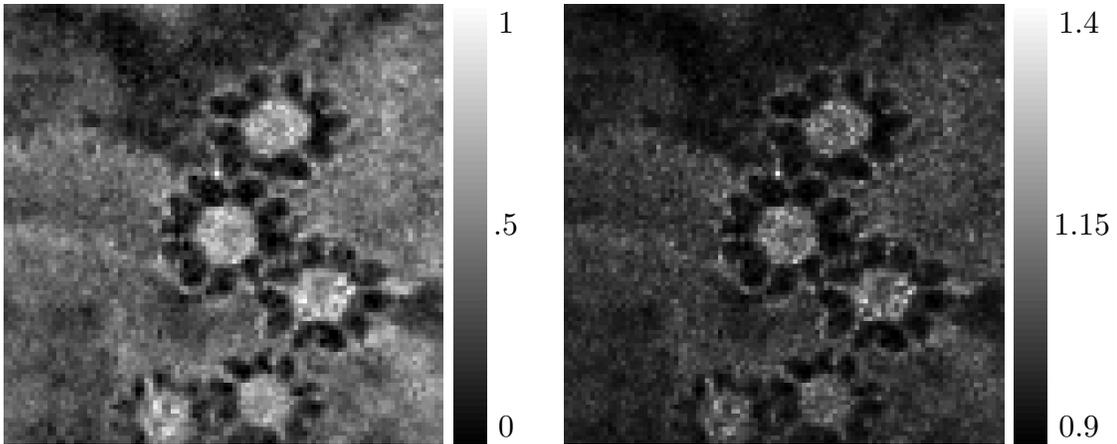


Figure 9-6: Plots versus normalized detector area of the MSE for the wavelet-basis GAMP and speckle-removal algorithms, with the BCRB provided for comparison. The MSE is computed for each of the 100 scenes in our dataset and then averaged.

9.4.4 Simulation Results

In Fig. 9-6 we present a comparison of the MSE of our new algorithm and the wavelet-basis algorithm. The BCRB is also presented, and our new speckle-removal algorithm is only about 20% above this bound. We remind the reader that the bound was derived for an ideal scenario with no discretization error, no inter-pixel interference, and no measurement noise. For qualitative results, we show the images of our test scenes for both our new algorithm, and the previous wavelet-basis GAMP algorithm. These are presented in Fig. 9-8, for $\beta = 3$ (β is the ratio of detector area to transmitter area), and in Fig. 9-9, for $\beta = 10$. In both figures the reconstructions were performed with a unity sampling ratio (the same number of measurements as image pixels). An example of a reconstruction along with the estimate of its associated speckle is shown in Fig. 9-7.

In general the images produced by the new algorithm have better contrast and



(a) MMSE estimation of scene in wavelet basis.

(b) MAP estimate of speckle.

Figure 9-7: The scene reconstruction for $\beta = 5$ is presented on the left, with the MAP estimate of the speckle on the right. Gray-scale bars are presented to show scale. In dark areas of the reconstruction, the speckle remains near its prior estimate.

sharper edges than those produced by the wavelet-basis GAMP estimator, which treated the speckle purely as noise. These new images are subjectively better reconstructions of the original scenes, and also have a lower MSE, as shown in Fig. 9-6. In our new algorithm most of the pixel-to-pixel variations in the scene are estimated to be speckle, which greatly reduces the uncertainty in our sensing matrix, enabling the better reconstructions. However, some of the speckle effects are estimated to be features of the target. This causes a small amount of pixel-to-pixel variation not seen in the images produced by the wavelet-basis GAMP algorithm, for which small-scale variations were strongly suppressed by the uncertainty in the sensing matrix.

To better illustrate the behavior of the new algorithm for changing bucket detector size, we also provide Fig. 9-10. Here we show reconstructions of the test scenes for β equaling 3, 5, 10, and 20, with a unity sampling ratio.

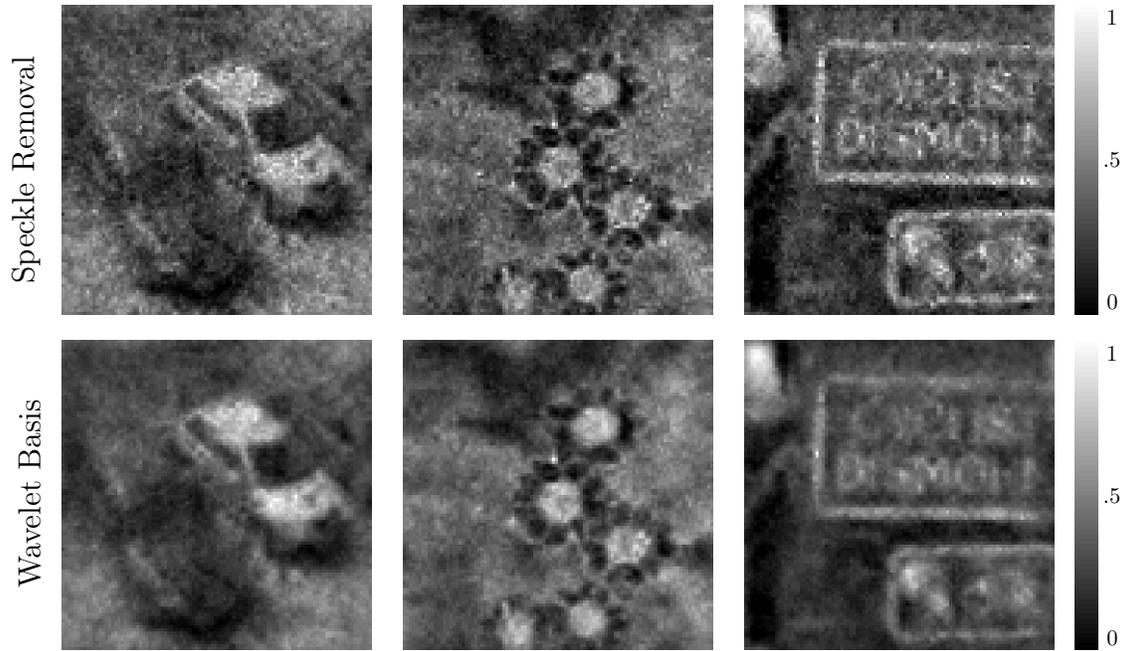


Figure 9-8: Images obtained using the GAMP-based algorithms with $\beta = 3$. The top row is reconstructions using the E-M based speckle-removal algorithm, while the bottom row is for the adaptive wavelet-basis algorithm.

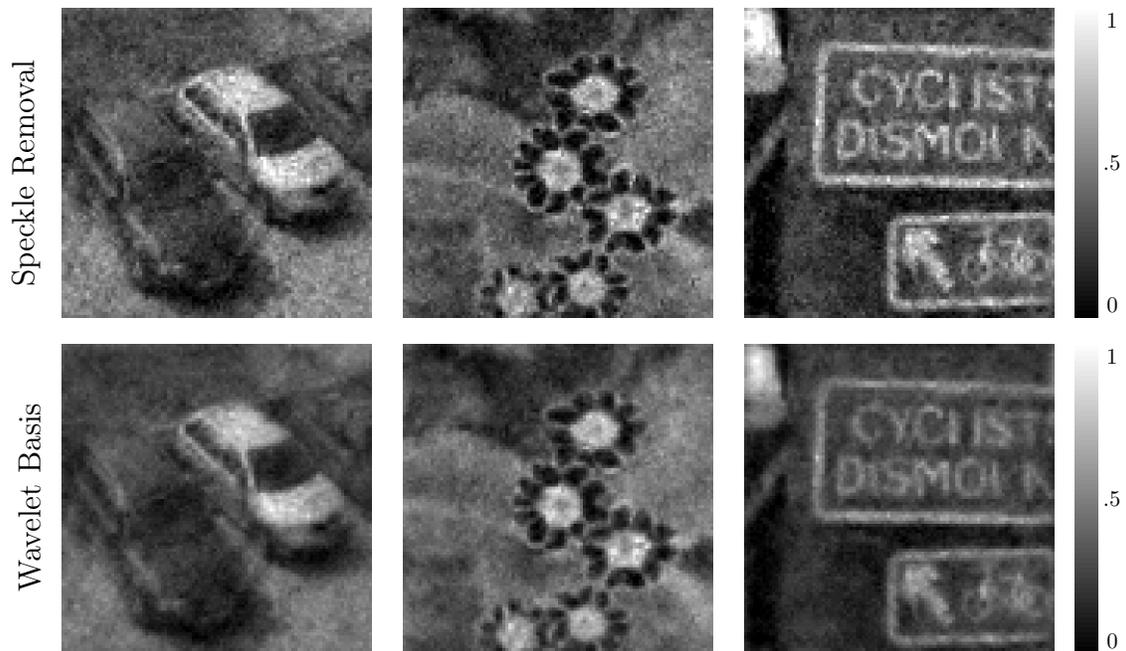


Figure 9-9: Images obtained using the GAMP-based algorithms with $\beta = 10$. The top row is reconstructions using the E-M based speckle-removal algorithm, while the bottom row is for the adaptive wavelet-basis algorithm.

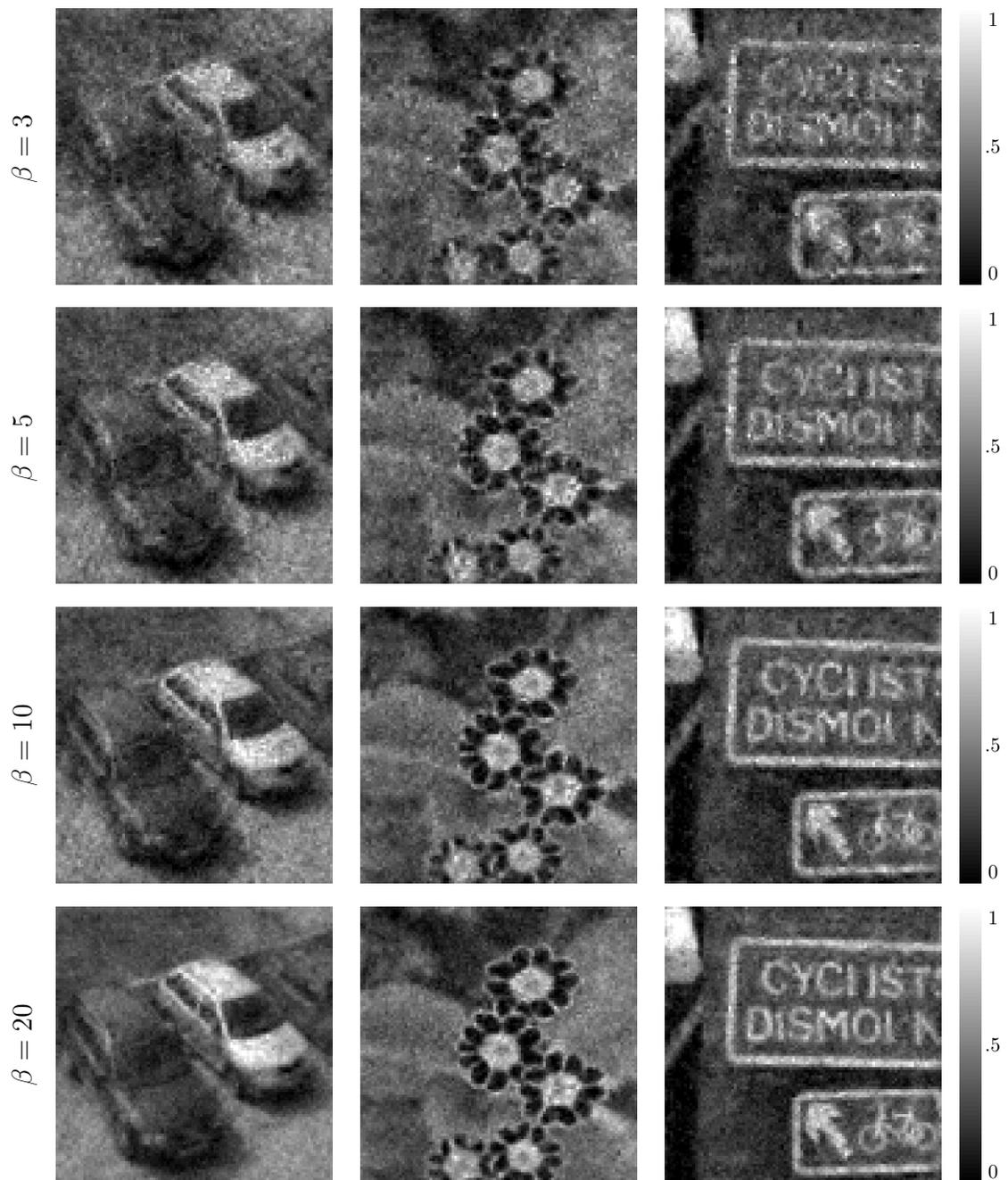


Figure 9-10: Images obtained using the E-M based speckle-removal algorithm. Each row used a different size detector. From top to bottom the β values are, in descending order, 3, 5, 10, and 20.

Chapter 10

Conclusions

In this thesis we have developed a state-of-the-art algorithm for standoff structured-laser-illumination imaging. We have combined a thorough statistical model of the physics of reflective structured-laser illumination, rough-surface reflection, and photodetection, with modern, probabilistic inference techniques to efficiently estimate the target while simultaneously suppressing the effects of speckle.

In Chs. 2 and 3 we developed a physical model for standoff structured-illumination imaging and analyzed the resolution and signal-to-noise ratio of ghost imaging, a simple type of structured-illumination imaging. We showed that ghost imaging had similar resolution and speckle-limited signal-to-noise ratio (SNR) when compared to a standard floodlight LADAR system. However, ghost imaging in general takes longer to form an image. In many ways ghost imaging is a dual of LADAR and allows for imaging when we can create a multi-spatial-mode field at our transmitter, but only have a single-element photodetector to measure reflections. Ghost imaging is a linear estimator, and while there have been some advances in using this method of image reconstruction, they were all heuristically arrived at from transmissive ghost imaging experiments.

With an understanding of the physics and performance of a basic structured-illumination imager (ghost imaging) in hand, our focus shifted to using these same measurements to produce higher quality images. Since new imaging methods require digital processing with a computer, in Ch. 4 we developed a discretized version of

the framework presented in Ch. 2. This allowed us to develop our algorithms while faithfully accounting for all non-idealities in the physical system, including the discretization error.

The second half of the thesis was focused on new algorithms, but first we took Ch. 5 to develop the CRB, which established an ultimate performance goal for our subsequent estimators. As suspected, the ghost imager fell far short of the CRB. Our first new algorithm was the linear minimum mean-squared error (LMMSE) estimator. Like the ghost imager it is a linear estimator, but it is derived so as to have the minimum mean-squared error (MMSE) among all estimators that are linear in the measurements. Unfortunately, a subjective analysis found its performance to be very similar to that of ghost imaging. This was clearly not the advancement in image reconstruction we sought.

Since linear estimators were not sufficient to form clear images for standoff structured-illumination imaging, we sought truly optimal estimators. Because images have thousands to millions of pixels, direct computation of these estimators was not feasible, and we needed to use approximate methods. In Ch. 7 we reviewed the basics of graphical modeling, and more specifically loopy belief propagation (LBP), one of the most successful tools for probabilistic inference. In the beginning of Ch. 8 we reviewed an efficient approximation to LBP for solving the linear mixing problem $\mathbf{y} = \mathbf{Ax}$, called GAMP. While GAMP had previously been extended to handle uncertainty in the measurement matrix, we had to find ways to further extend it to match our imaging scenario. This meant handling extra noise from inter-pixel interference, and more importantly modifying the underlying graph to handle the non-centered nature of our measurements matrix.

We then produced estimates using our modified GAMP algorithm, employing a simplistic iid Gaussian prior to model our target. However, we found these estimates performed only slightly better than the LMMSE estimate. Although at first discouraging, we realized this lack of improvement was simply because for an iid Gaussian prior and a strong detected signal¹, the signal and measurements had a jointly Gaus-

¹In the strong-signal limit the Poisson detection noise becomes Gaussian

sian distribution, for which the LMMSE estimator was the optimal MMSE estimator. Fortunately, there are much better models for targets which our new GAMP-based framework can employ.

Although we made initial connections to GAMP in Ch. 8, it was in Ch. 9 that we started making significant improvements to the reconstruction process. We presented a wavelet-basis model of our target that utilized a sparsifying Laplace distribution. This is consistent with standard compressed sensing techniques, as natural images are known to be sparsely represented in wavelet bases. We also applied the advances in adaptive GAMP to learn parameters of this prior. Both of these refinements used established results with regard to using GAMP for compressed imaging, but required significant calculations to implement effectively in our imaging scenario.

Our new, wavelet-based GAMP algorithm produced significantly better results, both in terms of MSE performance and from subjective analysis of the images. However, we were still treating the image speckle—caused by the rough-surface scattering—purely as noise. The culmination of the thesis was to re-derive our framework to expose the speckle as explicit variables in a joint distribution with the wavelet coefficients and the measurements. We then noted that if the target was known, we could use GAMP to estimate the speckle coefficients. From this observation we developed an alternating algorithm in which we made an MMSE estimate of the wavelet coefficients with GAMP, and then used this new model of the target to aid a maximum a posteriori probability (MAP) estimate of the speckle. We alternated these two steps to produce an estimate of the target that mitigated the effects of speckle. Moreover, we were able to show that the estimates produced by this approach converge to the estimates produced by the E-M algorithm as the number of pixels and measurements increase, providing us with a guarantee that our estimates will not get worse as we iterate.

This new estimate was a significant advancement, both in terms of mean-squared error (MSE) and subjective analysis, and represents the culmination of this thesis. To provide context for the total improvement in scene reconstruction in this thesis, we now present a comparison of the estimate produced by our final algorithm to that

produced by our first, the LMMSE estimate. These can be seen in Figs. 10-1, 10-2, and 10-3 for β —the ratio of the detector area to the transmitter area—being 3, 10, and 20.

As a final note, we return to the usefulness of structured-illumination imaging. In our analysis, structured illumination had its greatest utility at wavelengths for which we do not have good detector arrays, and therefore cannot use floodlight LADAR. However, raster-scanning LADAR only requires a single-pixel detector. The primary advantage of structured-illumination then lies in its ability to form images when undersampling. Our analysis focused on unity sampling and 0.8 undersampling to clarify the comparison among our different algorithms, since the scenes in our dataset did not have a guaranteed sparsity. At these sampling ratios there is only a marginal advantage in using structured-illumination versus a raster-scanning LADAR for standoff imaging. Yet, from the MSE analysis in Fig. 9-6, we know that the quality of our reconstruction is very similar for sampling ratios of 1 and 0.8. These sampling ratios allowed for good reconstructions of the entire dataset, but for many cases we can significantly reduce the sampling ratio while still forming high-quality images.

Although we did not have enough scenes in our dataset to provide a full analysis of image quality as a function of scene sparsity, we have selected three highly-structured scenes (that have very sparse wavelet representations) and present their reconstructions in Figs. 10-4 for a sampling ratio of 0.3, and $\beta = 10$. We also provide the original scenes, the speckled versions of the scenes, and LMMSE reconstructions for the same parameters. From these images we see that our final algorithm can form high-quality reconstructions with a significant undersampling ratio, and reasonably-sized detectors.

From our analysis in this thesis we have shown that our final algorithm has significantly better performance than known ghost imaging (covariance-based) methods for sub-unity sampling ratios. Furthermore, our demonstrated undersampling performance for sparse scenes is impossible for a raster-scanning LADAR to achieve, as it needs to individually image each pixel. Finally, all known attempts to using generic compressed sensing methods for reflective structured-illumination imaging have re-

quired significantly larger detectors than we have used to suppress image speckle and inter-pixel interference. Our results demonstrate state-of-the-art scene reconstruction for structured-laser-illumination imaging.

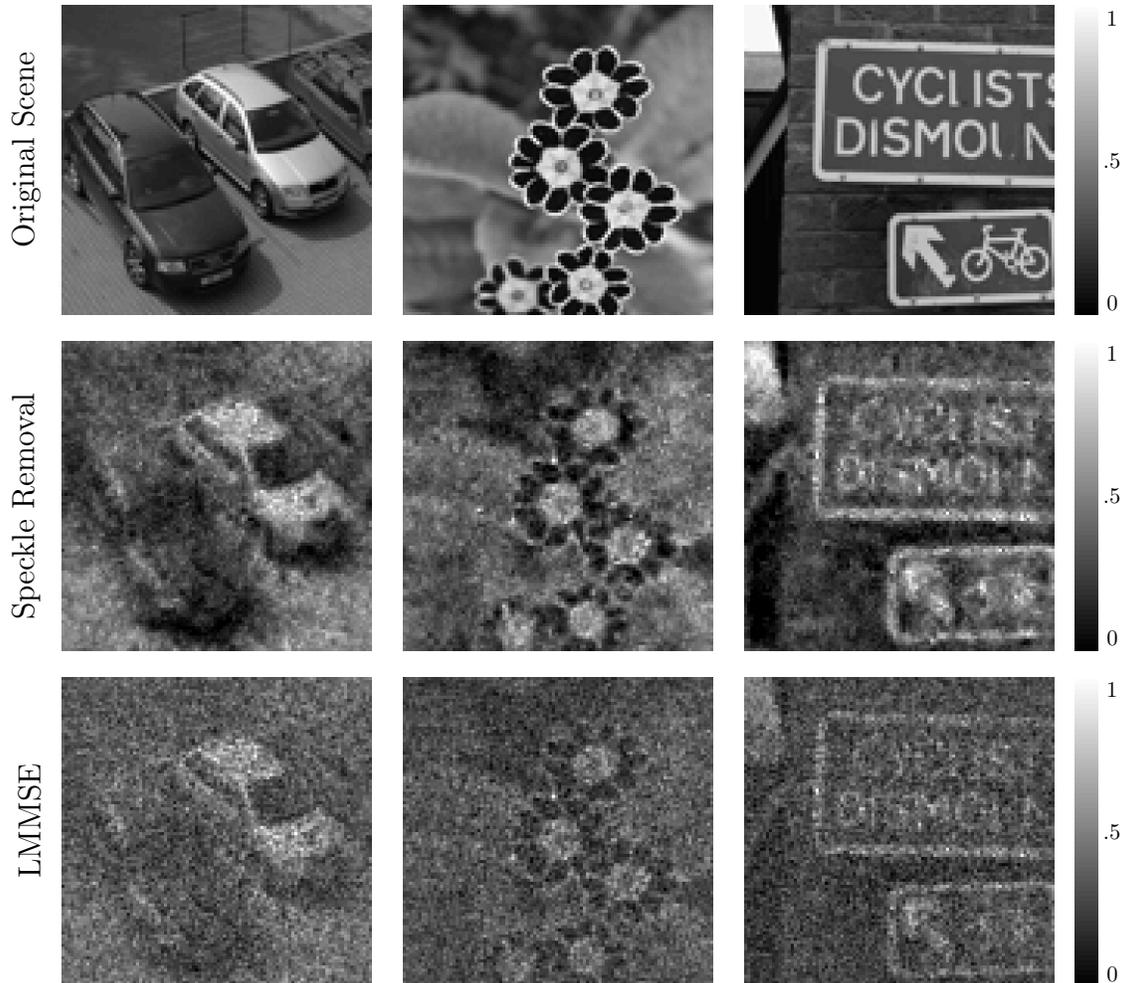


Figure 10-1: Scene reconstructions for $\beta = 3$. The top row is the original scene, the middle row is the speckle removal algorithm, and the bottom row is the LMMSE estimate.

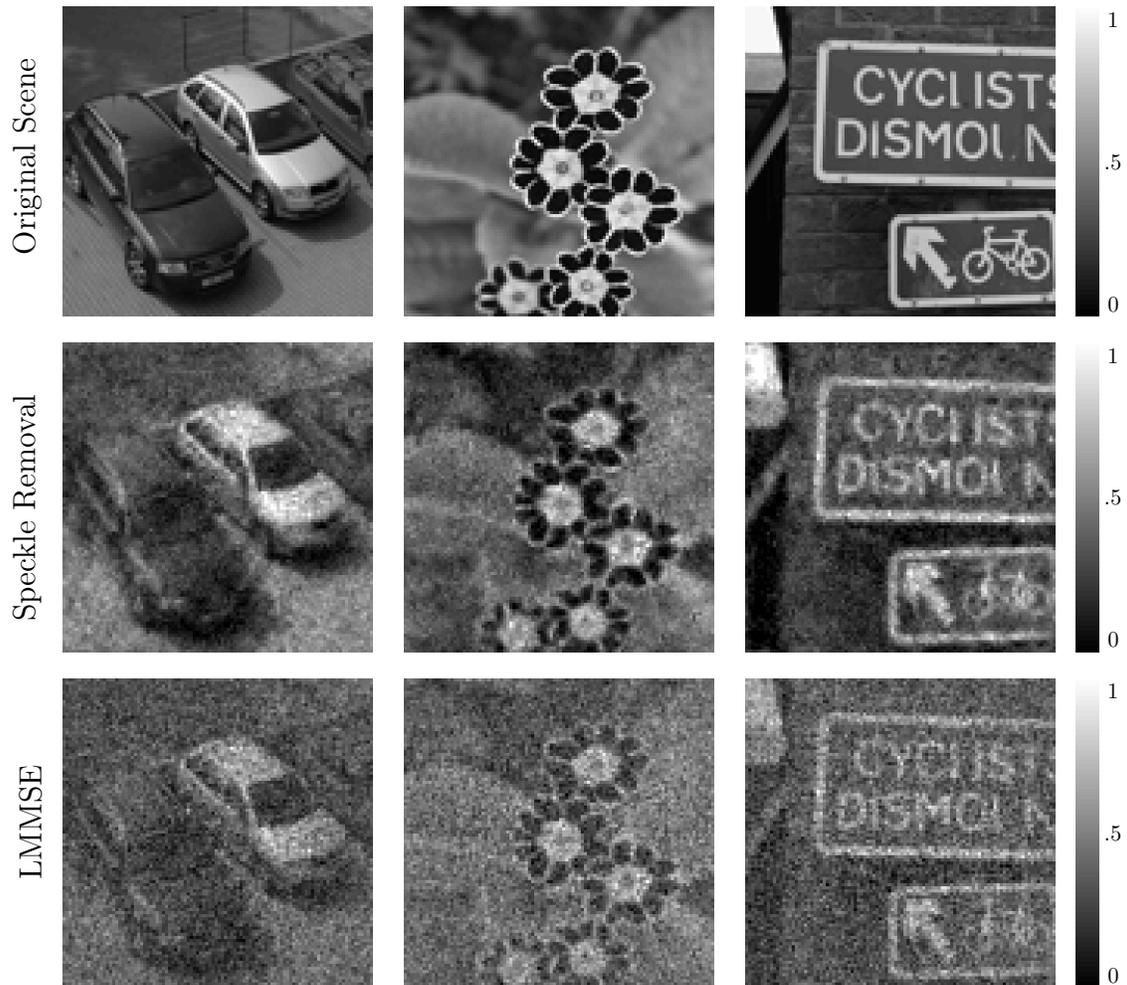


Figure 10-2: Scene reconstructions for $\beta = 10$. The top row is the original scene, the middle row is the speckle removal algorithm, and the bottom row is the LMMSE estimate.

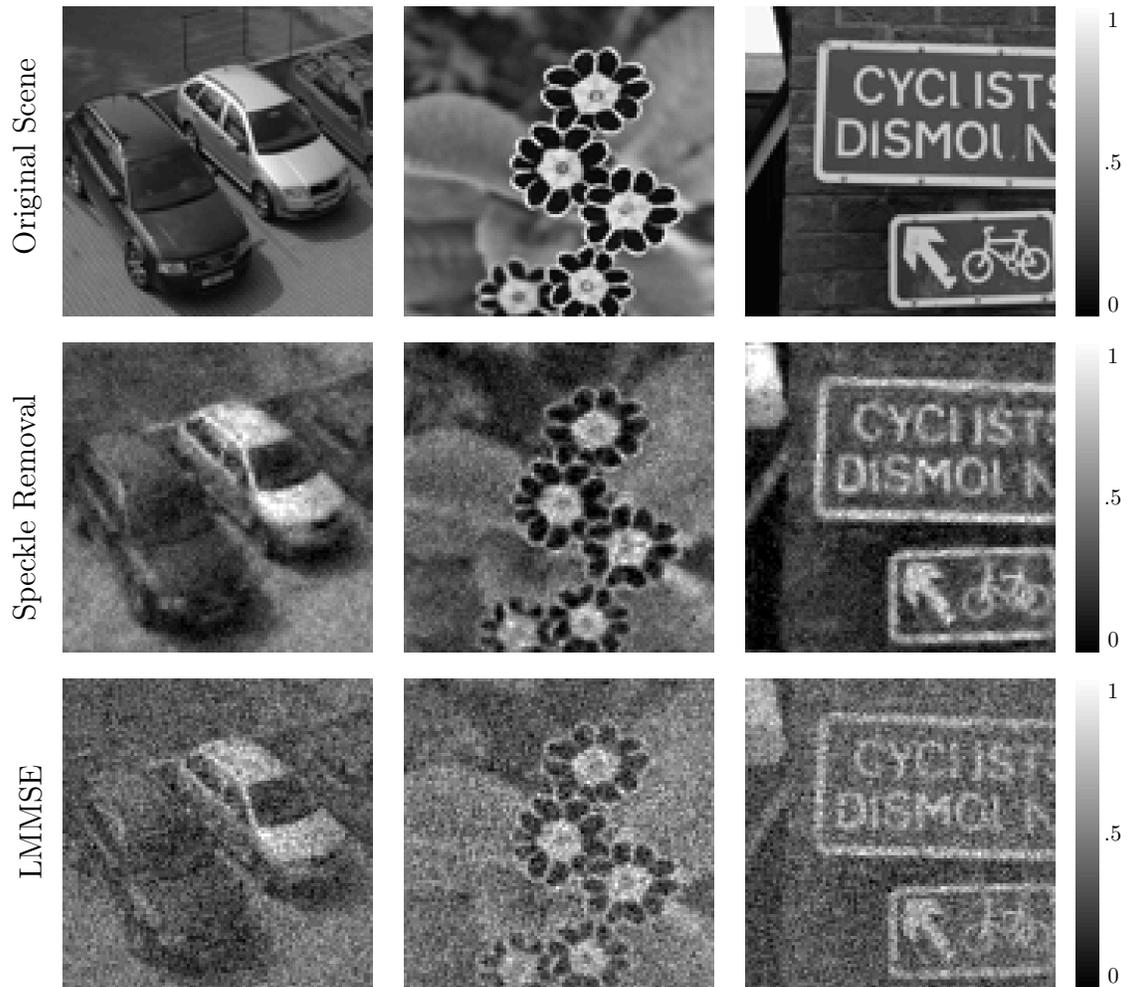


Figure 10-3: Scene reconstructions for $\beta = 20$. The top row is the original scene, the middle row is the speckle removal algorithm, and the bottom row is the LMMSE estimate.

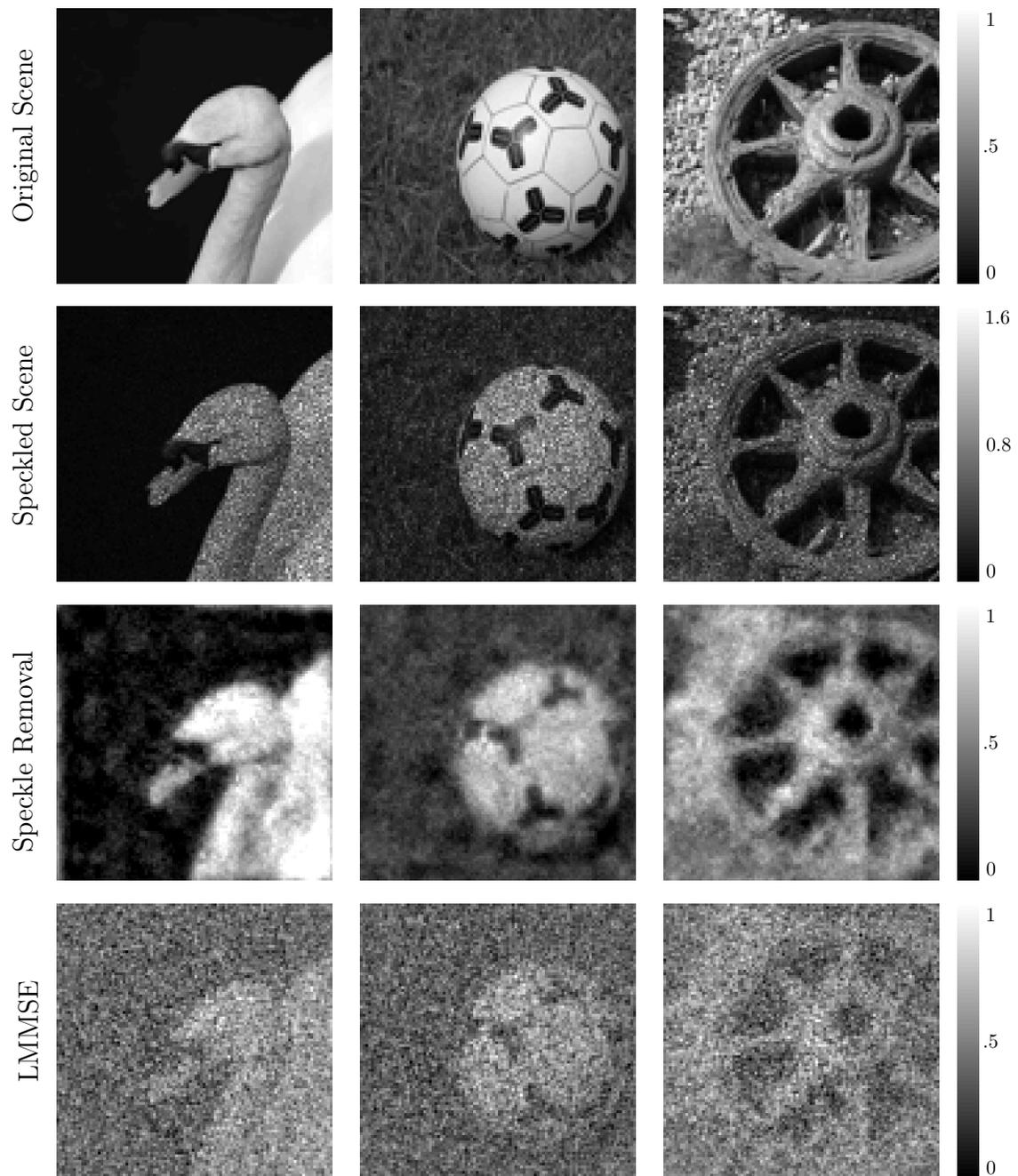


Figure 10-4: Scene reconstructions for $\beta = 10$ and sampling ratio $N/M = 0.3$. The top row is the original scene, the second row is the speckled scene, the third row is the speckle removal algorithm, and the bottom row is the LMMSE estimate.

Appendix A

Physical System

For many aspects of the physical framework in Chs. 2 and 4, we did not provide full derivations or calculations of every term. These include the turbulence square-law approximation, computing a reference field for light propagated from an SLM, and statistics of the rough-surface speckle. This appendix fills in those derivations.

A.1 Turbulence Square-Law Approximation

Herein we derive the square-law approximation for the turbulence structure function. This approximation is used when evaluating the turbulence coherence function, and thus is used in the ghost imager and LADAR resolution derivations. The structure function is originally defined in Eq. (2.8), which we reproduce here,

$$D_m(\boldsymbol{\rho}, \boldsymbol{\rho}') = 2.91k_0^2L \int_0^1 ds C_{n,m}^2(sL) |\boldsymbol{\rho}s + \boldsymbol{\rho}'(1-s)|^{\frac{5}{3}}. \quad (\text{A.1})$$

To aid our derivations we now re-formulate it in terms of the normalized turbulence strength $\mathcal{C}_{n,m}^2(sL)$. From Eq.(2.12) we have $C_{n,m}^2(sL) = \mathcal{C}_{n,m}^2(sL) \int_0^1 ds C_{n,m}^2(sL) = \mathcal{C}_{n,m}^2(sL)\mathbb{C}_{n,m}$, and the structure function becomes

$$D_m(\boldsymbol{\rho}, \boldsymbol{\rho}') = 2.91k_0^2L\mathbb{C}_{n,m} \int_0^1 ds \mathcal{C}_{n,m}^2(sL) |\boldsymbol{\rho}s + \boldsymbol{\rho}'(1-s)|^{\frac{5}{3}}. \quad (\text{A.2})$$

The square-law approximation replaces the 5/3 exponent with 2, so that we can analytically evaluate the integral. However, this does not maintain the dimensionality of the structure function, so an adjustment will need to be made to the leading coefficients. To motivate our approach, we first investigate the structure function when $\boldsymbol{\rho}' = \mathbf{0}$, for which the approximation is not needed to perform the integral. In this case,

$$\begin{aligned} D_m(\boldsymbol{\rho}, \mathbf{0}) &= 2.91k_0^2 LC_{n,m} |\boldsymbol{\rho}|^{\frac{5}{3}} \int_0^1 ds s^{\frac{5}{3}} \\ &= 1.09k_0^2 LC_{n,m} |\boldsymbol{\rho}|^{\frac{5}{3}} \end{aligned} \quad (\text{A.3})$$

$$= \left(\frac{|\boldsymbol{\rho}|}{\varrho_m} \right)^{\frac{5}{3}}. \quad (\text{A.4})$$

It is from this formulation that we derive the spherical-wave coherence length

$$\varrho_m = (1.09k_0^2 LC_{n,m})^{-3/5}. \quad (\text{A.5})$$

which was presented in Eq. (2.13), but with $C_{n,m}$ replaced by its definition. The square-law approximation is to then replace the 5/3 with 2, as

$$\left(\frac{|\boldsymbol{\rho}|}{\varrho_m} \right)^{\frac{5}{3}} \rightarrow \left(\frac{|\boldsymbol{\rho}|}{\varrho_m} \right)^2. \quad (\text{A.6})$$

In essence, the square law approximation in this case amounts to raising the structure function $D_m(\boldsymbol{\rho}, \mathbf{0})$ to the power of 6/5.

Therefore, our square law approximation for evaluating Eq. (A.2) is to replace the exponent 5/3 that appears inside the integral with 2, evaluate the integral, and then raise the leading coefficients to the power 6/5. Exchanging the exponents, the structure function becomes

$$\begin{aligned} D_m(\boldsymbol{\rho}, \boldsymbol{\rho}') &= 2.91k_0^2 LC_{n,m} \int_0^1 ds C_{n,m}^2(sL) |\boldsymbol{\rho}s + \boldsymbol{\rho}'(1-s)|^2 \\ &= 2.91k_0^2 LC_{n,m} \int_0^1 ds C_{n,m}^2(sL) [|\boldsymbol{\rho}|^2 s^2 + 2\boldsymbol{\rho} \cdot \boldsymbol{\rho}'(s-s^2) + |\boldsymbol{\rho}'|^2(1-s)^2]. \end{aligned}$$

Defining ζ_m and ζ'_m as in Eq. (2.10) and (2.11) this becomes

$$D_m(\boldsymbol{\rho}, \boldsymbol{\rho}') = 1.09k_0^2 L \mathbb{C}_{n,m} [|\boldsymbol{\rho}|^2 \zeta_m + \boldsymbol{\rho} \cdot \boldsymbol{\rho}' (1 - \zeta_m - \zeta'_m) + |\boldsymbol{\rho}'|^2 \zeta'_m]. \quad (\text{A.7})$$

Raising the leading coefficients to $6/5$, we achieve our final result,

$$\begin{aligned} D_m(\boldsymbol{\rho}, \boldsymbol{\rho}') &= (1.09k_0^2 L \mathbb{C}_{n,m})^{\frac{6}{5}} [|\boldsymbol{\rho}|^2 \zeta_m + \boldsymbol{\rho} \cdot \boldsymbol{\rho}' (1 - \zeta_m - \zeta'_m) + |\boldsymbol{\rho}'|^2 \zeta'_m] \\ &= \frac{|\boldsymbol{\rho}|^2 \zeta_m + \boldsymbol{\rho} \cdot \boldsymbol{\rho}' (1 - \zeta_m - \zeta'_m) + |\boldsymbol{\rho}'|^2 \zeta'_m}{\varrho_m^2}, \end{aligned} \quad (\text{A.8})$$

which is the square-law presented in Eq. (2.9).

A.2 Computing the Reference Irradiance Pattern for SLM-Generated Field

We primarily consider a computational approach to structured-illumination, in which the structured fields are generated by shining a laser on a pseudo-randomly modulated spatial light modulator (SLM). The reference on-target irradiance pattern is computed via diffraction theory. This pattern, originally presented in Eq. (3.1), is

$$I(\boldsymbol{\rho}, n) = \left| \int d\boldsymbol{\rho}' \xi(\boldsymbol{\rho}', n) \frac{k_0 e^{ik_0 |\boldsymbol{\rho} - \boldsymbol{\rho}'|^2 / 2L}}{i2\pi L} \right|^2, \quad (\text{A.9})$$

where $\xi(\boldsymbol{\rho}', n)$ is the SLM field pattern. In this section we first show how to compute the reference field efficiently, and then use that result to derive the average irradiance envelope for the reference.

A.2.1 Efficient Computation of Reference Field

To simplify our analysis we will work in Cartesian coordinates, $\boldsymbol{\rho} \equiv (\rho_x, \rho_y)$. Defining the reference field pattern as

$$\xi_L(\rho_x, \rho_y, n) = \frac{k_0}{i2\pi L} \int d\rho'_x \int d\rho'_y \xi(\rho'_x, \rho'_y, n) e^{\frac{ik_0}{2L} [(\rho_x - \rho'_x)^2 + (\rho_y - \rho'_y)^2]}, \quad (\text{A.10})$$

the reference irradiance pattern can be written $I(\rho_x, \rho_y, n) \equiv |\xi_L(\rho_x, \rho_y, n)|^2$.

The spatial pattern at the source is an array of phase-shifted pixels, and is normalized so that $\int d\boldsymbol{\rho}' |\xi(\boldsymbol{\rho}', n)|^2 = 1$. Therefore, we model $\xi(\boldsymbol{\rho}', n) \equiv \xi(\rho'_x, \rho'_y, n)$ as

$$\xi(\rho'_x, \rho'_y, n) = \frac{1}{A_s} \sum_s \text{rect}\left(\frac{\rho'_x - \rho'_{x_s}}{w_s}\right) \text{rect}\left(\frac{\rho'_y - \rho'_{y_s}}{w_s}\right) e^{i\phi_{n,s}}, \quad (\text{A.11})$$

where pixel s has width w_s , area $A_s = w_s^2$, is centered at ρ_{x_s} , and has a phase shift $\phi_{n,s}$ for pulse n . The reference field can then be rewritten as

$$\xi_L(\rho_x, \rho_y, n) = \frac{k_0}{i2\pi L} \sum_s \frac{e^{i\phi_{n,s}}}{A_s} \int_{\rho'_{x_s} - w_s/2}^{\rho'_{x_s} + w_s/2} d\rho'_x \int_{\rho'_{y_s} - w_s/2}^{\rho'_{y_s} + w_s/2} d\rho'_y e^{\frac{ik_0}{2L} [(\rho_x - \rho'_x)^2 + (\rho_y - \rho'_y)^2]}, \quad (\text{A.12})$$

Our goal in the rest of this section is to simplify this computation of the propagation of the pixel shape. It is evident from this formulation that the propagation is identical for both the ρ_x and ρ_y coordinates, so in making our approximations we only need to consider one dimension, but the results will apply for both.

If we were in the far-field of the SLM's pupil, we could make the standard Fraunhofer approximation and treat the propagation as a Fourier transform. However, we are in the more restrictive far-field of the coherence function. Our approximation will be a little different, but the end result is that we can still efficiently compute the field with an adjusted Fourier transform.

To start we note that each SLM pixel has a width w_s , while the total width of the source is denoted W_s . Our far-field conditions for partial-coherence propagation

from Sec. 2.5 can then be recast as

$$k_0 w_s W_s / 2L \ll 1. \quad (\text{A.13})$$

To approximate the propagation kernel for the field generated by SLM pixel s , we take a Taylor expansion of its exponent around its center coordinate $\boldsymbol{\rho}'_s \equiv (\rho'_{x_s}, \rho'_{y_s})$, and attempt to simplify the result with this far-field assumption.

Next we define the ρ_x -dependent part of phase of the propagation kernel in Eq. (A.12) as $f(\rho'_x) = \frac{k_0}{2L}(\rho_x - \rho'_x)^2$. In general, the full Taylor expansion of a function $f(\rho'_x)$ around ρ'_{x_s} is

$$f(\rho'_x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(\rho'_{x_s})}{n!} (\rho'_x - \rho'_{x_s})^n, \quad (\text{A.14})$$

where $f^{(n)}$ denotes the n^{th} derivative of f . Our first step is to find these derivatives. Since $f(\rho'_x)$ is simply a second-order polynomial in ρ'_x , $f^{(n)}(\rho'_x) = 0$ for $n \geq 3$, so we only need to find the first two derivatives, which are

$$f^{(1)}(\rho'_{x_s}) = \frac{\partial}{\partial \rho'_x} \frac{k_0}{2L} (\rho_x - \rho'_x)^2 \Big|_{\rho'_x = \rho'_{x_s}} = \frac{k_0}{L} (\rho'_{x_s} - \rho_x) \quad (\text{A.15})$$

$$f^{(2)}(\rho'_{x_s}) = \frac{\partial^2}{\partial \rho'^2_x} \frac{k_0}{2L} (\rho_x - \rho'_x)^2 \Big|_{\rho'_x = \rho'_{x_s}} = \frac{\partial}{\partial \rho'_x} \frac{k_0}{L} (\rho'_x - \rho_x) \Big|_{\rho'_x = \rho'_{x_s}} = \frac{k_0}{L}. \quad (\text{A.16})$$

Thus, the Taylor expansion simplifies to

$$\begin{aligned} f(\rho'_x) &= \frac{k_0}{2L} (\rho_x - \rho'_{x_s})^2 + \frac{k_0}{L} (\rho'_{x_s} - \rho_x) (\rho'_x - \rho'_{x_s}) + \frac{k_0}{2L} (\rho'_x - \rho'_{x_s})^2 \\ &= \frac{k_0}{2L} (\rho_x - \rho'_{x_s})^2 - \frac{k_0}{L} \rho_x (\rho'_x - \rho'_{x_s}) + \frac{k_0}{L} \rho'_{x_s} (\rho'_x - \rho'_{x_s}) + \frac{k_0}{2L} (\rho'_x - \rho'_{x_s})^2. \end{aligned} \quad (\text{A.17})$$

Now, since we are considering the field that propagates from a single pixel, we are restricting our possible coordinates at the SLM to $\{\rho_x : \rho_{x_s} - w_s/2 \leq \rho_x \leq \rho_{x_s} + w_s/2\}$, and therefore we are guaranteed $|\rho_x - \rho_{x_s}| \leq w_s/2$. Furthermore, $|\rho_{x_s}| < W_s/2$ must hold for the pixel to be on the SLM. Using these two inequalities, along with our

far-field condition in Eq. (A.13), we have that

$$\frac{k_0}{2L}(\rho'_x - \rho'_{x_s})^2 < \frac{k_0 w_s^2}{2L} \ll 1 \quad (\text{A.18})$$

$$\left| \frac{k_0}{L} \rho_{x_s} (\rho_x - \rho_{x_s}) \right| < \frac{k_0 W_s w_s}{2L} \ll 1. \quad (\text{A.19})$$

Thus the last two terms in Eq. (A.17) can be neglected, and expanding the remaining terms we find

$$f(x) \approx \frac{k_0}{2L} \rho'_{x_s}{}^2 + \frac{k_0}{2L} \rho_x^2 - \frac{k_0}{L} \rho'_x \rho_x. \quad (\text{A.20})$$

We see that kernel can be approximated by a curvature term for the center of the pixel, $\frac{k_0}{2z} \rho'_{x_s}{}^2$, the standard far-field Fourier transform term $-\frac{k_0}{z} \rho_x \rho'_x$, and the output plane curvature $\frac{k_0}{2z} \rho_x^2$. Thus, the far-field can be calculated via a Fourier transform of the SLM pattern for each pixel (which can be pre-computed), with the only difference from standard Fraunhofer diffraction being a phase adjustment term for each pixel, which arises from its offset from center. Equation (A.12) then becomes

$$\xi_L(\rho_x, \rho_y, n) = \frac{k_0 e^{\frac{ik_0}{2L}(\rho_x^2 + \rho_y^2)}}{i2\pi L} \sum_s e^{\frac{ik_0}{2L}(\rho'_{x_s}{}^2 + \rho'_{y_s}{}^2)} \frac{e^{i\phi_{s,n}}}{A_s} \int_{\rho'_{x_s}-w_s/2}^{\rho'_{x_s}+w_s/2} d\rho'_x \int_{\rho'_{y_s}-w_s/2}^{\rho'_{y_s}+w_s/2} d\rho'_y e^{-\frac{ik_0}{2L}(\rho_x \rho'_x + \rho_y \rho'_y)}, \quad (\text{A.21})$$

The final integral is the Fourier transform of a shifted rectangle in both coordinates, i.e., for ρ'_x the integral is

$$\int_{\rho'_{x_s}-w_s/2}^{\rho'_{x_s}+w_s/2} d\rho'_x e^{-\frac{ik_0}{L} \rho_x \rho'_x} = w_s \text{sinc} \left(\frac{w_s \rho_x}{\lambda_0 L} \right) e^{-\frac{ik_0}{L} \rho_x \rho'_{x_s}}, \quad (\text{A.22})$$

and our final calculation of the field profile at range L is

$$\begin{aligned}\xi_L(\rho_x, \rho_y, n) &= \frac{e^{\frac{ik_0}{2L}(\rho_x^2 + \rho_y^2)}}{i\lambda_0 L} \operatorname{sinc}\left(\frac{w_s \rho_x}{\lambda_0 L}\right) \operatorname{sinc}\left(\frac{w_s \rho_y}{\lambda_0 L}\right) \\ &\times \sum_s e^{\frac{ik_0}{2L}(\rho'_{x_s}{}^2 + \rho'_{y_s}{}^2)} e^{-\frac{ik_0}{L}(\rho_x \rho'_{x_s} + \rho_y \rho'_{y_s})} e^{i\phi_{s,n}}.\end{aligned}\quad (\text{A.23})$$

A.2.2 Average Reference Irradiance

In Eq. (4.6) we defined the average irradiance envelope

$$\epsilon_p = \frac{\sum_n \langle I_{np} \rangle}{\sum_{n,q} \langle I_{nq} \rangle}, \quad (\text{A.24})$$

and presented an explicit calculation for the Gaussian-Schell model in Eq. (4.7). The discrete reference pattern I_{np} is simply the continuous reference pattern in Eq. (A.9) evaluated at $\boldsymbol{\rho} = \boldsymbol{\rho}_p$, multiplied by a scaling term. Since ϵ_p is normalized this scaling does not matter, so we only need to compute

$$\begin{aligned}\langle I(\boldsymbol{\rho}_p, n) \rangle &\equiv \langle |\xi_L(\rho_{x_p}, \rho_{y_p}, n)|^2 \rangle \\ &= \frac{1}{\lambda_0^2 L^2} \operatorname{sinc}^2\left(\frac{w_s \rho_x}{\lambda_0 L}\right) \operatorname{sinc}^2\left(\frac{w_s \rho_y}{\lambda_0 L}\right) \\ &\times \left\langle \left| \sum_s e^{\frac{ik_0}{2L}(\rho'_{x_s}{}^2 + \rho'_{y_s}{}^2)} e^{-\frac{ik_0}{L}(\rho_x \rho'_{x_s} + \rho_y \rho'_{y_s})} e^{i\phi_{s,n}} \right|^2 \right\rangle\end{aligned}\quad (\text{A.25})$$

$$= \frac{1}{\lambda_0^2 L^2} \operatorname{sinc}^2\left(\frac{w_s \rho_x}{\lambda_0 L}\right) \operatorname{sinc}^2\left(\frac{w_s \rho_y}{\lambda_0 L}\right) \quad (\text{A.26})$$

since $\langle e^{i\phi_{s_1,n}} e^{i\phi_{s_2,n}} \rangle = \delta_{s_1, s_2}$. Therefore, the average irradiance envelope vector for light generated by an SLM is

$$\epsilon_p = \frac{\operatorname{sinc}^2\left(\frac{\rho_{x_p} w_s}{\lambda_0 L}\right) \operatorname{sinc}^2\left(\frac{\rho_{y_p} w_s}{\lambda_0 L}\right)}{\sum_q \operatorname{sinc}^2\left(\frac{\rho_{x_q} w_s}{\lambda_0 L}\right) \operatorname{sinc}^2\left(\frac{\rho_{y_q} w_s}{\lambda_0 L}\right)}, \quad (\text{A.27})$$

A.3 Rough-Surface Statistics

In Sec. 2.3 we developed a model for the reflection coefficient, $T(\boldsymbol{\rho})$, of a rough surface. We presented the target's rough-surface as a material with a slowly-varying average intensity reflectivity pattern $\mathcal{T}(\boldsymbol{\rho})$ that had microscopic depth variations across the surface, $d(\boldsymbol{\rho})$. These depth variations lead to phase modulation of the reflected light,

$$\vartheta(\boldsymbol{\rho}) = e^{2ik_0d(\boldsymbol{\rho})}, \quad (\text{A.28})$$

so that the reflection coefficient is

$$T(\boldsymbol{\rho}) = \sqrt{\mathcal{T}(\boldsymbol{\rho})}\vartheta(\boldsymbol{\rho}). \quad (\text{A.29})$$

In Sec. 2.3 we argued that the effect of $\vartheta(\boldsymbol{\rho})$, when the reflected light is measured in the far-field, is qualitatively and quantitatively similar to that of a zero-mean, complex-Gaussian random process, with correlation function

$$\langle \vartheta(\boldsymbol{\rho}_1)\vartheta^*(\boldsymbol{\rho}_2) \rangle = \pi \varrho_T^2 \delta(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2). \quad (\text{A.30})$$

This is because each point in the far-field is the superposition of reflections from many coherence areas of the rough surface, and a CLT argument can be applied. This Gaussian model was used throughout the thesis, and is standard in laser radar theory [22]. In this appendix we derive the mean and covariance function of $\vartheta(\boldsymbol{\rho})$ from the phase model in Eq. (A.28). We start by modeling the height variations $d(\boldsymbol{\rho})$ as a zero-mean stationary Gaussian random process [23] with a correlation length of ρ_d and variance σ_d^2 such that for two points on the surface the joint distribution is

$$p_d(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2) = \frac{\exp \left[-\frac{1}{2(1-\rho^2(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2))} \left(\frac{d(\boldsymbol{\rho}_1)^2}{\sigma_d^2} + \frac{d(\boldsymbol{\rho}_2)^2}{\sigma_d^2} - \frac{2\rho(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2)d(\boldsymbol{\rho}_1)d(\boldsymbol{\rho}_2)}{\sigma_d^2} \right) \right]}{2\pi\sigma_d^2\sqrt{1-\rho^2(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2)}} \quad (\text{A.31})$$

where the correlation coefficient is

$$\rho(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2) = \exp \left[-\frac{|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2|^2}{\rho_d^2} \right]. \quad (\text{A.32})$$

The mean and variance of the phase fluctuations can then be found from this distribution.

A.3.1 Mean

Since the height variations are a homogeneous process, the mean of $\vartheta(\boldsymbol{\rho})$ is a constant function of $\boldsymbol{\rho}$, and can be found from an ensemble average as

$$\langle \vartheta(\boldsymbol{\rho}) \rangle = e^{\langle 2ik_0d(\boldsymbol{\rho}) \rangle + \frac{1}{2} \text{var}[2ik_0d(\boldsymbol{\rho})]} \quad (\text{A.33})$$

$$= e^{-2k_0^2\sigma_d^2}, \quad (\text{A.34})$$

where the averaging is with respect to the distribution $p_d(\boldsymbol{\rho})$. For optical imaging the height variations are orders of magnitude larger than the wavelength of light, so $k_0^2\sigma_d^2 \gg 1$, which means that

$$\langle \vartheta(\boldsymbol{\rho}) \rangle \approx 0. \quad (\text{A.35})$$

A.3.2 Covariance

Since the rough-surface field reflectivity is essentially a zero-mean process, the covariance between two points ($\boldsymbol{\rho}_1$ and $\boldsymbol{\rho}_2$) is given by the phase-insensitive correlation function

$$\langle \vartheta(\boldsymbol{\rho}_1)\vartheta^*(\boldsymbol{\rho}_2) \rangle = \langle e^{2ik_0h(\boldsymbol{\rho}_1)}e^{-2ik_0h(\boldsymbol{\rho}_2)} \rangle = e^{-4k_0^2\sigma_d^2(1-\rho(\boldsymbol{\rho}_1,\boldsymbol{\rho}_2))}. \quad (\text{A.36})$$

The correlation coefficient of the height fluctuations, $\rho(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2)$, is given in Eq. (A.32), but this form makes it difficult to employ Eq. (A.36) in our subsequent calculations. However, we note that for $\frac{|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2|^2}{\rho_d^2} > 1$, $\rho(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2) \approx 0$. Therefore, we can take a two term Taylor expansion of the correlation coefficient, so that the phase fluctuation

correlation function is approximately

$$\langle \vartheta(\boldsymbol{\rho}_1)\vartheta^*(\boldsymbol{\rho}_2) \rangle \approx e^{-4k_0^2\sigma_d^2\frac{|\boldsymbol{\rho}_1-\boldsymbol{\rho}_2|^2}{\rho_d^2}}. \quad (\text{A.37})$$

This correlation function is very narrow, and the correlation length $\varrho_T^2 = \rho_d^2/4k_0^2\sigma_d^2$ is on the same order of magnitude as a wavelength. Since this is much narrower than any other feature of the target or impinging field, it can be approximated by a delta function response,

$$\langle \vartheta(\boldsymbol{\rho}_1)\vartheta^*(\boldsymbol{\rho}_2) \rangle \approx e^{-\frac{|\boldsymbol{\rho}_2-\boldsymbol{\rho}_1|^2}{\varrho_T^2}} \approx \pi\varrho_T^2\delta(\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1),$$

where the last approximation is valid inside an integral (such as for propagation). This is the model used in Eq. (2.14) for the analysis of ghost imaging and conventional LADAR.

A.4 Rough-Surface Induced Speckle

Light reflected from a rough surface suffers interference as it propagates, that leads to fluctuations in our measurements. The fluctuations are exacerbated by the non-uniform field impinging on each pixel, something that necessitated our calculating a variety of noise terms throughout the thesis. In this section we will be computing κ_α , Γ_0 , Γ_1 , Γ_2 , and Γ_3 , which first appear, respectfully, in Eqs. (4.23), (4.26), (4.27), (9.33), and (9.34).

For our simulations, we use a square pixel and a square detector. The interaction of the field and the pixel imply that some of these calculations are intractable for a square pixel, and so for those calculations we employ a Gaussian pixel. It is important to note that this Gaussian pixel is only being used to find the speckle-term variance, not the primary intensity-transfer term. There are many sources of noise in our structured-illumination imager and our algorithms have turn out to be robust to slight changes in these auxiliary-noises levels

A.4.1 Precise Noise Calculations

The speckle-noise terms that we need are the following,

$$\kappa_\alpha(p, q) = \frac{1}{\alpha A_p^2} \int_{\mathcal{A}_q^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_{p_2}^2} d\boldsymbol{\rho}_2 e^{-|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2|^2 / \alpha \rho_L^2} \quad (\text{A.38})$$

$$\Gamma_0(p, q) = \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_q^2} d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} \quad (\text{A.39})$$

$$\Gamma_1 = \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} e^{-\frac{|\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2|^2}{\rho_L^2}} \quad (\text{A.40})$$

$$\Gamma_2 = \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} e^{-\frac{|\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2|^2}{2\rho_L^2}} \quad (\text{A.41})$$

$$\Gamma_3 = \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^3} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_2 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_3 \\ \times e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} e^{-\frac{|\boldsymbol{\rho}'_3 - \boldsymbol{\rho}'_2|^2 + |\boldsymbol{\rho}'_3 - \boldsymbol{\rho}'_1|^2}{2\rho_L^2}}. \quad (\text{A.42})$$

We define these calculations for a square pixel and a square detector, with side lengths w_p and W_b , but for Γ_1 , Γ_2 , and Γ_3 we employ a Gaussian pixel to aid the analysis.

Calculating κ_α

The simplest of these terms is $\kappa_\alpha(p, q)$. We only end up needing the $p = q$ case, which becomes

$$\kappa_\alpha = \frac{1}{\alpha A_p^2} \int_{\mathcal{A}_p} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_p} d\boldsymbol{\rho}_2 e^{-|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2|^2 / \alpha \rho_L^2} \\ = \frac{\alpha \rho_L^4}{A_p^2} \left[\left(e^{-w_p^2 / \alpha \rho_L^2} - 1 \right) + \sqrt{\frac{\pi}{\alpha}} \frac{w_p}{\rho_L} \operatorname{erf} \left[\frac{w_p}{\rho_L \sqrt{\alpha}} \right] \right]^2,$$

where erf is the standard error function

Calculating $\Gamma_0(p, q)$

To evaluate $\Gamma_0(p, q)$, we first perform the $\boldsymbol{\rho}'_1, \boldsymbol{\rho}'_2$ integrations,

$$\begin{aligned} & \frac{1}{A_p^2} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_q^2} d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} \\ &= \text{sinc}^2 \left(\frac{(\rho_1^x - \rho_2^x)w_p}{\lambda_0 L} \right) \text{sinc}^2 \left(\frac{(\rho_1^y - \rho_2^y)w_p}{\lambda_0 L} \right) \\ & \times \cos \left[\frac{k_0}{L}(\rho_p^x - \rho_q^x)(\rho_1^x - \rho_2^x) + \frac{k_0}{L}(\rho_p^y - \rho_q^y)(\rho_1^y - \rho_2^y) \right]. \end{aligned} \quad (\text{A.43})$$

where $\boldsymbol{\rho}_i = (\rho_i^x, \rho_i^y)$, and we have anticipated the fact that $\Gamma_0(p, q)$ is real-valued by only including the real part of the double integral. Assuming that the detectors are squares of width W_b , we calculate $\Gamma_0(p, q)$ as

$$\begin{aligned} \Gamma_0(p, q) &= \frac{1}{A_d^2} \int_{-W_b/2}^{W_b/2} d\rho_1^x \int_{-W_b/2}^{W_b/2} d\rho_2^x \int_{-W_b/2}^{W_b/2} d\rho_1^y \int_{-W_b/2}^{W_b/2} d\rho_2^y \text{sinc}^2 \left(\frac{(\rho_1^x - \rho_2^x)w_p}{\lambda_0 L} \right) \\ & \times \text{sinc}^2 \left(\frac{(\rho_1^y - \rho_2^y)w_p}{\lambda_0 L} \right) \cos \left[\frac{k_0}{L}(\rho_p^x - \rho_q^x)(\rho_1^x - \rho_2^x) + \frac{k_0}{L}(\rho_p^y - \rho_q^y)(\rho_1^y - \rho_2^y) \right]. \end{aligned}$$

To evaluate this expression, we move to sum and difference coordinates, $x_- = \rho_1^x - \rho_2^x$ and $y_- = \rho_1^y - \rho_2^y$, and transform the problem into a single integration in each Cartesian direction. We also split our cosine term with the angle-sum identity for cosines, $\cos(a + b) = \cos(a)\cos(b) - \sin(a)\sin(b)$. The integrations over the resulting sine terms will be the integration of an odd function over an even region, so they vanish. This allows us to decouple the integration over the different Cartesian directions so that, defining $\delta x = \rho_p^x - \rho_q^x$ and $\delta y = \rho_p^y - \rho_q^y$, we have

$$\Gamma_0(p, q) = K(\delta x)K(\delta y)$$

where

$$\begin{aligned}
K(\delta x) &= \frac{1}{A_d} \int_{-W_b}^{W_b} dx_- (W_b - |x_-|) \text{sinc}^2 \left(\frac{x_- w_p}{\lambda_0 L} \right) \cos \left[\frac{k_0}{L} \delta_x x_- \right] \\
&= \frac{2}{\mathcal{B}^2} \left[2(1 - \cos(\mathcal{B})) \cos(\mathcal{B} \delta x) + 2(\text{Ci}[\mathcal{B} \delta x] + \mathcal{B} \delta x \text{Si}[\mathcal{B} \delta x]) \right. \\
&\quad - (\text{Ci}[\mathcal{B}(\delta x - 1)] + \mathcal{B}(\delta x - 1) \text{Si}[\mathcal{B}(\delta x - 1)]) \\
&\quad \left. - (\text{Ci}[\mathcal{B}(\delta x + 1)] + \mathcal{B}(\delta x + 1) \text{Si}[\mathcal{B}(\delta x + 1)]) + \log \frac{\delta x^2}{\delta x^2 + 1} \right].
\end{aligned}$$

Here, $\mathcal{B} = W_b w_p k_0 / L$ and $\text{Si}(\cdot)$ and $\text{Ci}(\cdot)$ are the sine and cosine integrals

$$\text{Si}(x) = \int_0^x dt \frac{\sin(t)}{t} \quad (\text{A.44})$$

$$\text{Ci}(x) = - \int_x^\infty dt \frac{\cos(t)}{t}. \quad (\text{A.45})$$

In the limit $\delta x \rightarrow 0$, this solution simplifies to

$$K(0) = \frac{4}{\mathcal{B}^2} \left[(1 - \cos[\mathcal{B}]) + \gamma_0 - (\text{Ci}[\mathcal{B}] + \mathcal{B} \text{Si}[\mathcal{B}]) + \log \mathcal{B} \right],$$

where γ_0 is the Euler gamma constant.

Unfortunately, the calculations of Γ_1 , Γ_2 , and Γ_3 —all of which involve integrations over the field correlation functions—are hard for the square-pixel, square-detector geometry.

A.4.2 Approximate Noise Calculations

We have been unable to evaluate Γ_1 , Γ_2 , and Γ_3 for a square pixel. Thus we will do so using Gaussian pixels instead, for which pupil function is

$$\mathcal{A}_p(\boldsymbol{\rho}) = e^{-\frac{\pi|\boldsymbol{\rho}|^2}{2w_p^2}}. \quad (\text{A.46})$$

The width of the Gaussian distribution was chosen so that $\int d\boldsymbol{\rho} \mathcal{A}_p^2(\boldsymbol{\rho}) = w_p^2 = A_p$ preserves the pixel's intensity reflectivity. Our three remaining noise terms will eval-

uate to similar forms, and to simplify their final results we define a few additional terms,

$$\mathcal{B}_0 = \mathcal{B}/(2\pi) = W_b w_p k_0 / (L\sqrt{2\pi}) \quad (\text{A.47})$$

$$v_1 = \pi\rho_L^2 / (2w_p^2 + \pi\rho_L^2) \quad (\text{A.48})$$

$$v_2 = \pi\rho_L^2 / (w_p^2 + \pi\rho_L^2) \quad (\text{A.49})$$

$$v_3 = 2\pi\rho_L^2 / (w_p^2 + 2\pi\rho_L^2). \quad (\text{A.50})$$

Evaluating Γ_1 , with the above definitions, we find

$$\begin{aligned} \Gamma_1 &= \frac{1}{A_d^2} \int_{\mathcal{A}_b} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b} d\boldsymbol{\rho}_2 \frac{1}{A_p^2} \int d\boldsymbol{\rho}'_1 \int d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2) - \frac{|\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2|^2}{\rho_L^2}} e^{-\frac{\pi|\boldsymbol{\rho}'_1|^2}{w_p^2} - \frac{\pi|\boldsymbol{\rho}'_2|^2}{w_p^2}} \\ &= \left(\frac{e^{-v_1 \mathcal{B}_0^2} - 1 + \mathcal{B}_0 \sqrt{\pi v_1} \operatorname{erf}(\mathcal{B}_0 \sqrt{v_1})}{\mathcal{B}_0^2 \sqrt{v_1}} \right)^2, \end{aligned} \quad (\text{A.51})$$

while Γ_2 becomes

$$\begin{aligned} \Gamma_2 &= \frac{1}{A_d^2} \int_{\mathcal{A}_b} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b} d\boldsymbol{\rho}_2 \frac{1}{A_p^2} \int d\boldsymbol{\rho}'_1 \int d\boldsymbol{\rho}'_2 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2) - \frac{|\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2|^2}{2\rho_L^2}} e^{-\frac{\pi|\boldsymbol{\rho}'_1|^2}{w_p^2} - \frac{\pi|\boldsymbol{\rho}'_2|^2}{w_p^2}} \\ &= \left(\frac{e^{-v_2 \mathcal{B}_0^2} - 1 + \mathcal{B}_0 \sqrt{\pi v_2} \operatorname{erf}(\mathcal{B}_0 \sqrt{v_2})}{\mathcal{B}_0^2 \sqrt{v_2}} \right)^2, \end{aligned} \quad (\text{A.52})$$

and finally Γ_3 is

$$\begin{aligned} \Gamma_3 &= \frac{1}{A_b^2} \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_1 \int_{\mathcal{A}_b^2} d\boldsymbol{\rho}_2 \frac{1}{A_p^3} \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_1 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_2 \int_{\mathcal{A}_p^2} d\boldsymbol{\rho}'_3 e^{-\frac{ik_0}{L}(\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2) \cdot (\boldsymbol{\rho}'_1 - \boldsymbol{\rho}'_2)} \\ &\quad \times e^{-\frac{|\boldsymbol{\rho}'_3 - \boldsymbol{\rho}'_2|^2 + |\boldsymbol{\rho}'_3 - \boldsymbol{\rho}'_1|^2}{2\rho_L^2}} e^{-\frac{\pi|\boldsymbol{\rho}'_1|^2}{w_p^2} - \frac{\pi|\boldsymbol{\rho}'_2|^2}{w_p^2} - \frac{\pi|\boldsymbol{\rho}'_3|^2}{w_p^2}} \\ &= \left(\sqrt{v_2} \frac{e^{-v_3 \mathcal{B}_0^2} - 1 + \mathcal{B}_0 \sqrt{\pi v_3} \operatorname{erf}(\mathcal{B}_0 \sqrt{v_3})}{\mathcal{B}_0^2} \right)^2. \end{aligned} \quad (\text{A.53})$$

Appendix B

Useful Mathematics

In this appendix we lay out several mathematical properties that we use in the thesis. These include properties of exponential families, the Gaussian moment-factoring theorem, and several inequalities. We also sketch a derivation of the E-M algorithm.

B.1 Exponential Families

An exponential family of \mathbf{u} in terms of its natural statistic \mathbf{v} can be written as

$$p_{\mathbf{u}|\mathbf{v}}(\mathbf{u}|\mathbf{v}) = \frac{e^{f(\mathbf{u})+\mathbf{u}\mathbf{v}}}{Z(\mathbf{v})}, \quad (\text{B.1})$$

where $Z(\mathbf{v})$ is the partition function

$$Z(\mathbf{v}) = \int d\mathbf{u} e^{f(\mathbf{u})+\mathbf{u}\mathbf{v}}. \quad (\text{B.2})$$

Then, the first two derivatives of the log-partition function are

$$\frac{\partial}{\partial v} \log Z(\mathbf{v}) = \langle \mathbf{u} | \mathbf{v} \rangle \quad (\text{B.3})$$

$$\frac{\partial^2}{\partial v^2} \log Z(\mathbf{v}) = \mathbf{var}[\mathbf{u} | \mathbf{v}], \quad (\text{B.4})$$

a standard property of exponential families [57].

B.2 Gaussian Moment-Factoring Theorem

The Gaussian moment-factoring theorem allows even-order moments of a zero-mean Gaussian random vector to be expressed in terms of second-order moments [25]. It is of great use to us in Ch. 3, where we routinely encounter fourth, sixth, and even eighth-order moments of zero-mean Gaussian random fields.

For a vector $\mathbf{x} = [x_1, x_2, \dots, x_{2n}]$ comprised of $2n$ zero-mean jointly Gaussian random variables,

$$\left\langle \prod_{i=1}^{2n} x_i \right\rangle = \sum_{a \in \mathcal{A}} \prod_{i=1}^n \langle x_{a(i,1)} x_{a(i,2)} \rangle \quad (\text{B.5})$$

where a is an $n \times 2$ matrix of pair-wise permutations of the indices, and \mathcal{A} is the set of all unique pair-wise permutations. This is the sum of all unique combinations of products of n second-order moments, and for $2n$ variables there will be $(2n)!/(2^n n!)$ terms in the sum. As an example, for $n = 2$ (fourth-order moment) the first pair-wise permutation is

$$a = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad (\text{B.6})$$

while the set of allowed permutation is

$$\mathcal{A} = \left\{ \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}, \begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix} \right\} \quad (\text{B.7})$$

As an example, to achieve Eq. (3.11) we need to evaluate the fourth order moment of the transmitter's spatial pattern, which becomes

$$\begin{aligned} \langle \xi(\boldsymbol{\rho}_1, t) \xi^*(\boldsymbol{\rho}_2, t) \xi(\boldsymbol{\rho}_3, t) \xi^*(\boldsymbol{\rho}_4, t) \rangle &= \langle \xi(\boldsymbol{\rho}_1, t) \xi^*(\boldsymbol{\rho}_2, t) \rangle \langle \xi(\boldsymbol{\rho}_3, t) \xi^*(\boldsymbol{\rho}_4, t) \rangle \\ &+ \langle \xi(\boldsymbol{\rho}_1, t) \xi(\boldsymbol{\rho}_3, t) \rangle \langle \xi^*(\boldsymbol{\rho}_2, t) \xi^*(\boldsymbol{\rho}_4, t) \rangle \\ &+ \langle \xi(\boldsymbol{\rho}_1, t) \xi^*(\boldsymbol{\rho}_4, t) \rangle \langle \xi(\boldsymbol{\rho}_3, t) \xi^*(\boldsymbol{\rho}_2, t) \rangle, \end{aligned} \quad (\text{B.8})$$

where these second-order moments are evaluated using Eqs. (2.2) and (2.3).

B.3 Jensen's Inequality

Jensen's inequality has many forms, but in general it is used to upper bound the convex function of an integral with the integral of a convex function. For our purposes we used its information-theoretic expression. That is, for a convex function $f(\cdot)$,

$$f(\langle \mathbf{x} \rangle) \leq \langle f(\mathbf{x}) \rangle. \quad (\text{B.9})$$

B.4 Cauchy-Schwarz Inequality

The Cauchy-Schwarz inequality also has many expressions. Its most basic is that the square of inner product of two real-valued vectors \mathbf{x} and \mathbf{y} is less than or equal to the product of their lengths, i.e.

$$(\mathbf{x}^T \mathbf{y})^2 \leq (\mathbf{x}^T \mathbf{x})(\mathbf{y}^T \mathbf{y}). \quad (\text{B.10})$$

This inequality can also be extended to probabilistic functions. Expectation is a weighted sum, so we find that for two real-valued random variables x and y ,

$$\langle xy \rangle^2 \leq \langle x^2 \rangle \langle y^2 \rangle. \quad (\text{B.11})$$

This result can be used to show that

$$\langle x \rangle \geq \frac{1}{\langle x^{-1} \rangle}, \quad (\text{B.12})$$

by setting $y = x^{-1}$ and applying the inequality to $\langle \sqrt{xy} \rangle$. Equation (B.12) has a natural extension to random matrices, for which

$$\langle \mathbf{A} \rangle \geq \langle \mathbf{A}^{-1} \rangle^{-1}, \quad (\text{B.13})$$

implying that $\langle \mathbf{A} \rangle - \langle \mathbf{A}^{-1} \rangle^{-1}$ is positive semi-definite.

B.5 Generalized Expectation-Maximization Algorithm

The Expectation-Maximization (E-M) algorithm is an iterative algorithm that provides an estimate of some unknown parameter $\boldsymbol{\theta}$, given some measurements \mathbf{y} [58]. It alternates between making a new estimate of the parameter, $\boldsymbol{\theta}^{(t)}$, and using the new estimate to update an objective function, $U(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$, which we then maximize to find the new estimate. When $U(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is chosen appropriately, the new estimate is guaranteed to be no worse than the previous value, so that by iterating these two steps we converge to a local maximum that under certain conditions will be the global maximum, which is ML estimate¹.

The E-M algorithm is derived as follows. First, we define the log-likelihood of the joint distribution of the measurements \mathbf{y} and the unknown parameter

$$L_{\mathbf{y}}(\boldsymbol{\theta}) = \log p_{\mathbf{y},\boldsymbol{\theta}}(\mathbf{y}, \boldsymbol{\theta}). \quad (\text{B.14})$$

We then find a lower bound on the log-likelihood in terms of a previous estimate $\boldsymbol{\theta}^{(t)}$,

$$U(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \leq L_{\mathbf{y}}(\boldsymbol{\theta}). \quad (\text{B.15})$$

This bound must be constructed so that it is tight at the previous estimate, i.e.

$$U(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) = L_{\mathbf{y}}(\boldsymbol{\theta}^{(t)}). \quad (\text{B.16})$$

Then, we find the new estimate by maximizing the bound,

$$\boldsymbol{\theta}^{(t+1)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} U(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \quad (\text{B.17})$$

We are guaranteed that the likelihood of this new estimate is no worse than the

¹The E-M algorithm converges to the ML estimate when the log-likelihood is a concave function. Otherwise it will converge to a saddle point of the distribution.

likelihood of the old estimate, i.e. $L_{\mathbf{y}}(\boldsymbol{\theta}^{(t+1)}) \geq L_{\mathbf{y}}(\boldsymbol{\theta}^{(t)})$, because of the following chain of inequalities,

$$L_{\mathbf{y}}(\boldsymbol{\theta}^{(t+1)}) \geq U(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) \geq U(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) = L_{\mathbf{y}}(\boldsymbol{\theta}^{(t)}). \quad (\text{B.18})$$

In theory, any lower bound on the log-likelihood that is tight at the previous estimate can be used. We next present a derivation of the standard lower bound on the log-likelihood, and the resulting form of the E-M algorithm.

B.5.1 Derivation of Standard Lower Bound

To find an appropriate lower bound on $L_{\mathbf{y}}(\boldsymbol{\theta})$, we first find a lower bound for its difference with the log-likelihood of the previous estimate, $L_{\mathbf{y}}(\boldsymbol{\theta}^{(t)})$, i.e., we seek $\ell(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ such that

$$\ell(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \leq L(\boldsymbol{\theta}) - L(\boldsymbol{\theta}^{(t)}). \quad (\text{B.19})$$

We then define the lower bound on the log-likelihood as $U(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \ell(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) + L(\boldsymbol{\theta}^{(t)})$, since by construction we are guaranteed that $U(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \leq L(\boldsymbol{\theta})$.

The log-likelihood difference is lower-bounded by exposing a hidden variable \mathbf{x} , finding the posterior distribution of \mathbf{x} in terms of $\boldsymbol{\theta}^{(t)}$, and then using Jensen's inequality to interchange the order of the logarithms and integration so that

$$\begin{aligned} L(\boldsymbol{\theta}) - L(\boldsymbol{\theta}^{(t)}) &= \log p_{\mathbf{y},\boldsymbol{\theta}}(\mathbf{y}, \boldsymbol{\theta}) - \log p_{\mathbf{y},\boldsymbol{\theta}^{(t)}}(\mathbf{y}, \boldsymbol{\theta}^{(t)}) \\ &= \log \left[\int d\mathbf{x} p_{\mathbf{x}|\mathbf{y},\boldsymbol{\theta}}(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta}^{(t)}) \frac{p_{\mathbf{y},\mathbf{x},\boldsymbol{\theta}}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta})}{p_{\mathbf{x}|\mathbf{y},\boldsymbol{\theta}}(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta}^{(t)})} \right] - \log p_{\mathbf{y},\boldsymbol{\theta}^{(t)}}(\mathbf{y}, \boldsymbol{\theta}^{(t)}) \\ &\geq \int d\mathbf{x} p_{\mathbf{x}|\mathbf{y},\boldsymbol{\theta}^{(t)}}(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta}^{(t)}) \log \frac{p_{\mathbf{y},\mathbf{x},\boldsymbol{\theta}}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta})}{p_{\mathbf{x}|\mathbf{y},\boldsymbol{\theta}^{(t)}}(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta}^{(t)})} - \log p_{\mathbf{y},\boldsymbol{\theta}^{(t)}}(\mathbf{y}, \boldsymbol{\theta}^{(t)}) \\ &= \int d\mathbf{x} p_{\mathbf{x}|\mathbf{y},\boldsymbol{\theta}^{(t)}}(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta}^{(t)}) [\log p_{\mathbf{y},\mathbf{x},\boldsymbol{\theta}}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}) - \log p_{\mathbf{y},\mathbf{x},\boldsymbol{\theta}^{(t)}}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}^{(t)})] \\ &= \ell(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \end{aligned} \quad (\text{B.20})$$

We can immediately see that $\ell(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) = 0$, so that our lower bound is tight at

the previous estimate, i.e., $U(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) = L(\boldsymbol{\theta}^{(t)})$. Because we now have a lower bound that is tight at the previous estimate, we simply need to maximize it to find a new estimate. This optimization can be simplified as follows

$$\begin{aligned}
\boldsymbol{\theta}^{(t+1)} &= \operatorname{argmax}_{\boldsymbol{\theta}} U(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \\
&= \operatorname{argmax}_{\boldsymbol{\theta}} \int d\mathbf{x} p_{\mathbf{x}|\mathbf{y},\boldsymbol{\theta}}(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta}^{(t)}) [\log p_{\mathbf{y},\mathbf{x},\boldsymbol{\theta}}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}) - \log p_{\mathbf{y},\mathbf{x},\boldsymbol{\theta}}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}^{(t)})] + L(\boldsymbol{\theta}^{(t)}) \\
&= \operatorname{argmax}_{\boldsymbol{\theta}} \int d\mathbf{x} p_{\mathbf{x}|\mathbf{y},\boldsymbol{\theta}}(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta}^{(t)}) [\log p_{\mathbf{y},\mathbf{x},\boldsymbol{\theta}}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta})] \\
&= \operatorname{argmax}_{\boldsymbol{\theta}} \langle \log p_{\mathbf{y},\mathbf{x},\boldsymbol{\theta}}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}) | \mathbf{y}; \boldsymbol{\theta}^{(t)} \rangle_{\mathbf{x}}.
\end{aligned} \tag{B.21}$$

This final form is where we get the name ‘‘Expectation Maximization.’’ We first take the expectation of the log of the joint distribution in the expectation step, and then maximize the resulting function to find the new estimate.

The E-M algorithm can be simplified further for certain situations. In GAMP we often want the ML estimate of a scalar parameter θ of the prior distribution, (the parameter itself is treated as a deterministic). The joint distribution then factors according to $p_{\mathbf{y},\mathbf{x}}(\mathbf{y}, \mathbf{x}; \theta) = \prod_i p_{y_i, \mathbf{x}}(y_i | \mathbf{x}; \theta) \prod_j p_{\mathbf{x}}(\mathbf{x}; \theta)$, and the E-M algorithm becomes

$$\theta^{(t+1)} = \operatorname{argmax}_{\theta} \langle \log p_{\mathbf{y},\mathbf{x}}(\mathbf{x}; \theta) | \mathbf{y}; \theta^{(t)} \rangle_{\mathbf{x}}, \tag{B.22}$$

which is the update used in the EM-GAMP variant [49, 50].

B.6 Rank-One Updates

The inverse and determinant of a rank-one update have simple, closed-form solutions in terms of the inverse and determinant of the original matrix. That is, for $\mathbf{B} = \mathbf{A} + \mathbf{u}\mathbf{v}^T$, the inverse is found with the Sherman-Morrison formula [43, 59] as

$$\mathbf{B}^{-1} = (\mathbf{A} + \mathbf{u}\mathbf{v}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^T\mathbf{A}^{-1}}{1 + \mathbf{v}^T\mathbf{A}^{-1}\mathbf{u}}. \tag{B.23}$$

The determinant can then be calculated from the matrix determinant lemma [43], as

$$\det \mathbf{B} = \det(\mathbf{A} + \mathbf{u}\mathbf{v}^T) = \det(\mathbf{A}) \det(1 + \mathbf{v}^T \mathbf{A}^{-1} \mathbf{u}). \quad (\text{B.24})$$

Appendix C

Additional Derivations for GAMP

In this appendix we work through some of the details needed by the GAMP algorithm. Primarily these are a proof of our correction for non-centered matrices, as well as specific calculations for the input and output estimation functions that we use.

C.1 Correctness of Non-Centered GAMP Variant

Here we will prove the claim made in Sec. 8.2.5 that the messages from our modified GAMP algorithm are a good approximation for the true likelihood function. In this proof we are ignoring any extra measurement noise (interference term \mathbf{v}_i and measurement noise), since the issue is not estimating \mathbf{z}_i from \mathbf{y}_i , but rather estimating x_j from \mathbf{z} . We start with the log-likelihood for \mathbf{z} , given x_j , as shown in Eq. (8.77). In summation notation this becomes

$$\begin{aligned} \log p_{\mathbf{z}|\mathbf{x}}(\mathbf{z}|x_j) &= -\frac{x_j^2}{2} \sum_i \sum_k \hat{A}_{kj}(\mathbf{C}_{\mathbf{z}}^{-1})_{ki} \hat{A}_{ij} + x_j \sum_i \sum_k \hat{A}_{kj}(\mathbf{C}_{\mathbf{z}}^{-1})_{ki} (z_i - \hat{z}_i) \\ &\quad + x_j \hat{x}_j \sum_i \sum_k \hat{A}_{kj}(\mathbf{C}_{\mathbf{z}}^{-1})_{ki} \hat{A}_{ij} + \text{const}, \end{aligned} \tag{C.1}$$

where the constant term does not depend on \mathbf{x}_j . Pattern matching this to the Gaussian exponent $-(\hat{r}_j - \mathbf{x}_j)^2/2\nu_j^r$, we immediately have that

$$\hat{r}_j = \hat{x}_j + \nu_j^r \sum_i \sum_k \hat{A}_{kj} (\mathbf{C}_z^{-1})_{ki} (z_i - \hat{z}_i) \quad (\text{C.2})$$

$$\nu_j^r = \left[\sum_i \sum_k \hat{A}_{kj} (\mathbf{C}_z^{-1})_{ki} \hat{A}_{ij} \right]^{-1}. \quad (\text{C.3})$$

Our goal is then to verify that \hat{r}_j and ν_j^r as calculated above are equivalent to what is calculated by the GAMP messages. In each iteration, the combination of messages at variable node j is approximated with same Gaussian form, only the estimate and variance are calculated as

$$\hat{r}_j = \hat{x}_j + \nu_j^r \sum_{i=1}^{N+1} \hat{A}_{ij} \hat{s}_i \quad (\text{C.4})$$

$$\nu_j^r = \left[\sum_{i=1}^{N+1} \hat{A}_{ij} \nu_i^s \right]^{-1}. \quad (\text{C.5})$$

We now must show that Eqs. (C.3) and (C.5) are equivalent, and that Eqs. (C.2) and (C.4) are the same. To do this we will first decompose the message forms of \hat{r}_j and ν_j^r from Eqs. (C.4) and (C.5) as much as possible, then simplify the versions for the log-likelihood until they are equivalent.

C.1.1 Simplify Message Components

Decomposing \hat{A}_{ij} , Eqs.(C.5) and (C.4) become

$$\nu_j^r = \left[\sum_{i=1}^N \tilde{A}_{ij}^2 \nu_i^s + \epsilon_j^2 \nu_+^s \right]^{-1} \quad (\text{C.6})$$

$$\hat{r}_j = \hat{x}_j + \nu_j^r \left[\sum_{i=1}^N \tilde{A}_{ij} \hat{s}_i + \epsilon_j \hat{s}_+ \right]. \quad (\text{C.7})$$

Under our assumption that we directly measure \mathbf{z} , we have that $\hat{s}_i = (z_i - \hat{z}_i)/\tilde{\nu}_i^z$ and $\nu_i^s = \frac{1}{\tilde{\nu}_i^z}$, where $\tilde{\nu}_i^z = \sum_j \tilde{A}_{ij}^2 \nu_j^x + (\mathbf{D}_z)_{ii}$. This second component, $(\mathbf{D}_z)_{ii}$, is

the additional noise from the matrix uncertainty, as shown in Eq. (8.56). At the new factor node, where $z_+ = 0$, we have $\hat{s}_+ = -\hat{z}_+/\nu_+^z$, for $\hat{z}_+ = \sum_j \epsilon_j \hat{x}_j - \hat{x}_+$ and $\nu_+^z = \Omega$. This last identity can be inferred from Eq. (8.56). The variance at this node is then $\nu_+^z = 1/\nu_+^z = 1/\Omega$. Making these substitutions, our variance and estimate become

$$\hat{r}_j = \hat{x}_j + \nu_j^r \left[\sum_{i=1}^N \tilde{A}_{ij} \frac{(z_i - \hat{z}_i)}{\tilde{\nu}_i^z} - \epsilon_j \frac{\hat{z}_+}{\Omega} \right] \quad (\text{C.8})$$

$$\nu_j^r = \left[\sum_{i=1}^N \frac{\tilde{A}_{ij}^2}{\tilde{\nu}_i^z} + \frac{\epsilon_j^2}{\Omega} \right]^{-1}. \quad (\text{C.9})$$

C.1.2 Simplify Log-Likelihood Components

To simplify the log-likelihood forms of \hat{r}_j and ν_j^r from Eqs. (C.2) and (C.3), we need to introduce several additional approximations. First, since the GAMP algorithm presumed that approximating $\tilde{\mathbf{A}}\mathbf{C}_x\tilde{\mathbf{A}}^T$ with a diagonal matrix yields sufficiently accurate messages for the algorithm to work, we apply this same assumption to the covariance matrix in Eq. (8.56), which becomes the sum of a diagonal and rank-one matrix as

$$\mathbf{C}_z = \tilde{\mathbf{C}}_z + \bar{\mathbf{C}}_z \quad (\text{C.10})$$

where $(\tilde{\mathbf{C}}_z)_{ii} = \sum_j \tilde{A}_{ij}^2 \nu_j^x + (\mathbf{D}_z)_{ii}$ and $\bar{\mathbf{C}}_z = \boldsymbol{\mu}\boldsymbol{\mu}^T\Omega$. We note that the elements of the diagonal matrix are the variances calculated in GAMP, viz., $(\tilde{\mathbf{C}}_z)_{ii} \equiv \tilde{\nu}_i^z$.

Next, we will need several simplifications based on the strong-law of large numbers (SLLN). This law states that the sample mean of an iid random sequence converges to its ensemble mean almost surely. That is, for the iid components $\{x_n\}$, with mean \bar{x} , $\frac{1}{N} \sum_{n=1}^N x_n \xrightarrow{w.p.1} \bar{x}$, where $\xrightarrow{w.p.1}$ means convergence almost surely (convergence with probability 1) as N goes to infinity. From the SLLN we have that for two iid vectors \mathbf{x} and \mathbf{y} , where the elements of \mathbf{x} have mean \bar{x} and the elements of \mathbf{y} have mean 0, that $\sum_{n=1}^N x_n + \sum_{y=1}^N y_n \xrightarrow{w.p.1} N\bar{x}$. As a result, we can make the following

simplifications:

$$\mathbf{A}^T \tilde{\mathbf{C}}_{\mathbf{z}}^{-1} \boldsymbol{\mu}^T \xrightarrow{w.p.1} \boldsymbol{\epsilon} \boldsymbol{\mu}^T \tilde{\mathbf{C}}_{\mathbf{z}}^{-1} \boldsymbol{\mu}^T \quad (\text{C.11})$$

$$\widehat{\mathbf{A}}^T \tilde{\mathbf{C}}_{\mathbf{z}}^{-1} \boldsymbol{\mu}^T \xrightarrow{w.p.1} \boldsymbol{\epsilon} \boldsymbol{\mu}^T \tilde{\mathbf{C}}_{\mathbf{z}}^{-1} \boldsymbol{\mu}^T \quad (\text{C.12})$$

since both \mathbf{A} and its estimate $\widehat{\mathbf{A}}$ are the sum of the background term $\bar{\mathbf{A}} = \boldsymbol{\mu} \boldsymbol{\epsilon}^T$ and a zero-mean, iid matrix.

To compute Eqs. (C.3) and (C.2), we will need the inverse of the covariance matrix. Because $\mathbf{C}_{\mathbf{z}}$ is the sum of a diagonal and rank-one matrix it can be inverted using the Sherman-Morrison formula in Eq. (B.23), giving us

$$\mathbf{C}_{\mathbf{z}}^{-1} = \left[\tilde{\mathbf{C}}_{\mathbf{z}}^{-1} - \frac{\Omega \tilde{\mathbf{C}}_{\mathbf{z}}^{-1} \boldsymbol{\mu} \boldsymbol{\mu}^T \tilde{\mathbf{C}}_{\mathbf{z}}^{-1}}{1 + \Omega \boldsymbol{\mu}^T \tilde{\mathbf{C}}_{\mathbf{z}}^{-1} \boldsymbol{\mu}} \right]. \quad (\text{C.13})$$

Defining $\gamma = \Omega \boldsymbol{\mu}^T \tilde{\mathbf{C}}_{\mathbf{z}}^{-1} \boldsymbol{\mu}$, and again using Eq. (C.12) we have that

$$\widehat{\mathbf{A}}^T \mathbf{C}_{\mathbf{z}}^{-1} \xrightarrow{w.p.1} \left[\tilde{\mathbf{A}}^T + \frac{\boldsymbol{\epsilon} \boldsymbol{\mu}^T}{1 + \gamma} \right] \tilde{\mathbf{C}}_{\mathbf{z}}^{-1}. \quad (\text{C.14})$$

Now we can compute the variance from Eq. (C.3), obtaining

$$\nu_j^r = \left[\sum_i \sum_k \widehat{A}_{kj} (\mathbf{C}_{\mathbf{z}}^{-1})_{ki} \widehat{A}_{ij} \right]^{-1} \quad (\text{C.15})$$

$$\xrightarrow{w.p.1} \left[\sum_i \sum_k \left[\tilde{A}_{kj} + \epsilon_j \mu_k \frac{1}{1 + \gamma} \right] (\tilde{\mathbf{C}}_{\mathbf{z}}^{-1})_{ki} \widehat{A}_{ij} \right]^{-1} \quad (\text{C.16})$$

$$\xrightarrow{w.p.1} \left[\sum_i \frac{\tilde{A}_{kj}^2}{\tilde{\nu}_j^z} + \frac{\epsilon_j}{1 + \gamma} \sum_i \frac{\mu_i \widehat{A}_{ij}}{\tilde{\nu}_i^z} \right]^{-1} \quad (\text{C.17})$$

$$\xrightarrow{w.p.1} \left[\sum_i \frac{\tilde{A}_{kj}^2}{\tilde{\nu}_j^z} + \frac{\epsilon_j^2}{1 + \gamma} \sum_i \frac{\mu_i^2}{\tilde{\nu}_i^z} \right]^{-1}, \quad (\text{C.18})$$

Finally, we have that $\gamma = \Omega \boldsymbol{\mu}^T \tilde{\mathbf{C}}_{\mathbf{z}}^{-1} \boldsymbol{\mu} = \Omega \sum \frac{\mu_i^2}{\tilde{\nu}_i^z} \gg 1$, so $\sum \frac{\mu_i^2}{\tilde{\nu}_i^z} / (1 + \gamma) \approx 1/\Omega$, and

$$\nu_j^r \xrightarrow{w.p.1} \left[\sum_i \frac{\tilde{A}_{kj}^2}{\tilde{\nu}_j^z} + \frac{\epsilon_j^2}{\Omega} \right]^{-1}, \quad (\text{C.19})$$

which is *exactly* what was calculated for ν_j^r by the GAMP messages in Eq. (C.9).

Turning to the calculation of \hat{r}_j from Eq. (C.2), we find

$$\begin{aligned} \hat{r}_j &= \hat{x}_j + \nu_j^r \sum_i \sum_k \hat{A}_{kj} (\mathbf{C}_{\mathbf{z}}^{-1})_{ki} (z_i - \hat{z}_i) \\ &\xrightarrow{w.p.1} \hat{x}_j + \nu_j^r \sum_i \left[\tilde{A}_{ij} \frac{(z_i - \hat{z}_i)}{\tilde{\nu}_i^z} + \mu_j \epsilon_j \frac{(z_i - \hat{z}_i)}{\tilde{\nu}_i^z (1 + \gamma)} \right] \\ &\xrightarrow{w.p.1} \hat{x}_j + \nu_j^r \left[\sum_i \tilde{A}_{ij} \frac{(z_i - \hat{z}_i)}{\tilde{\nu}_i^z} + \epsilon_j \sum_i \frac{\mu_i (z_i - \hat{z}_i)}{\tilde{\nu}_i^z (1 + \gamma)} \right]. \end{aligned} \quad (\text{C.20})$$

Recognizing that $z_i = \sum_{ij} A_{ij} x_j$ and $\hat{z}_i = \sum_{ij} A_{ij} \hat{x}_j$, this becomes

$$\begin{aligned} \hat{r}_j &\xrightarrow{w.p.1} \hat{x}_j + \nu_j^r \sum_i \tilde{A}_{ij} \frac{(z_i - \hat{z}_i)}{\tilde{\nu}_i^z} + \nu_j^r \epsilon_j \sum_i \frac{\mu_i}{\tilde{\nu}_i^z} \sum_{\ell} A_{i\ell} \frac{(x_{\ell} - \hat{x}_{\ell})}{(1 + \gamma)} \\ &\xrightarrow{w.p.1} \hat{x}_j + \nu_j^r \sum_i \tilde{A}_{ij} \frac{(z_i - \hat{z}_i)}{\tilde{\nu}_i^z} + \nu_j^r \epsilon_j \sum_i \frac{\mu_i^2}{\tilde{\nu}_i^z (1 + \gamma)} \sum_{\ell} (\epsilon_{\ell} x_{\ell} - \epsilon_{\ell} \hat{x}_{\ell}) \\ &\xrightarrow{w.p.1} \hat{x}_j + \nu_j^r \sum_i \tilde{A}_{ij} \frac{(z_i - \hat{z}_i)}{\tilde{\nu}_i^z} + \nu_j^r \frac{\epsilon_j}{\Omega} \sum_{\ell} (\epsilon_{\ell} x_{\ell} - \epsilon_{\ell} \hat{x}_{\ell}). \end{aligned} \quad (\text{C.21})$$

Finally, re-introducing our auxiliary variable $\hat{x}_+ = \boldsymbol{\epsilon}^T \mathbf{x}$, the estimate $z_+ = \boldsymbol{\epsilon}^T \hat{\mathbf{x}} - \hat{x}_+$, gives us our final form

$$\hat{r}_j \xrightarrow{w.p.1} \hat{x}_j + \nu_j^r \left[\sum_i \tilde{A}_{ij} \frac{(z_i - \hat{z}_i)}{\tilde{\nu}_i^z} - \epsilon_j \frac{\hat{z}_+}{\Omega} \right], \quad (\text{C.22})$$

which is *exactly* what we calculated for the estimate using the GAMP messages in Eq. (C.8).

In this derivation we showed, under some mild assumptions and simplifications based on the strong law of large numbers, that the log-likelihood of \mathbf{z} given x_j is accurately approximated by the combination of messages at the variable node in our

modified GAMP algorithm. We therefore expect our modification will allow GAMP to work when the measurement matrix is non-centered.

C.2 Input Estimators

We used three GAMP input estimators in this thesis, each associated with a different prior. The first was a Gaussian distribution, used for MMSE estimation of the spatial-basis pixels. Second was a Laplace distribution, which we used for MMSE estimation of the wavelet-basis coefficients. For this prior we also derived an adaptation function to learn its rate parameter. Finally, we used a gamma distribution for MAP estimation of the speckle. For a derivation of GAMP, see Ch. 8. As a reminder, we will be computing quantities based on the posterior distribution

$$p_{\mathbf{x}|r}(x_j|\hat{r}_j; \theta) = \frac{p_{r|\mathbf{x}}(\hat{r}_j|x_j)p_{\mathbf{x}}(x_j; \theta)}{p_r(\hat{r}_j)}, \quad (\text{C.23})$$

where $p_{\mathbf{x}}(x_j; \theta)$ is the prior distribution for \mathbf{x}_j , and $p_{r|\mathbf{x}}(\hat{r}_j|x_j)$ is the Gaussian likelihood

$$p_{r|\mathbf{x}}(\hat{r}_j|x_j) = \frac{\exp\left(\frac{-(\hat{r}_j - x_j)^2}{2\nu_j^r}\right)}{\sqrt{2\pi\nu_j^r}} \quad (\text{C.24})$$

formed by the returning messages from the factor nodes. We have dropped the iteration index t that appears in the original definitions since these computations have identical forms for each iteration. The estimation functions we will be calculating are specified in Tab. 8.1. For MMSE estimation, they are

$$\hat{x}_j = \langle \mathbf{x}_j | \hat{r}_j, \nu_j^r; \theta \rangle = \int dx_j x_j p_{\mathbf{x}|r}(x_j | \hat{r}_j; \theta) \quad (\text{C.25})$$

$$\nu_j^x = \mathbf{var}[\mathbf{x}_j | \hat{r}_j, \nu_j^r; \theta] = \int dx_j x_j^2 p_{\mathbf{x}|r}(x_j | \hat{r}_j; \theta) - \hat{x}_j^2. \quad (\text{C.26})$$

while for MAP estimation, the calculation in Tab. 8.1 reduce to

$$\hat{x}_j = \underset{x_j}{\operatorname{argmax}} [\log p_{x|r}(x_j|\hat{r}_j; \theta)] \quad (\text{C.27})$$

$$\nu_j^x = \frac{\nu_j^r}{1 - \nu_j^r \frac{\partial^2 \log p_x(\hat{x}_j; \theta)}{\partial \hat{x}_j^2}}. \quad (\text{C.28})$$

C.2.1 Gaussian Prior for MMSE Estimation

A Gaussian distribution has two parameters, its means and variance. Therefore, setting $\theta = (\bar{x}, \sigma_x^2)$, the probability density function of the prior is

$$p_x(x_j; \theta) = \frac{e^{-\frac{(x_j - \bar{x})^2}{2\sigma_x^2}}}{\sqrt{2\pi\sigma_x^2}}, \quad (\text{C.29})$$

the posterior mean and variance are

$$\hat{x}_j = \frac{\sigma_x^2 \hat{r}_j + \nu_j^r \bar{x}}{\sigma_x^2 + \nu_j^r} \quad (\text{C.30})$$

$$\nu_j^x = \frac{\nu_j^r \sigma_x^2}{\sigma_x^2 + \nu_j^r}. \quad (\text{C.31})$$

Laplace distribution

The Laplace distribution is the double sided exponential distribution, that for inverse-scale parameter θ is

$$p_x(x_j; \theta) = \frac{\theta}{2} e^{-\theta|x_j|}. \quad (\text{C.32})$$

The posterior mean and variance of under this prior distribution are

$$\hat{x}_j = \sqrt{2\nu_j^r} \left[\frac{be^{b^2} \operatorname{efrc}(b_j) - ae^{a^2} \operatorname{efrc}(a_j) - 2}{e^{b^2} \operatorname{efrc}(b_j) + e^{a^2} \operatorname{efrc}(a_j)} \right] \quad (\text{C.33})$$

$$\nu_j^x = \nu_j^r \left[\frac{(1 + 2b^2)e^{b^2} \operatorname{efrc}(b_j) + (1 + 2a^2)e^{a^2} \operatorname{efrc}(a_j) - \theta 2\sqrt{\frac{2\nu_j^r}{\pi}}}{e^{b^2} \operatorname{efrc}(b_j) + e^{a^2} \operatorname{efrc}(a_j)} \right]. \quad (\text{C.34})$$

where a_j and b_j are

$$a_j = \frac{\theta \nu_j^r - \widehat{r}_j}{\sqrt{2\nu_j^r}} \quad (\text{C.35})$$

$$b_j = \frac{\theta \nu_j^2 + \widehat{r}_j}{\sqrt{2\nu_j^r}}, \quad (\text{C.36})$$

and erfc is the complementary error function.

Update of Prior Parameter

In Sec. 9.2 did not model the wavelet coefficients with an iid distribution, but rather with an iid distribution at each level of the decomposition. This is because it is known that the variance goes down about an order of magnitude at each level in the wavelet decomposition. For \mathcal{J}_ℓ being the set of coefficients j in level ℓ , we define our prior in terms of a parameter θ_ℓ as

$$p_{\mathbf{x}}(x_j; \theta_\ell) = \frac{\theta_\ell}{2} e^{-\theta_\ell |x_j|}, \quad \text{for } j \in \mathcal{J}_\ell \quad (\text{C.37})$$

For our adaptive step we use the E-M approach, shown in Eq. (9.4). The expectation part of the update for θ_ℓ , operating on indices \mathcal{J}_ℓ , is

$$\langle \log p_{\mathbf{x}}(\mathbf{x}_{\mathcal{J}_\ell}; \theta_\ell) | \widehat{r}_j; \nu_j^r, q'_\ell, q_f \rangle = \sum_{j \in \mathcal{J}_\ell} \left[\log \frac{\theta_\ell}{2} - \theta_\ell \langle |x_j| | \widehat{r}_j; \nu_j^r, \theta_\ell \rangle \right]. \quad (\text{C.38})$$

To find the maximum of this function we first take the derivative of Eq. (C.38) with respect to θ_ℓ and set it equal to 0, yielding

$$\sum_{j \in \mathcal{J}_\ell} \left[\frac{1}{\theta_\ell} - \varpi_j \right] = 0, \quad (\text{C.39})$$

where

$$\varpi_j = \langle |x_j| | \widehat{r}_j; \nu_j^r, \theta_\ell \rangle = \int dx_j p_{\mathbf{x}|r}(x_j | \widehat{r}_j; \nu_j^r, \theta_\ell) |x_j|. \quad (\text{C.40})$$

The parameter update is then

$$\theta_\ell = \left[\frac{1}{|\mathcal{J}_\ell|} \sum_{j \in \mathcal{J}_\ell} \varpi_j \right]^{-1}. \quad (\text{C.41})$$

To verify that this is at a maximum and not a minimum, we check that the second derivative of Eq. (C.38) is negative, which we see from

$$\frac{\partial^2}{\partial \theta_\ell^2} \langle \log p_{\mathbf{x}}(\mathbf{x}_{\mathcal{J}_\ell}; \theta_\ell) | \hat{r}_j; \nu_j^r, q_\ell', q_f \rangle = -\frac{|\mathcal{J}_\ell|}{\theta_\ell^2} < 0, \quad (\text{C.42})$$

since $\theta_\ell > 0$.

C.2.2 Gamma Prior for MAP Estimate

The gamma distribution is a function of a shape parameter k , and a scale parameter θ , and has the probability density function

$$p_{\mathbf{x}}(x_j) = \begin{cases} \frac{x_j^{k-1} e^{-x_j/\theta}}{\theta^k \Gamma(k)}, & x > 0, k > 0, \theta > 0 \\ 0 & \text{otherwise} \end{cases}, \quad (\text{C.43})$$

where $\Gamma(k)$ is the gamma function. The estimate is computed as

$$\hat{x} = \operatorname{argmax}_x \left[\log p_{\mathbf{x}}(x_j) - \frac{(\hat{r}_j - x_j)^2}{2\nu_j^r} \right] \quad (\text{C.44})$$

$$= \operatorname{argmax}_x \left[(k-1) \log x_j - \frac{x_j}{\theta} - \frac{(\hat{r}_j - x_j)^2}{2\nu_j^r} \right]. \quad (\text{C.45})$$

We find the local extrema by setting the derivative with respect to x_j equal to 0, as

$$\frac{\partial}{\partial x_j} \left[(k-1) \log x_j - \frac{x_j}{\theta} - \frac{(\hat{r}_j - x_j)^2}{2\nu_j^r} \right] = \frac{k-1}{x_j} - \frac{1}{\theta} - \frac{(x_j - \hat{r}_j)}{\nu_j^r} = 0 \quad (\text{C.46})$$

Rearranging, this is a quadratic equation in x_j , so the solution is

$$\hat{x}_j = \frac{-b + \sqrt{b^2 - 4ac}}{2a}. \quad (\text{C.47})$$

where $a = \theta$, $b = \nu_j^r - \hat{r}_j\theta$, and $c = (1 - k)\nu_j^r\theta$. To confirm this is a local maximum, we check that the second derivative is negative, viz.,

$$\frac{\partial^2}{\partial x_j^2} \left[(k - 1) \log x_j - \frac{x_j}{\theta} - \frac{(\hat{r}_j - x_j)^2}{2\nu_j^r} \right] = \frac{1 - k}{x_j^2} - \frac{1}{\nu_j^r}. \quad (\text{C.48})$$

Since $\nu_j^r > 0$, if $k \geq 1$, the second derivative is negative. Since $k = 1$ corresponds to the smallest possible detector (for which the intensity is an exponential distribution), and k grows as the detector size increases, we are guaranteed that $k \geq 1$, and we are finding a maximum.

To compute ν_j^x as in Eq. (C.28) we need to find

$$\frac{\partial^2 \log p_{\times}(\hat{x}_j; \theta)}{\partial \hat{x}_j^2} = \frac{1 - k}{\hat{x}_j^2}, \quad (\text{C.49})$$

from which we can compute

$$\nu_j^x = \frac{\nu_j^r}{1 - \nu_j^r \frac{1-k}{\hat{x}_j^2}}. \quad (\text{C.50})$$

C.3 Output Estimators

We modeled photodetection shot-noise with a Poisson process. However, we are operating in the high-photon count region for which the Poisson distribution converges to a Gaussian distribution. We developed estimators for both of these distributions, but because the Poisson estimator was much more computationally expensive to compute we used the Gaussian estimator, which for our simulation gave identical reconstructions. Since the measurements are used for both the MMSE estimation of the target, as well as the MAP estimation of the speckle, we compute \hat{s}_i and ν_i^s for both situations. However, the additive white-noise Gaussian measurement channel is the special case for which \hat{s}_i and ν_i^s are the same for both MMSE and MAP estimation

As a reminder, we will be computing quantities based on the posterior distribution

$$p_{z|y}(z_i|y_i) = \frac{p_{y|z}(y_i|z_i)p_z(z_i)}{p_y(y_i)}. \quad (\text{C.51})$$

where $p_{y|z}(y_i|z_i)$ is the measurement distribution, and $p_z(z_i)$ is the distribution from Eq. (8.7) that we reproduce here,

$$p_z(z_i) = \frac{e^{-(z_i - \hat{p}_i)^2 / 2\nu_i^p}}{\sqrt{2\pi\nu_i^p}}. \quad (\text{C.52})$$

As with the derivation of input estimators, we are dropping the iteration index t since these computations happen for each iteration. The estimation functions we will be calculating are specified in Tab. 8.1. For MMSE estimation, we need to compute

$$\hat{s}_i = \frac{\langle z_i | \hat{p}_i, \nu_i^p, y_i \rangle - \hat{p}_i}{\nu_i^p} \quad (\text{C.53})$$

$$\nu_i^s = \frac{1}{\nu_i^p} \left[1 - \frac{\mathbf{var}[z | \hat{p}_i, \nu_i^p, y_i]}{\nu_i^p} \right], \quad (\text{C.54})$$

where

$$\langle z_i | \hat{p}_i, \nu_i^p, y_i \rangle = \int dz_i z_i p_{z|y}(z_i|y_i) \quad (\text{C.55})$$

$$\mathbf{var}[z_i | \hat{p}_i, \nu_i^p, y_i] = \int dz_i z_i^2 p_{z|y}(z_i|y_i) - \hat{z}_i^2. \quad (\text{C.56})$$

For MAP estimation, the calculations in Tab. 8.1 become

$$\hat{s}_i = \frac{\hat{z}_i^0 - \hat{p}_i}{\nu_i^p} \quad (\text{C.57})$$

$$\nu_i^s = \frac{\frac{\partial^2}{\partial(\hat{z}_i^0)^2} \log p_{y|z}(y_i|\hat{z}_i^0)}{\nu_i^p \frac{\partial^2}{\partial(\hat{z}_i^0)^2} \log p_{y|z}(y_i|\hat{z}_i^0) - 1}, \quad (\text{C.58})$$

where

$$\hat{z}_i^0 = \underset{z_i}{\operatorname{argmax}} [\log p_{z|y}(z_i|y_i)]. \quad (\text{C.59})$$

C.3.1 Gaussian Measurement Channel

For a measurement channel comprised of zero-mean, additive Gaussian noise, the measurement distribution is

$$p_{y|z}(y_i|z_i) = \frac{\exp\left(-\frac{(y_i-z_i)^2}{2\nu_i^y}\right)}{\sqrt{2\pi\nu_i^y}}, \quad (\text{C.60})$$

where ν_i^y is the variance of the additive noise.

MMSE estimation

For this distribution the required computations for MMSE estimation are

$$\langle z_i | \hat{p}_i, \nu_i^p, y_i \rangle = \frac{\nu_i^y \hat{p}_i + \nu_i^p y_i}{\nu_i^y + \nu_i^p} \quad (\text{C.61})$$

$$\text{var}[z_i | \hat{p}_i, \nu_i^p, y_i] = \frac{\nu_i^y \nu_i^p}{\nu_i^y + \nu_i^p}. \quad (\text{C.62})$$

MAP Estimate

For a Gaussian distribution, \hat{s}_i and ν_i^s are the same for MMSE and MAP estimation. To see this we first take the maximization in Eq. (C.59), which becomes

$$\hat{z}_i^0 = \frac{\nu_i^y \hat{p}_i + \nu_i^p y_i}{\nu_i^y + \nu_i^p}. \quad (\text{C.63})$$

The computation for \hat{s}_i is the same for MMSE and MAP estimation (Eqs. (C.53) and (C.57), respectively), but with \hat{z}_i^0 replacing $\langle z_i | \hat{p}_i, \nu_i^p, y_i \rangle$ in the MAP version. Since we find $\hat{z}_i^0 = \langle z_i | \hat{p}_i, \nu_i^p, y_i \rangle$, both methods yield the same computation of \hat{s}_i .

To compare the computations of ν_i^s for the two methods, we take the second derivative

$$\frac{\partial^2}{\partial (\hat{z}_i^0)^2} \log p_{y|z}(y_i | \hat{z}_i^0) = -\frac{1}{\nu_i^y}, \quad (\text{C.64})$$

which is required to compute ν_i^s in Eq. (C.58) for MAP estimation, so that

$$\nu_i^s = \frac{-\frac{1}{\nu_i^y}}{-\frac{\nu_i^p}{\nu_i^y} - 1} = \frac{1}{\nu_i^y + \nu_i^p}. \quad (\text{C.65})$$

For comparison, we compute ν_i^s for MMSE estimation using Eqs. (C.54) and (C.62) as

$$\nu_i^s = \frac{1}{\nu_i^p} \left[1 - \frac{\nu_i^y \nu_i^p}{\nu_i^y + \nu_i^p} \right] = \frac{1}{\nu_i^y + \nu_i^p}, \quad (\text{C.66})$$

which is the same result seen for MAP estimation.

Bibliography

- [1] R. M. Marino and W. R. Davis, “Jigsaw: a foliage-penetrating 3d imaging laser radar system,” *Lincoln Lab. J* **15**, 23 (2005).
- [2] N. D. Hardy, *Analyzing and improving image quality in reflective ghost imaging*, S.M. thesis, Massachusetts Institute of Technology (2011).
- [3] N. D. Hardy and J. H. Shapiro, “Ghost imaging in reflection: Resolution, contrast, and signal-to-noise,” *Proc. SPIE*, vol. 7815, 78150L (2010).
- [4] N. D. Hardy and J. H. Shapiro, “Reflective ghost imaging through turbulence,” *Phys. Rev. A* **84**, 063824 (2011).
- [5] N. D. Hardy and J. H. Shapiro, “Computational ghost imaging versus imaging laser radar for three-dimensional imaging,” *Phys. Rev. A* **87**, 023820 (2013).
- [6] K. Khoshelham and S. O. Elberink, “Accuracy and resolution of kinect depth data for indoor mapping applications.” *Sensors (Basel)*. **12**, 1437 (2012).
- [7] T. Pittman, Y. Shih, D. Strekalov, and A. V. Sergienko, “Optical imaging by means of two-photon quantum entanglement,” *Phys. Rev. A* **52**, R3429 (1995).
- [8] F. Ferri, D. Magatti, A. Gatti, M. Bache, E. Brambilla, and L. Lugiato, “High-resolution ghost image and ghost diffraction experiments with thermal light,” *Phys. Rev. Lett.* **94**, 183602 (2005).
- [9] A. Valencia, G. Scarcelli, M. D’Angelo, and Y. Shih, “Two-photon imaging with thermal light,” *Phys. Rev. Lett.* **94**, 063601 (2005).

- [10] G. Scarcelli, V. Berardi, and Y. Shih, “Can two-photon correlation of chaotic light be considered as correlation of intensity fluctuations?” *Phys. Rev. Lett.* **96**, 063602 (2006).
- [11] B. I. Erkmen and J. H. Shapiro, “Unified theory of ghost imaging with Gaussian-state light,” *Phys. Rev. A* **77**, 043809 (2008).
- [12] B. I. Erkmen and J. H. Shapiro, “Signal-to-noise ratio of Gaussian-state ghost imaging,” *Phys. Rev. A* **79**, 023833 (2009).
- [13] J. H. Shapiro, “Imaging and optical communication through atmospheric turbulence,” J. W. Strohbern, editor, *Laser Beam Propagation in the Atmosphere*, chap. 6 (Springer-Verlag, Berlin, 1978).
- [14] J. Cheng, “Ghost imaging through turbulent atmosphere,” *Opt. Express* **17**, 7916 (2009).
- [15] H. L. Van Trees, *Detection, Estimation, and Modulation Theory* (John Wiley & Sons, 1968).
- [16] D. L. Donoho, A. Maleki, and A. Montanari, “Message-passing algorithms for compressed sensing,” *Proc. Natl. Acad. Sci. U. S. A.* **106**, 18914 (2009).
- [17] S. Rangan, “Generalized approximate message passing for estimation with random linear mixing,” *Proc. IEEE Int. Symp. Inf. Theory*, 2168–2172 (IEEE, 2011).
- [18] J. H. Shapiro, “Computational ghost imaging,” *Phys. Rev. A* **78**, 061802 (2008).
- [19] V. I. Tatarskii, A. Ishimaru, and V. U. Zavorotn, editors, *Wave Propagation in Random Media (Scintillation)* (The International Society for Optical Engineering, Bellingham, 1992).
- [20] A. Ishimaru, *Wave Propagation and Scattering in Random Media*, vol. 2 (Academic Press, New York, 1978).
- [21] J. W. Goodman, *Speckle Phenomena in Optics: Theory and Applications* (Roberts & Co., Englewood, 2007).

- [22] J. H. Shapiro, B. A. Capron, and R. C. Harney, "Imaging and target detection with a heterodyne-reception optical radar." *Appl. Opt.* **20**, 3292 (1981).
- [23] G. R. Osche, *Optical Detection Theory for Laser Applications* (John Wiley & Sons, Hoboken, New Jersey, 2002).
- [24] J. W. Goodman, *Statistical Optics* (Wiley-Interscience, New York, 1985).
- [25] L. Mandel and E. Wolf, *Optical Coherence and Quantum Optics* (Cambridge University Press, 1995).
- [26] P. B. Dixon, G. A. Howland, K. W. C. Chan, C. O'Sullivan-Hale, B. Rodenburg, N. D. Hardy, J. H. Shapiro, D. S. Simon, A. V. Sergienko, R. W. Boyd, and J. C. Howell, "Quantum ghost imaging through turbulence," *Phys. Rev. A* **83**, 051803 (2011).
- [27] B. I. Erkmen, "Computational ghost imaging for remote sensing," *J. Opt. Soc. Am. A* **29**, 782 (2012).
- [28] B. I. Erkmen and J. H. Shapiro, "Ghost imaging: from quantum to classical to computational," *Adv. Opt. Photonics* **2**, 405 (2010).
- [29] D. Venkatraman, N. D. Hardy, F. N. C. Wong, and J. H. Shapiro, "Classical far-field phase-sensitive ghost imaging." *Opt. Lett.* **36**, 3684 (2011).
- [30] B. I. Erkmen, N. D. Hardy, D. Venkatraman, F. N. C. Wong, and J. H. Shapiro, "Phase-sensitive coherence and the classical-quantum boundary in ghost imaging," *Proc. SPIE*, vol. 8122, 81220M (2011).
- [31] Y. Bromberg, O. Katz, and Y. Silberberg, "Ghost imaging with a single detector," *Phys. Rev. A* **79**, 053840 (2009).
- [32] O. Katz, Y. Bromberg, and Y. Silberberg, "Compressive ghost imaging," *Appl. Phys. Lett.* **95**, 131110 (2009).
- [33] R. E. Meyers, K. S. Deacon, and Y. Shih, "Ghost-imaging experiment by measuring reflected photons," *Phys. Rev. A* **77**, 041801 (2008).

- [34] R. E. Meyers and K. S. Deacon, “Quantum ghost imaging experiments at ARL,” Proc. SPIE, vol. 7815, 78150I (2010).
- [35] C. Zhao, W. Gong, M. Chen, E. Li, H. Wang, W. Xu, and S. Han, “Ghost imaging lidar via sparsity constraints,” Appl. Phys. Lett. **101**, 141123 (2012).
- [36] F. Ferri, D. Magatti, L. A. Lugiato, and A. Gatti, “Differential ghost imaging,” Phys. Rev. Lett. **104**, 253603 (2010).
- [37] B. Sun, S. S. Welsh, M. P. Edgar, J. H. Shapiro, and M. J. Padgett, “Normalized ghost imaging,” Opt. Express **20**, 16892 (2012).
- [38] E. J. Candes, J. Romberg, and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” IEEE Trans. Inf. Theory **52**, 489 (2006).
- [39] D. L. Donoho, “Compressed sensing,” IEEE Trans. Inf. Theory **52**, 1289 (2006).
- [40] E. J. Candes and T. Tao, “Near-optimal signal recovery from random projections: Universal encoding strategies?” IEEE Trans. Inf. Theory **52**, 5406 (2006).
- [41] *6.437 Inference and Information Lecture Notes* (MIT, 2013).
- [42] “Microsoft Research Cambridge Object Recognition Image Database, version 1.0.” .
- [43] K. B. Petersen and M. S. Pedersen, “The matrix cookbook,” Technical University of Denmark (2012).
- [44] B. Frey and D. MacKay, “A revolution: Belief propagation in graphs with cycles,” Proc. Neural Inform. Process. Syst. Conf., 479–485 (1997).
- [45] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference* (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988).
- [46] Y. Weiss and W. T. Freeman, “Correctness of belief propagation in Gaussian graphical models of arbitrary topology,” Neural Comput. **13**, 2173 (2001).

- [47] S. Rangan, A. K. Fletcher, V. K. Goyal, and P. Schniter, “Hybrid generalized approximate message passing with applications to structured sparsity,” Proc. IEEE Int. Symp. Inf. Theory, 1236–1240 (IEEE, 2012).
- [48] J. T. Parker, V. Cevher, and P. Schniter, “Compressive sensing under matrix uncertainties: An Approximate Message Passing approach,” Conf. Rec. Forty Fifth Asilomar Conf. Signals, Syst. Comput., 804–808 (IEEE, 2011).
- [49] J. Vila and P. Schniter, “Expectation-maximization Bernoulli-Gaussian approximate message passing,” Conf. Rec. Forty Fifth Asilomar Conf. Signals, Syst. Comput., 799–803 (IEEE, 2011).
- [50] J. Vila and P. Schniter, “Expectation-maximization Gaussian-mixture approximate message passing,” 46th Annu. Conf. Inf. Sci. Syst., 1–6 (IEEE, 2012).
- [51] U. S. Kamilov, S. Rangan, A. K. Fletcher, and M. Unser, “Approximate message passing with consistent parameter estimation and applications to sparse learning,” F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Adv. Neural Inf. Process. Syst. 25*, 2438–2446 (Curran Associates, Inc., 2012).
- [52] J. Romberg, “Imaging via compressive sampling,” IEEE Signal Process. Mag. **25**, 14 (2008).
- [53] S. Mallat, *A Wavelet Tour of Signal Processing* (Academic Press, San Diego, CA, 1999), 2nd ed.
- [54] I. Daubechies, *Ten Lectures on Wavelets* (Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992).
- [55] J. Kovačević, V. K. Goyal, and M. Vetterli, *Fourier and Wavelet Signal Processing* (Cambridge University Press, Cambridge, UK, 2015 (anticipated)).
- [56] S. Som and P. Schniter, “Compressive imaging using approximate message passing and a markov-tree prior,” IEEE Trans. Signal Process. **60**, 3439 (2012).

- [57] M. J. Wainwright and M. I. Jordan, “Graphical models, exponential families, and variational inference,” *Found. Trends Mach. Learn.* **1**, 1 (2007).
- [58] A. Dempster, N. Laird, and D. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *J. R. Stat. Soc. Ser. B* **39**, 1 (1977).
- [59] M. S. Bartlett, “An inverse matrix adjustment arising in discriminant analysis,” *Ann. Math. Stat.* **22**, 107 (1951).

Acronyms

BCRB Bayesian Cramér-Rao bound.

CLT Central Limit Theorem.

CRB Cramér-Rao bound.

E-M Expectation-Maximization.

GAMP generalized approximate message-passing.

iid independent, identically distributed.

LBP loopy belief propagation.

LMMSE linear minimum mean-squared error.

MAP maximum a posteriori probability.

MMSE minimum mean-squared error.

MSE mean-squared error.

LLN strong-law of large numbers.

SLM spatial light modulator.

SNR signal-to-noise ratio.

SPDC spontaneous parametric downconversion.

Glossary

Optical Propagation and Reflection

a_0 intensity radius of field at transmitter. 25

a_L intensity radius of field at range L . 25

c speed of light. 24

$E_m(\boldsymbol{\rho}, t)$ complex field at the beginning of path m . 26

$E_m(\boldsymbol{\rho}, t, L)$ complex field at range L on path m . 26

$E_p(\boldsymbol{\rho}, t)$ complex field impinging pixel p of LADAR CCD array. 53

$\xi(\boldsymbol{\rho}, n)$ normalized spatial mode of field at transmitter for pulse n . 24

$\xi_b(\boldsymbol{\rho}, n, k)$ returned field spatial mode for pulse n , range k (k often omitted). 33

$\xi_p(\boldsymbol{\rho}, n, k)$ field spatial mode at pixel p of LADAR CCD array for pulse n , range k (k often omitted). 54

$\xi_t(\boldsymbol{\rho}, n, k)$ on-target field spatial mode for pulse n , range k (k often omitted). 33

$h_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ extended Huygens-Fresnel propagation kernel on path m . 26

k_0 wavenumber at wavelength λ_0 . 24

L distance from the transmitter to the target. 25

λ_0 wavelength of light, equal to $2\pi c/\omega_0$. 24

N number of transmitted pulses/number of measurements. 24

\mathbb{N}_s number of transmitted photons per pulse. 24

ρ_0 coherence length of field at transmitter. 25

ρ_L coherence length of field at range L . 25

ϱ_T coherence length of the rough surface. 29

$T(\boldsymbol{\rho})$ field reflection coefficient for the rough surface. 29

$\mathcal{T}(\boldsymbol{\rho})$ average intensity reflection coefficient for the rough surface. 29

$\vartheta(\boldsymbol{\rho})$ contribution to the target's field-reflection coefficient from the surface roughness. 29

τ_p pulse duration of transmitted field. 24

τ_s pulse repetition interval of the transmitter. 24

w_0 beam waist of laser at transmitter. 52

w_L beam waist of laser at range L . 55

ω_0 center frequency of the propagating light wave. 24

Turbulence

$\chi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ log-amplitude of the turbulence fluctuations on path m . 26, 47

$C_{n,m}^2(z)$ turbulence strength on path m at range z . 27

$\mathcal{C}_{n,m}^2(z)$ normalized turbulence strength on path m at range z . 28

$D_m(\boldsymbol{\rho}, \boldsymbol{\rho}')$ turbulence structure function on path m between $\boldsymbol{\rho}$ and $\boldsymbol{\rho}'$. 27

$\phi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ phase of the turbulence fluctuations on path m . 27

ϱ_m coherence length of the turbulence structure function on path m . 28

Ψ averaging function for time-varying turbulence fluctuations. 48

$\psi_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ contribution of turbulence to the propagation kernel $h_m(\boldsymbol{\rho}, \boldsymbol{\rho}', t)$ on path m . 26

τ_ψ coherence time of the turbulence fluctuations. 33

ζ_m output-plane weighting coefficient for the square-law approximation to the turbulence structure function on path m . 28

ζ'_m input-plane weighting coefficient for the square-law approximation to the turbulence structure function on path m . 28

Photodetection

A_b area of the bucket detector. 44

A_ℓ area of the receiver lens for LADAR. 55

A_p area of pixel; used for both LADAR CCD pixels, and later for on-target pixels in the discretized model. 56

\mathcal{A}_b field-pupil function for the bucket detector. 32

\mathcal{A}_ℓ field-pupil function for the lens used in LADAR. 53

\mathcal{A}_p field-pupil function for a pixel; used for both LADAR CCD pixels, and later for on-target pixels in the discretized model. 73
 η photodetection quantum efficiency. 32
 $g(t)$ matched filter for transmitted waveform. 32
 $i_b(n, k)$ sample (n, k) of the bucket detector photocurrent, (k often omitted). 34
 $i_b(t)$ photocurrent produced by the bucket detector at time t . 32
 $i_p(n, k)$ sample (n, k) of CCD pixel p 's photocurrent, (k often omitted). 54
 $\Delta i_b(n, k)$ filtered bucket-detector shot noise for sample (n, k) , (k often omitted). 34
 $\Delta i_b(t)$ bucket-detector shot noise at time t . 32
 $\Delta i_p(n, k)$ filtered pixel- p shot noise for sample (n, k) , (k often omitted). 54
 $\mathbb{N}_b(n, k)$ number of photons received by the bucket detector for pulse n and range k , (k often omitted). 34
 $\mathbb{N}_p(n, k)$ number of photons received at pixel p of LADAR CCD array for pulse n and range k , (k often omitted). 54
 q electron charge. 32
 r_b radius of the bucket detector. 48
 r_ℓ radius of the receiver lens for LADAR. 53
 r_p radius of a LADAR CCD pixel. 56
 τ'_p effective pulse width with matched-filter detection. 34
 τ''_p effective shot-noise pulse width for matched-filter detection. 34

Ghost Imaging and LADAR

α turbulence degradation factor for ghost imaging. 44
 α' turbulence degradation factor for LADAR. 55
 A_T measure of the target's reflective area. 48
 A'_T another measure of the target's reflective area. 48
 β ratio of detector's and transmitter's pupil areas. 48
 β' ratio of the CCD pixel's area and the lens PSF's area. 56
 $\Delta^2 \mathcal{D}$ noise term in the ghost imaging SNR analysis. 49
 $\Delta^2 \mathcal{F}$ noise term in the ghost imaging SNR analysis. 49

$\Delta^2\mathcal{R}$ noise term in the ghost imaging SNR analysis. 49
 $\Delta^2\mathcal{S}$ noise term in the ghost imaging SNR analysis. 49
 $\mathcal{G}_N(\boldsymbol{\rho})$ N -pulse ghost image at location $\boldsymbol{\rho}$. 41
 $\tilde{i}_b(n)$ empirically-centered bucket-detector measurement vector. 41
 $I(\boldsymbol{\rho}, n)$ continuous-space reference pattern for pulse n . 41
 $\Delta I(\boldsymbol{\rho}, n)$ statistically-centered continuous-space reference pattern. 42
 $\mathcal{L}_N(\boldsymbol{\rho}_p)$ N -pulse LADAR image at pixel p . 55

Discretized Framework

$\mathbf{C}_{\mathbf{z}|\mathcal{T}}$ covariance matrix for signal vector \mathbf{z} given the target \mathcal{T} . 84
 ϵ normalized intensity envelope of the reference irradiance pattern. 71
 \mathcal{F} Fresnel number product quantifying the strength of the photon transfer from the target to the detector. 71
 $\Gamma_0(p, q)$ noise associated with the speckled fields from pixels p and q ; equal to the turbulence-free variance of \mathcal{S}_p when $p = q$. 78
 $\Gamma_1(p, q)$ noise term associated with the interaction of the light field and rough surface at pixels p and q . 78
 Γ_2 noise term associated with the interaction of the light field and rough surface over a pixel. 183
 Γ_3 noise term associated with the interaction of the light field and rough surface over a pixel. 183
 \mathbf{I} discretized reference intensity pattern. 71
 $\bar{\mathbf{I}}$ rank-one background component of reference pattern. 72
 $\tilde{\mathbf{I}}$ zero-mean (nearly) iid component of reference pattern. 73
 $\mathbf{1}$ identity matrix. 97
 $\mathcal{I}(p, q, n)$ coefficient that weights contribution of the interference pattern from pixels p and q to measurement n . 75
 $\kappa_\alpha(p, p')$ measure of discretization error. 76
 $\boldsymbol{\mu}$ vector of the total number of on-target photons for each pulse, scaled by the Fresnel number product \mathcal{F} . 72
 \mathcal{N}_n number of photons received by the bucket detector for pulse n ($\mathbb{N}_b(n)$). 70

$\Phi_{np}(\boldsymbol{\rho})$ speckled field caused by the reflection of pulse n from pixel p . 74

\mathcal{S}_p speckle coefficient. 179

\mathcal{T}_p average intensity reflectivity for pixel p in the discretized framework. 73

\mathbf{tr} trace of a matrix. 90

\mathbf{v}_n zero-mean, inter-pixel noise term that is added to \mathbf{z}_n to form the total number of received photons, \mathcal{N}_n . 80

\mathbf{z}_n signal term, a linear transformation of the discretized target pixels \mathcal{T}_p . 80

Probability Theory and Estimation

\mathbf{A} measurement matrix. 92

$\bar{\mathbf{A}}$ mean elements of \mathbf{A} . 113

\mathbf{A}_k column k of matrix \mathbf{A} . 93

$\tilde{\mathbf{A}}$ centered version of \mathbf{A} . 113

$\mathcal{C}(\mathbf{x})$ Cramér-Rao bound for average mean-squared error, given by the normalized trace of the inverse Fisher information. 91

$\mathcal{C}_{\mathbf{B}}(\mathbf{x})$ Bayesian Cramér-Rao bound for average mean-squared error, given by the normalized trace of the inverse Bayesian Fisher information. 103

$\mathbf{C}_{\mathbf{e}(\mathbf{x})}$ error covariance matrix for a linear estimator of \mathbf{x} . 76

$\mathbf{C}_{\mathbf{x}}$ autocovariance matrix of \mathbf{x} . 76

$\mathbf{C}_{\mathbf{xy}}$ cross-covariance matrix of \mathbf{x} and \mathbf{y} . 75

$\mathbf{C}_{\mathbf{y}}^{-1}$ inverse of the autocovariance matrix of \mathbf{y} . 75

$\mathbf{C}_{\mathbf{z}|\mathbf{x}}$ covariance matrix of \mathbf{z} , given \mathbf{x} . 92

$|\mathbf{C}_{\mathbf{z}|\mathbf{x}}|$ determinant of $\mathbf{C}_{\mathbf{z}|\mathbf{x}}$. 92

$\mathbf{D}_{\mathcal{X}}$ diagonal matrix whose diagonal is vector \mathcal{X} . 97

$\mathbf{e}(\mathbf{x})$ error in the linear estimator, $\mathbf{x} - \hat{\mathbf{x}}^{(L)}$. 76

$\Gamma(\cdot)$ standard gamma function. 105

$\mathbf{J}_{\mathbf{y}}(\mathbf{x})$ Fisher information matrix in \mathbf{y} about \mathbf{x} . 90

$\mathbf{J}_{\mathbf{z},\mathbf{x}}$ joint Fisher information matrix for \mathbf{z} and \mathbf{x} . 101

Ω scaling coefficient for $\bar{\mathbf{C}}_{\mathbf{z}}$. 113, 114

$p_{\mathbf{x}}(x)$ probability density function for \mathbf{x} . 20

$p_{\mathbf{x}}(x; \theta)$ probability density function for \mathbf{x} for parameter θ . 20

$p_{x|y}(x|y)$ conditional probability density function for x given y . 20

$p_{y,x}(y, x)$ joint probability density function for y and x . 20

strong law of large numbers sample average of iid random variables converges

to the sample mean *with probability one, or almost surely*, and denoted $\xrightarrow{w.p.1}$.

229

x a random variable; a sans-serif font is used to denote the random variable, while the regular font, x , is used for a realization of that variable. 21

\bar{x} prior mean of the random variable x . 21

\hat{x} estimate of the random variable x . 21

$\hat{\mathbf{x}}^{(L)}$ linear estimate of the vector \mathbf{x} . 75

\mathcal{X} vector comprised of the elements $\Gamma_0 x_m^2$. 97

Message Passing and GAMP

\hat{A}_{ij} estimate of measurement matrix elements. 146

ν_{ij}^A variance of measurement matrix elements. 146

$\Delta_{j \leftarrow i}^t(x_j)$ message from factor node i to variable node j . 125

$\Delta_{j \rightarrow i}^t(x_j)$ message from variable node j to factor node i . 125

\mathcal{F}_i neighbors of factor node i . 124

$f_i(\cdot)$ factor node i 's potential function. 124

$f_{\mathbf{in}}^{(j)}(x_j, \theta)$ input factor-node potential function. 133

$f_{\mathbf{out}}^{(i)}(y_i, z_i)$ output factor-node potential function. 133

$G_x(\cdot)$ input estimation function. 141

$G_y(\cdot)$ output estimation function. 138

$H_x(\cdot)$ input adaptation function. 168

$\hat{p}_i(t)$ estimate in GAMP algorithm. 136

$\nu_i^p(t)$ uncertainty associated with $\hat{p}_i(t)$. 136

$\hat{r}_j(t)$ estimate in GAMP algorithm. 139

$\nu_j^r(t)$ uncertainty associated with $\hat{r}_j(t)$. 139

$\hat{s}_i(t)$ estimate in GAMP algorithm. 138

$\nu_i^s(t)$ uncertainty associated with $\hat{s}_i(t)$. 138

\mathcal{X}_j neighbors of variable node j . 124