9-2020

# Improving Closely Spaced Dim Object Detection Through Improved Multiframe Blind Deconvolution

Ronald M. Aung


# IMPROVING CLOSELY SPACED DIM OBJECT DETECTION THROUGH IMPROVED MULTI-FRAME BLIND DECONVOLUTION

DISSERTATION

Ronald M. Aung, Major, USAF

AFIT-ENG-DS-20-S-004

**DEPARTMENT OF THE AIR FORCE**
**AIR UNIVERSITY**

**AIR FORCE INSTITUTE OF TECHNOLOGY**

**Wright-Patterson Air Force Base, Ohio**

AFIT-ENG-DS-20-S-004

# IMPROVING CLOSELY SPACED DIM OBJECT DETECTION THROUGH IMPROVED MULTI-FRAME BLIND DECONVOLUTION

DISSERTATION

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

Ronald M. Aung, BS EECS, MS IE, MS EE

Major, USAF

September 2020

AFIT-ENG-DS-20-S-004

# IMPROVING CLOSELY SPACED DIM OBJECT DETECTION THROUGH IMPROVED MULTI-FRAME BLIND DECONVOLUTION

Ronald M. Aung, BS EECS, MS IE, MS EE
Major, USAF

Committee Membership:

Stephen C. Cain, PhD
Chair

Benjamin F. Akers, PhD
Member

Major David J. Becker, PhD
Member

Major Tyler J. Hardy, PhD
Member

ADEDJI B. BADIRU, PhD
Dean, Graduate School of Engineering and Management

AFIT-ENG-DS-20-S-004


## Abstract

This dissertation focuses on improving the ability to detect dim stellar objects that are in close proximity to a bright one, through statistical image processing using short exposure images. The goal is to improve the space domain awareness capabilities with the existing infrastructure. In this research, two new algorithms are developed to improve dim object detection. The first one is through the Neighborhood System Blind Deconvolution where the data functions are separated into the bright object, the neighborhood system around the bright object, and the background function. The second one is through the Dimension Reduction Blind Deconvolution, where the object function is represented by the product of two matrices, whose ranks are lower than the size of the object function. Both are designed to overcome the photon counting noise and the random and turbulent atmospheric conditions. The performance of the algorithms are compared with that of the multi-frame blind deconvolution approach by Schulz because the new algorithms also use the Poisson noise model similar to Schulz. The new algorithms are tested and validated with computer generated data. The Neighborhood System Blind Deconvolution is also modified to overcome the undersampling effects since it is validated on the undersampled laboratory collected data. Even though the algorithms are designed for ground to space imaging systems, the same concept can be extended for space to space imaging. This research provides two better techniques to improve closely spaced dim object detection.

## Acknowledgements

I would first like to thank my advisor Dr. Stephen Cain. His guidance, mentoring, and patience have been invaluable in helping me complete this dissertation. I am truly grateful to have him as my advisor. I would like to thank Dr. Benjamin Akers, Maj David Becker, and Maj Tyler Hardy for reviewing my research and giving me valuable feedback.

I would like to thank Brig Gen (ret) Richard Scott Stapp for mentoring me and giving me invaluable advices. His encouragement and confidence in me were critical in my decision to pursue this PhD. I would also like to thank Lt Col Robert Koo for motivating me to pursue a PhD ever since I was a young officer. In addition to friendship and guidance, I am very grateful for the opportunity he gave me to become a flight test engineer a decade ago, which ultimately put me on the path to pursue this PhD. I would like to thank Dr. Michael Temple and Lt Col Nathaniel Liefer for giving me this PhD opportunity. I would like to thank Maj (ret) Greg Padilla and family, Maj Nicholas Estep and family, Maj David Lynes and family, John Lee, and Noah Leslie for their friendships during this journey. I would also like to thank my four nieces Maybe, Norah, Indy, and Julia for giving me lots of much needed cuteness and love.

Finally, I would like to thank my parents for instilling in me the discipline and the value of education from a very young age. I would like to thank my sister for all her love, support, and friendship while keeping me honest. I am truly grateful for all the opportunities given to me by the U.S. Air Force and the National Reconnaissance Office.

Ron Aung

This page is left blank intentionally.

# Table of Contents

Page

# List of Figures

Page

# List of Appendix Figures

## Appendix B



## Appendix C



## Appendix D

# List of Tables

Page

## List of Commonly Used Acronyms

| Acronym | Definition |
|---|---|
| 1D | One-Dimensional |
| 1DBD | One-Dimensional Multi-Frame Blind Deconvolution |
| 2D | Two-Dimensional |
| AMOS | Air Force Maui Optical & Supercomputing Site |
| ASAT | Anti-Satellite Weapon |
| AWGN | Additive White Gaussian Noise |
| CCD | Charge-Coupled Device |
| CSPOC | Combined Space Operations Center |
| DRBD | Dimension Reduction Blind Deconvolution |
| EM | Expectation-Maximization Algorithm |
| GS | Gerchberg-Saxton Phase Retrieval Algorithm |
| HST | Hubble Space Telescope |
| JSPOC | Joint Space Operations Center |
| MFBD | Multi-Frame Blind Deconvolution |
| MLE | Maximum Likelihood Estimation |
| NSBD | Neighborhood System Blind Deconvolution |
| OTF | Optical Transfer Function |
| PMF | Probability Mass Function |
| PSF | Point Spread Function |
| ROC | Receiver Operating Characteristic Function |
| SDA | Space Domain Awareness |
| SOR | Starfire Optical Range |
| SSN | Space Surveillance Network |
| STRATCOM | United States Strategic Command |
| SVD | Singular Value Decomposition |
| USAF | United States Air Force |
| USSF | United States Space Force |

## List of Commonly Used Symbols

| Symbol | Description |
|---|---|
| $\alpha$ | Level of Significance |
| $\gamma$ | Bright Object Intensity |
| $\delta(\cdot)$ | Dirac Delta Function |
| $\eta$ | Total Intensity of Object Function |
| $\lambda$ | Wavelength |
| $\phi(u)$ | Atmospheric Phase Screen |
| $A(u)$ | Aperture Transmittance Function |
| A.T. | Additional Term |
| $a_k$ | Zernike Coefficient |
| $B$ | Background Intensity |
| $D$ | Aperture Diameter |
| $d_k(y)$ | Data Function |
| $d_k(z)$ | Undersampled Data Function |
| $\tilde{d}_k(y,x)$ | Complete Data Function |
| $\tilde{d}_k(z,x,\upsilon)$ | Undersampled Complete Data Function |
| $\mathrm{E}[\cdot]$ | Expectation or Mean |
| $\mathfrak{F}\{\cdot\}$ | Fourier Transform Operator |
| $h_k(y;x)$ | Point Spread Function |
| $i_k(y)$ | Image Function |

| Symbol | Description |
|---|---|
| $\tilde{i}_k(y,x)$ | Complete Image Function |
| $K$ | Number of Frames Used in Statistical Image Processing |
| $L$ | Undersampling Factor |
| $L(\cdot)$ | Joint Likelihood Function |
| $\ell(\cdot)$ | Joint Logarithm Likelihood Function |
| ln | Natural Logarithm |
| $N \times N$ | Size of Object Function |
| $o(x)$ | Object Function |
| $o_1(x_1)$ or $o_1(x_1,\upsilon)$ | Object Function with Dimension Reduced in Y-Direction |
| $o_2(x_2)$ or $o_2(\upsilon,x_2)$ | Object Function with Dimension Reduced in X-Direction |
| $o_b(x)$ | Background Function |
| $o_p(x)$ | Primary Bright Object Function |
| $o_s(x)$ | Neighborhood System Function |
| $\mathrm{P}[\cdot]$ | Probability Distribution or Mass Function |
| $Q(\cdot)$ | Conditional Expectation Function |
| $r_0$ | Fried Parameter |
| $r(y)$ | Rectangle Average Function |
| $Z_k(u)$ | Zernike Polynomial |

**IMPROVING CLOSELY SPACED DIM OBJECT DETECTION THROUGH IMPROVED MULTI-FRAME BLIND DECONVOLUTION**

# 1. Introduction

## 1.1. Chapter Overview

This chapter starts with the motivation for the need to detect dim or small stellar objects that are in close proximity to a bright object. Then, it explains the overall purpose and specific research tasks to improve such capabilities. It also describes the assumptions that are made throughout this research. This chapter is concluded with the organization of this dissertation.

## 1.2. Motivation

The earth orbits have been increasingly congested ever since the launch of the Sputnik I in October 1957 [1]. Each launch adds debris and rocket bodies into these earth orbits. When China launched the anti-satellite (ASAT) weapon against their weather satellite in January 2007, it added a cloud of debris into the already congested space [2]. In February 2009, the collision between a Russian satellite and an American Iridium global communications satellite introduced an additional cloud of space debris [3]. Figure 1.1 shows the increasing trend for the total amount of space debris with two sharp increases in 2007 and 2009 from these events.

In addition, the SpaceWorks Enterprise, an aerospace engineering company that specializes in the design and assessment of advanced space concepts for both government and commercial customers, estimated in 2019 that there is a market for 2,000 to 2,800

nanosatellites and microsatellites to be launched over the next 5 years [4]. The categorization of the satellites are shown in Figure 1.2.

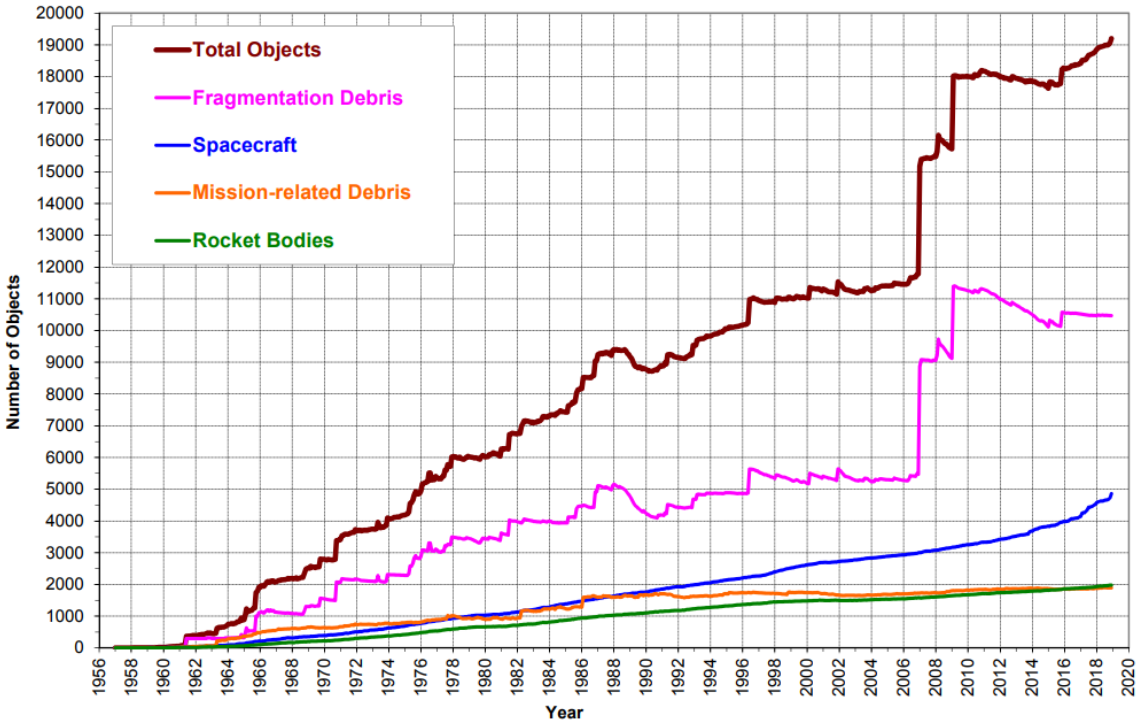


**Figure 1.1. Number of Objects in Earth Orbit by Object Type [5].**

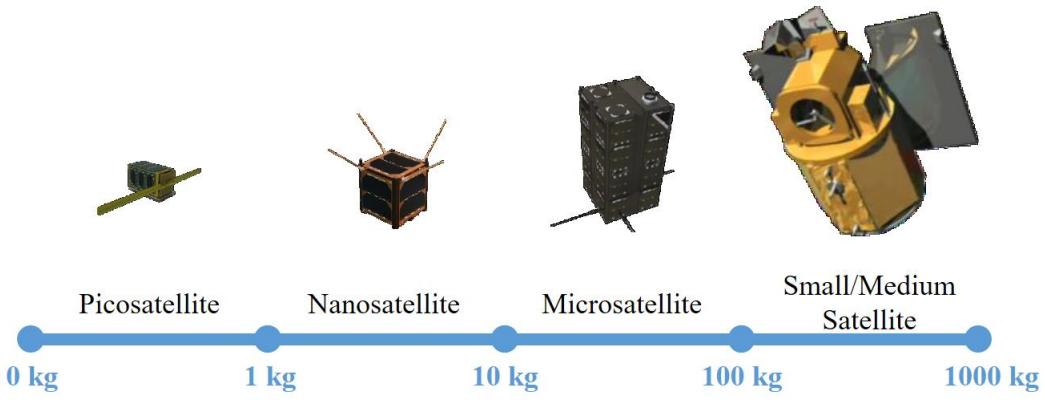


**Figure 1.2. Satellite Categorization [4].**

With the proliferation of nanosatellites and microsatellites and the huge amount of space debris, maintaining situational awareness on the high value space assets is very critical. These orbital debris and small objects can instantly destroy or disable the space assets. A collision with a ten centimeter object would catastrophically damage a typical satellite, a one centimeter object would likely disable a spacecraft, and a one millimeter object could destroy a satellite sub-system [6]. For example, in 2013, when the Russian nanosatellite called BLITS ("Ball Lens in the Space") was knocked out of its orientation by a debris from the 2007 Chinese ASAT [7]. Therefore, detecting these small objects is very critical to ensure that they would not intentionally or unintentionally collide with the high value space assets, damaging and degrading the critical missions and services for the military, the commercial industries, and the science and academic institutions.

### 1.3. Overall Purpose and Research Tasks

Since dependable and safe operations in space are vital to the United States national security as well as global economic viability [8], the United States Strategic Command (STRATCOM) is charged with the space control mission, as one of its missions. The space control mission is conducted at the Combined Space Operations Center (CSPOC), which detects, tracks, and identifies all man-made objects in the Earth orbits [9]. The CSPOC was transitioned from the Joint Space Operations Center (JSPOC) on 18 July 2019 to improve coordination between the U.S., its allies, and the commercial and civil partners for defensive space efforts [10]. On 20 December 2019, the United States Space Force (USSF) became the sixth branch of the armed forces because of the importance of space for the way of life, the economy, and the national interests of the United States [11].

The overall purpose of this research is to help contribute the space control mission of the USSF by improving the algorithm for detecting small objects through statistical image processing techniques, using the existing infrastructures. In addition to the national security applications, the techniques found in this research can be adopted for astronomy and other academic disciplines. To support the overall purpose, this research proposes the following tasks.

1. Develop algorithm(s) to improve the ability of detecting small space objects using short exposure images.
2. Validate the algorithm(s) with computer generated data.
3. Validate the algorithm(s) with the telescope systems within the Space Surveillance Network or the academic and scientific communities.

### 1.4. Sources of Image Degradation

This research addresses three major sources of image degradation. The first source is the photon counting noises associated with the charge-coupled devices (CCD). The second source is the atmospheric turbulence. The last source is the spatial undersampling. The first two are random in nature and the last one is deterministic. In addition, for ground to space imaging, the first two are always coupled, with or without the last one. Depending on the image collection setup, such as the size of the CCD pixel and the geometry of the CCD and the stellar objects of interest, the last source may or may not be coupled with the first two.

## 1.5. Assumptions and Limitations

To address these sources of degradation, the following assumptions are made throughout this research. They are explained in the Methodology chapter.

1. The Fresnel far-field propagation criteria is met, allowing to implement the propagation as a Fourier Transform.
2. The imaging region is smaller than the isoplanatic patch size, where the imaging system is linear and shift-invariant.
3. The effects of the atmospheric turbulence is frozen for short exposure images.
4. The atmospheric point spread functions are uncorrelated.
5. The CCD introduces signal dependent Poisson noise.
6. The background noise also follows Poisson distribution.
7. The light incident on the CCD is temporally incoherent.
8. The intensities of the data function pixels are statistically independent.

## 1.6. Document Outline

This document is organized in the following way. Chapter 2 provides the relevant background on the noise introduced by the CCD, the mathematical model for representing and generating the atmospheric turbulence using the Zernike technique, and the evolution of the blind deconvolution algorithm from the approach by Richardson and Lucy to the multi-frame blind deconvolution (MFBD) approach by Schulz. It also explains the Gerchberg-Saxton phase retrieval algorithm as well as the mathematical model for describing the effects of spatial undersampling.

Chapter 3 provides the methodology where the assumptions are explained. The mathematical model for the imaging system is explained, along with the parameters used for generating atmospheric turbulence phase screens using the Zernike technique. These phase screens along with signal dependent Poisson noise are used for generating the data functions to test and validate the algorithms. A total of eight scenarios are generated where the intensities of the dim object vary from 10% to 0.25% of the intensity of the bright object. It also explains the two metrics for comparing the performance of the new algorithms with the MFBD approach by Schulz.

In Chapter 4, the Neighborhood System Blind Deconvolution (NSBD) is derived, where the data functions are separated into three sets, which are the primary bright object, the neighborhood system around the bright object, and the background. The performance of the NSBD algorithm is compared with that of the MFBD using the computer generated data.

In Chapter 5, the NSBD is modified to overcome the undersampling effects so that it can be used for testing the undersampled laboratory data collected at the Air Force Institute of Technology. In addition, a computer generated data is created to mimic the laboratory conditions to ensure that the performance results are consistent. The performance of the undersampled NSBD is compared with that of the MFBD, which is also modified to take into account for the undersampling effects.

In Chapter 6, the Dimension Reduction Blind Deconvolution (DRBD) is derived, where the object function is represented as a product of two matrices, whose ranks are lowered than the dimension of the object function. In the first case, the object function is

assumed to be spatially separable, meaning that it can be represented by the outer product of two one-dimensional vectors. The second case no longer assumes the object function to be spatially separable where the object function with the size of $N \times N$ pixels is represented by the product of $N \times 2$ and $2 \times N$ matrices. The DRBD algorithm is tested on the computer generated data and the performance results are compared with that of the MFBD algorithm.

This dissertation is concluded with Chapter 7, which summarizes the work completed. It also discusses the potential future work and the publications.

# 2. Background and Literature Review

## 2.1. Chapter Overview

This chapter presents the background and literature review as well as the foundation for the tools and methods that are used throughout this research. As mentioned in Section 1.4, there are three major sources of degradation to the quality of astronomical images: (1) the photon counting noise associated with the charge-coupled device (CCD), (2) the atmospheric turbulence, and (3) the spatial undersampling. This chapter provides the background for the sources of the degradation and the techniques to overcome them. It starts with an overview of the astronomy detectors, including the CCDs. It then explains the statistics of the associated photon counting noise. Next, it provides an overview of the atmospheric turbulence and a technique to simulate the turbulence using the Zernike polynomials since the turbulence can be observed in the point spread function (PSF). Afterwards, it provides methods for overcoming the first two sources of image degradation and an explanation for the last one. Then, it introduces the blind deconvolution algorithm to recover the degraded images. Combining the developments in electrical engineering, astronomy, statistics, and medical communities, this chapter shows how the Richardson-Lucy method is evolved into the Expectation-Maximization (EM) based Multi-Frame Blind Deconvolution (MFBD) algorithm, which is the current standard for the United States Air Force's Space Domain Awareness (SDA) applications and also used by the scholars in the electro-optics community [12]–[15]. Then, this chapter describes a phase retrieval method to recover the two-dimensional phase to address and overcome the image

degradation caused by the atmospheric turbulence. Lastly, it explains the process of the spatial undersampling so that the effects can be compensated.

## 2.2. Evolution of Astronomy Detectors

Astronomers have been observing the skies in search of asteroids, planets, and stars for centuries. The act of finding space objects with ground based telescopes has been performed by astronomers since the 17th century [16]. Even though he did not invent the telescope, Galileo Galilei was the first person to use a telescope to look at the celestial bodies [17]. Since then, astronomy has made great strides. From observation through naked eyes, the technology has evolved to photography, to photoelectric single-channel devices, to plate scanners, to television-type imagers, to semiconductor-based devices, and to energy-resolution arrays [18]. The CCDs and avalanche photodiode detectors fall under the semiconductor-based device category, while the superconducting tunnel junction falls under the energy resolution array category.

Different types of sensors introduce different types of limitations as well as different types of noise. The CCDs have been the most common imaging sensors in the astronomy and space surveillance communities for visible and near ultraviolet light since the 1980s [19], [20]. Since the overall goal of this research is to improve closely spaced dim object detection and image reconstruction using the existing infrastructure, the Poisson noise model associated with the CCD is used throughout this research.

## 2.3. Photon Counting Noise Overview

In any optical telescope system, there are many sources of noise that can negatively affect the ability to detect closely spaced objects, especially when they are very dim. These

include photon counting or photo-conversion noise, background light, readout noise, non-uniform flat-field response, non-uniform spectral response, and extraneous charge carriers resulting from bias, dark current, and both internal and external background radiation [21], [22]. The systems deployed for the SDA applications generally utilize the CCDs where the primary concern is the Poisson distributed photon counting noise [23]–[25]. Additionally, the internal and external background radiation, the dark current, and the bias are also Poisson distributed random variables [26]. As such, this research derives the new algorithms using the Poisson noise model.

This section introduces the Poisson random variable and its key characteristics that will be used throughout this research. Let $\{X_1, X_2, ..., X_K\}$ be a set of independent Poisson random variables with their corresponding means of $\{\lambda_{X_1}, \lambda_{X_2}, ..., \lambda_{X_K}\}$.

$$X_j \sim \text{Poisson}\left(\lambda_{X_j}\right) \text{ where } j = 1, ..., K. \tag{2.1}$$

The first key characteristic is that the sum of independent Poisson random variables is also a Poisson random variable. In addition, the mean of the sum of the Poisson random variables is equal to the sum of the means of the Poisson random variables [27]. In other words,

$$\begin{aligned} Y &= X_1 + X_2 + ... + X_K, \\ \lambda_Y &= \lambda_{X_1} + \lambda_{X_2} + ... + \lambda_{X_K}, \\ Y &\sim \text{Poisson}\left(\lambda_Y\right). \end{aligned} \tag{2.2}$$

The second key characteristic is that the conditional probability $\text{P}\left(X_j \middle| Y\right)$ takes on the form of a Binomial random variable [28]. To make the derivation simple, an intermediate random variable, *Z*, is introduced.

$$
\begin{aligned}
Z &= \sum_{\substack{i=1 \\ i \neq j}}^{K} X_i, \\
\lambda_Z &= \sum_{\substack{i=1 \\ i \neq j}}^{K} \lambda_{X_i}, \\
Z &\sim \text{Poisson}\left(\lambda_z\right).
\end{aligned}
\tag{2.3}
$$

Next, the conditional probability is calculated as the following:

$$
\begin{aligned}
\mathrm{P}\left(X_j = k \mid Y = n\right) &= \frac{\mathrm{P}\left(X_j = k, Y = n\right)}{\mathrm{P}\left(Y = n\right)} \\
&= \frac{\mathrm{P}\left(X_j = k, X_j + Z = n\right)}{\mathrm{P}\left(X_j + Z = n\right)} \\
&= \frac{\mathrm{P}\left(X_j = k, Z = n - k\right)}{\mathrm{P}\left(X_j + Z = n\right)} \\
&= \frac{\mathrm{P}\left(X_j = k\right)\mathrm{P}\left(Z = n - k\right)}{\mathrm{P}\left(X_j + Z = n\right)}.
\end{aligned}
\tag{2.4}
$$

After rearranging the variables, the probabilities on the right side of Equation (2.4) are substituted with their associated Poisson probability mass functions (PMF). Rearranging the terms results in a binomial PMF as shown in Equation (2.5).

$$
\begin{aligned}
\mathrm{P}\left(X_j = k \mid Y = n\right) &= \frac{\left(e^{-\lambda_{X_j}} \frac{\lambda_{X_j}^{k}}{k!}\right)\left(e^{-\lambda_Z} \frac{\lambda_Z^{n-k}}{(n-k)!}\right)}{e^{-\left(\lambda_{X_j}+\lambda_Z\right)} \frac{\left(\lambda_{X_j}+\lambda_Z\right)^{n}}{n!}} \\
&= \left(\frac{n!}{k!(n-k)!}\right) \frac{\left(\lambda_{X_j}\right)^{k}\left(\lambda_Z\right)^{n-k}}{\left(\lambda_{X_j}+\lambda_Z\right)^{n}} \\
&= \left(\frac{n!}{k!(n-k)!}\right)\left(\frac{\lambda_{X_j}}{\lambda_{X_j}+\lambda_Z}\right)^{k}\left(\frac{\lambda_Z}{\lambda_{X_j}+\lambda_Z}\right)^{n-k} \\
&= \left(\frac{n!}{k!(n-k)!}\right)\left(\frac{\lambda_{X_j}}{\lambda_Y}\right)^{k}\left(\frac{\lambda_Y-\lambda_{X_j}}{\lambda_Y}\right)^{n-k} \\
&= \mathrm{P}_{\text{Binomial}}\left(X_j = k; N = n, p = \frac{\lambda_{X_j}}{\lambda_Y}\right).
\end{aligned}
\tag{2.5}
$$

With the second key characteristic, the third key characteristic can be derived, which is the conditional expectation, $\mathrm{E}\left[X_k \middle| Y = X_1 + \ldots + X_N\right], 1 \le k \le N$, using the expectation of a binomial PMF described in Equation (2.6) [27].

$$
\mathrm{E}_{\text{Binomial}}\left[X; N, p\right] = Np\,, \tag{2.6}
$$

$$
\mathrm{E}_{\text{Poisson}}\left[X_j \mid Y = n\right] = \mathrm{E}_{\text{Binomial}}\left[X; N = n, p = \frac{\lambda_{X_j}}{\lambda_Y}\right] = n\frac{\lambda_{X_j}}{\lambda_Y}\,. \tag{2.7}
$$

The above equation (2.7) plays an important role in the subsequent section on deriving the blind deconvolution algorithm.

## 2.4. Atmospheric Turbulence Simulation

Since the atmospheric turbulence is one of the three major sources of image degradation, the simulation of the atmospherically distorted waveforms is important in

understanding the propagation of light and imaging through the atmospheric turbulence. The optical effects of atmospheric turbulence arise from random inhomogeneities in the temperature distribution of the atmosphere [29]. This section explains the technique for simulating such effects.

Even though there are many techniques such as Kolmogoroff, Tatarskii, von Karman, and the exponential spectrum models [30]–[33], these techniques often underrepresent the lower-order aberrations such as tilt, which makes up a majority of the atmospheric energy spectrum [34], [35]. Since the Zernike polynomial-based phase screen addresses such lower order aberrations [35], [36], the Zernike technique is used in this research.

### 2.4.1. Zernike Polynomial Overview

This research uses the modified Zernike polynomial, proposed by Noll in 1976, which permits all the statistical aberration strengths to be calculated analytically [37], even though the Zernike representation was originally presented by Fried in 1965 [38]. In this research, the modified Zernike polynomials are used and they are simply called the Zernike polynomials.

The Zernike polynomials are a set of polynomials defined on a unit circle, with a key property of being orthogonal over the unit circle. The Zernike polynomials are defined as shown below.

$$Z_n^m(r,\theta) = \sqrt{2(n+1)}R_n^m(r)G^m(\theta), \tag{2.8}$$

where $r$ and $\theta$ are the radial distance and polar angle in the polar coordinate system, $m$ and $n$ are non-negative integers, representing the azimuthal frequency and radial degree, such

that $m \le n$, $R_n^m(r)$ is the radial factor, and $G^m(\theta)$ is the annular factor. However, it is convenient to express the Zernike polynomials with one index as shown in the equation below, compared to that shown in Equation (2.8).

$$Z_i(r,\theta) = \begin{cases} \sqrt{2(n+1)} R_n^m(r) G^m(\theta) & m \neq 0, \\ R_n^0(r) & m = 0, \end{cases} \tag{2.9}$$

$$R_n^m(r) = \sum_{s=0}^{(n-m)/2} \frac{(-1)^s (n-s)!}{s!\left(\frac{n+m}{2}-s\right)!\left(\frac{n-m}{2}-s\right)!} r^{n-2s}, \tag{2.10}$$

$$G^m(\theta) = \begin{cases} \sin(m\theta) & i \text{ odd}, \\ \cos(m\theta) & i \text{ even}. \end{cases} \tag{2.11}$$

Even though the mapping $(n,m) \to i$ looks complicated, it can be easily implemented in programming languages, such as MATLAB. The graphical representations for the first 15 Zernike polynomials are shown in Figure 2.1. The mathematical equations for the first 24 Zernike polynomials are shown in Table 2.1, along with their names. The orthogonality relationship for the Zernike polynomials is given by the following equations.

$$\int_0^1 R_n^m(r) R_{n'}^m(r)\, r\, dr = \frac{1}{2n+1} \delta(n,n'), \tag{2.12}$$

$$\int_0^{2\pi} G^m(\theta) G^{m'}(\theta) d\theta = \pi \delta(m,m'), \tag{2.13}$$

$$\int_0^{2\pi} \int_0^1 Z_i(r,\theta) Z_{i'}(r,\theta)\, r\, dr\, d\theta = \pi \delta(i,i'), \tag{2.14}$$

$$\delta(k,k') = \begin{cases} 1 & k = k', \\ 0 & k \neq k'. \end{cases} \tag{2.15}$$

With the polynomials completely defined, any wavefront, $\phi(r,\theta)$, can be expressed as a linear combination of the Zernike polynomials with the corresponding Zernike coefficients, $a_i$, as shown in the equation below.

$$\phi(r,\theta) = \sum_{i=1}^{\infty} a_i Z_i(r,\theta). \tag{2.16}$$

Due to the orthogonality principle, the Zernike coefficients can be decomposed using the following equation.

$$a_i = \frac{\int_0^{2\pi}\int_0^1 \phi(r,\theta) Z_i(r,\theta)\, r\, dr\, d\theta}{\int_0^{2\pi}\int_0^1 Z_i^2(r,\theta)\, r\, dr\, d\theta}. \tag{2.17}$$

When converted to the discrete two-dimensional Cartesian coordinates from the polar coordinates, the Zernike coefficients are decomposed using the following equation.

$$a_i = \frac{\sum_{\forall x}\sum_{\forall y} \phi(x,y) Z_i(x,y)}{\sum_{\forall x}\sum_{\forall y} Z_i^2(x,y)}. \tag{2.18}$$

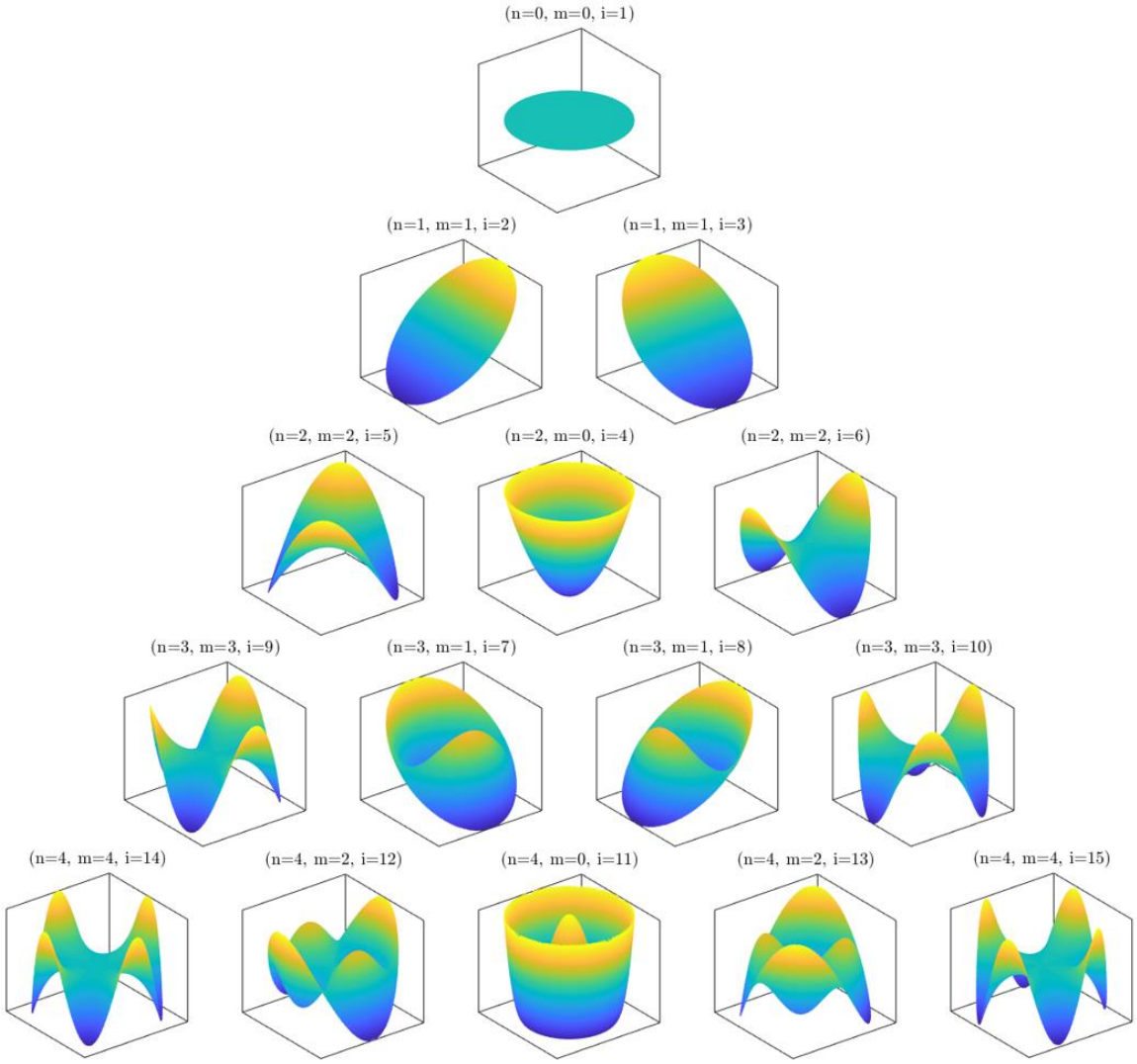


**Figure 2.1. The Graphical Representation of the first 15 Zernike Polynomials.**

**Table 2.1. The First 24 Zernike Polynomials [34].**

| $n$ | $m$ | $i$ | $Z_n^m(r,\theta)$ or $Z_i(r,\theta)$ | Name |
|---|---|---|---|---|
| 0 | 0 | 1 | $1$ | piston |
| 1 | 1 | 2 | $2r\cos(\theta)$ | $x$ tilt |
| 1 | 1 | 3 | $2r\sin(\theta)$ | $y$ tilt |
| 2 | 0 | 4 | $\sqrt{3}(2r^2-1)$ | defocus |
| 2 | 2 | 5 | $\sqrt{6}r^2\sin(2\theta)$ | $y$ primary astigmatism |
| 2 | 2 | 6 | $\sqrt{6}r^2\cos(2\theta)$ | $x$ primary astigmatism |
| 3 | 1 | 7 | $\sqrt{8}(3r^3-2r)\sin(\theta)$ | $y$ primary coma |
| 3 | 1 | 8 | $\sqrt{8}(3r^3-2r)\cos(\theta)$ | $x$ primary coma |
| 3 | 3 | 9 | $\sqrt{8}r^3\sin(3\theta)$ | $y$ trefoil |
| 3 | 3 | 10 | $\sqrt{8}r^3\cos(3\theta)$ | $x$ trefoil |
| 4 | 0 | 11 | $\sqrt{5}(6r^4-6r^2+1)$ | primary spherical |
| 4 | 2 | 12 | $\sqrt{10}(4r^4-3r^2)\cos(2\theta)$ | $x$ secondary astigmatism |
| 4 | 2 | 13 | $\sqrt{10}(4r^4-3r^2)\sin(2\theta)$ | $y$ secondary astigmatism |
| 4 | 4 | 14 | $\sqrt{10}r^4\cos(4\theta)$ | $x$ tetrafoil |
| 4 | 4 | 15 | $\sqrt{10}r^4\sin(4\theta)$ | $y$ tetrafoil |
| 5 | 1 | 16 | $\sqrt{12}(10r^5-12r^3+3r)\cos(\theta)$ | $x$ secondary coma |
| 5 | 1 | 17 | $\sqrt{12}(10r^5-12r^3+3r)\sin(\theta)$ | $y$ secondary coma |
| 5 | 3 | 18 | $\sqrt{12}(5r^5-4r^3)\cos(3\theta)$ | $x$ secondary trefoil |
| 5 | 3 | 19 | $\sqrt{12}(5r^5-4r^3)\sin(3\theta)$ | $y$ secondary trefoil |
| 5 | 5 | 20 | $\sqrt{12}r^5\cos(5\theta)$ | $x$ pentafoil |
| 5 | 5 | 21 | $\sqrt{12}r^5\sin(5\theta)$ | $y$ pentafoil |
| 6 | 0 | 22 | $\sqrt{7}(20r^6-30r^4+12r^2-1)$ | secondary spherical |
| 6 | 2 | 23 | $\sqrt{14}(15r^6-20r^4+6r^2)\sin(2\theta)$ | $y$ tertiary astigmatism |
| 6 | 2 | 24 | $\sqrt{14}(15r^6-20r^4+6r^2)\cos(2\theta)$ | $x$ tertiary astigmatism |

### 2.4.2. Zernike Phase Screen Generation

In this research, a set of Zernike coefficients is generated to simulate a turbulent wavefront. As shown in Equation (2.16), a turbulent wavefront is made up of infinitely many Zernike coefficients. Since it is impossible to generate infinitely many Zernike coefficients in simulation, this research uses a fixed number of coefficients, *p*, to simulate a wavefront.

The Zernike coefficients of an atmospheric wavefront is said to be Gaussian with zero mean and some variance [35]. However, it is not possible to directly simulate a wavefront by using Gaussian random numbers as the Zernike coefficients using Equation (2.16) because there exists a covariance among the coefficients, meaning the Zernike coefficients are not statistically independent [35].

In this research, the desire is to generate a set of uncorrelated atmospheric wavefronts. To do such, an orthonormal basis with a set of completely uncorrelated random variables is formed with the Karhunen-Loève functions as described subsequently.

The following derivations are adopted from the work by Putnam and Cain [36] and that by Roddier [35]. Let us define **A** as a vector of *p* Zernike coefficients and **C** is the covariance matrix among the first *p* modes. Since the first Zernike polynomial is a piston or a constant and it does not have any effects on the aberration, it is ignored.

$$\mathbf{A} = \begin{bmatrix} a_2 & a_3 & \dots & a_p \end{bmatrix}^T . \tag{2.19}$$

$$\mathbf{C} = \mathrm{E}\left[\mathbf{A}\mathbf{A}^T\right] = \begin{bmatrix} \mathrm{E}\left[a_2, a_2\right] & \mathrm{E}\left[a_2, a_3\right] & \cdots & \mathrm{E}\left[a_2, a_p\right] \\ \mathrm{E}\left[a_3, a_2\right] & \mathrm{E}\left[a_3, a_3\right] & \cdots & \mathrm{E}\left[a_3, a_p\right] \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{E}\left[a_p, a_2\right] & \mathrm{E}\left[a_p, a_3\right] & \cdots & \mathrm{E}\left[a_p, a_p\right] \end{bmatrix}. \tag{2.20}$$

$$\begin{pmatrix} 0.4557 & 0 & 0 & 0 & 0 & 0 & -0.0144 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.4557 & 0 & 0 & 0 & -0.0144 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0236 & 0 & 0 & 0 & 0 & 0 & 0 & -0.0039 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0236 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.0039 & 0 \\ 0 & 0 & 0 & 0 & 0.0236 & 0 & 0 & 0 & 0 & 0 & -0.0039 & 0 & 0 \\ 0 & -0.0144 & 0 & 0 & 0 & 0.0063 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.0144 & 0 & 0 & 0 & 0 & 0 & 0.0063 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.0039 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.0039 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 & 0 \\ 0 & 0 & 0 & -0.0039 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0025 \end{pmatrix}$$

**Figure 2.2. The covariance matrix, C, among the first ($p$ = 14) Zernike modes without the piston term, in $(D/r_0)^{5/3}$ units [35].**

The individual elements of the covariance matrix can be computed using the following equations [35].

$$\mathrm{E}\left[a_j, a_{j'}\right] = \frac{K_{zz'}\delta_z \Gamma\left(n + n' - 5/4\right)\left(D/r_0\right)^{5/3}}{\Gamma\left((n - n' + 17/3)/2\right)\Gamma\left((n' - n + 17/3)/2\right)\Gamma\left((n + n' + 23/3)/2\right)}, \tag{2.21}$$

$$\begin{aligned} K_{zz'} &= \frac{\Gamma\left(14/3\right)\left[(24/5)\Gamma\left(6/5\right)\right]^{5/6}\left[\Gamma\left(11/6\right)\right]^2}{2\pi^2}\left(-1\right)^{(n+n'-2m)/2}\sqrt{(n+1)(n'+1)} \\ &= 2.2698\left(-1\right)^{(n+n'-2m)/2}\sqrt{(n+1)(n'+1)}, \end{aligned} \tag{2.22}$$

$$\delta_z = \left(m = m'\right) \wedge \left(\overline{\mathrm{parity}\left(j, j'\right)} \vee \left(m = 0\right)\right), \tag{2.23}$$

where $n$, $m$, $n'$, and $m'$ are the radial degrees and azimuthal frequencies of $Z_j$ and $Z_{j'}$ associated with the coefficients $a_j$ and $a_{j'}$, $\delta_z$ is a logical Kronecker symbol, $D$ is the diameter of the aperture of the optical system, and $r_0$ is the Fried's seeing parameter.

From the numerical analysis by Roddier, $\mathbf{C}$ is known to be Hermitian as shown in Figure 2.2. Therefore, there exists a unitary matrix $\mathbf{U}$, ($\mathbf{U}^{-1} = \mathbf{U}^T$), such that

$$\mathbf{D} = \mathbf{U}\mathbf{C}\mathbf{U}^T, \tag{2.24}$$

where $\mathbf{D}$ is a diagonal matrix. Note that $\mathbf{C}$ is unique for every unique $D / r_0$ ratio. Next another vector, $\mathbf{N}$, is defined such that its elements are the coefficients of the Karhunen-Loève function as shown in Equation (2.26).

$$\mathbf{N} = \begin{bmatrix} n_2 & n_3 & \dots & n_p \end{bmatrix}^T, \tag{2.25}$$

$$\mathbf{N} = \mathbf{U}\mathbf{A}. \tag{2.26}$$

Next, the Cholesky decomposition of $\mathbf{C}$ can be derived by combining Equation (2.20), Equation (2.24), and Equation (2.26).

$$\begin{aligned} \mathbf{U}\mathbf{C}\mathbf{U}^T &= \mathbf{U}\left(\mathrm{E}\left[\mathbf{A}\mathbf{A}^T\right]\right)\mathbf{U}^T \\ &= \mathrm{E}\left[\mathbf{U}\mathbf{A}\mathbf{A}^T\mathbf{U}^T\right] \\ &= \mathrm{E}\left[\mathbf{N}\mathbf{N}^T\right]. \end{aligned} \tag{2.27}$$

Since a set of independent Gaussian random variables are also uncorrelated, selecting $\mathbf{N}$ to be a set of independent and identically distributed zero mean, unit variance Gaussian random variables gives the following.

$$\begin{aligned} \mathbf{U}\mathbf{C}\mathbf{U}^T &= \mathrm{E}\left[\mathbf{N}\mathbf{N}^T\right] = \mathbf{I} \\ \mathbf{C} &= \mathbf{U}^T\mathbf{U}. \end{aligned} \tag{2.28}$$

Since $\mathbf{N} = \mathbf{UA}$, a set of Zernike coefficients is generated from a set of independent or uncorrelated Gaussian random variables as the following.

$$\mathbf{A} = \mathbf{U}^T \mathbf{N}. \tag{2.29}$$

With the resultant Zernike coefficients from the equation above, they can be used in Equation (2.16) to simulate a turbulent wavefront. Since **N** is a set of independent Gaussian random variables, any two wavefronts generated using this technique are uncorrelated.

## 2.5. Point Spread Function

In an aberration free optical system, an aperture function is defined shown in the equation below.

$$A(u) = \begin{cases} 1 & u \in \text{Transparent Region}, \\ 0 & u \in \text{Opaque Region}, \end{cases} \tag{2.30}$$

where $u$ represents a region over which the two dimension aperture region is defined. The physical and mathematical representations are shown in Figure 2.3.

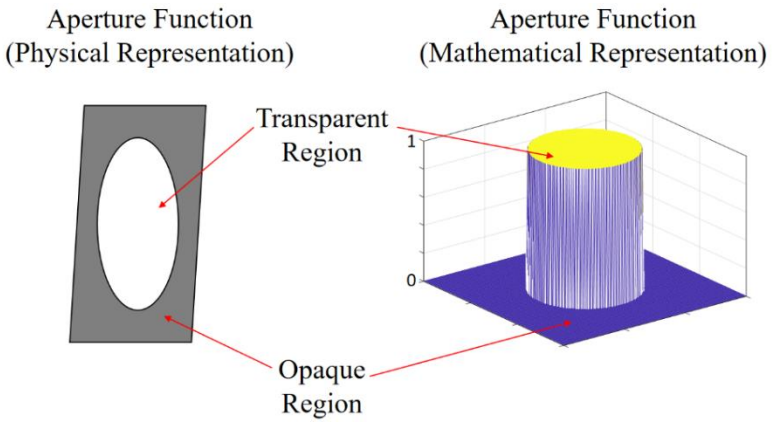


**Figure 2.3. Graphical Representation of Aberration Free Pupil Function.**

For incoherent light, an aberration free PSF, *h*, is defined as the square of the magnitude of the Fourier Transform of the aperture function as shown in the equation below [39].

$$h(x) = \frac{1}{U}\left|\mathcal{F}\left\{A(u)\right\}\right|^2, \tag{2.31}$$

where *x* is a pixel in the discrete two-dimensional region over which the PSF is defined, $\mathcal{F}$ is the Fourier Transform function, and *U* is some constant to ensure that the PSF is a conservative process and sums up to unity as shown in the equation below.

$$\sum_{\forall x} h(x) = 1. \tag{2.32}$$

In the presence of atmospheric turbulence, the atmospheric PSF can be derived as shown in the equation below.

$$h(x) = \frac{1}{U}\left|\mathcal{F}\left\{A(u)e^{j\phi(u)}\right\}\right|^2 = \frac{1}{U}\left|\mathcal{H}(x)\right|^2, \tag{2.33}$$

where $\phi$ is the atmospheric wavefront derived in Equation (2.16) and $j$ is the imaginary number. The Fourier Transform of the aperture function with an atmospheric wavefront, $\mathcal{H}$, is defined as shown in the equation below. This term is used in the Gerchberg-Saxton phase retrieval algorithm descried in the later section.

$$\mathcal{H}(x) = \mathcal{F}\left\{A(u)e^{j\phi(u)}\right\}. \tag{2.34}$$

The Fourier Transform of the PSF is called the optical transfer function (OTF), which is represented by *H*.

$$H(f) = \mathcal{F}\left\{h(x)\right\}. \tag{2.35}$$

## 2.6. Blind Deconvolution Algorithm

In signal processing, convolution refers to computing the output function from a given input function and an impulse response of a linear shift-invariant system as shown in Figure 2.4. The dimensionality of the input, output, and impulse response functions is dependent on the application. In the electro-optics community, the impulse response function is also called the PSF. In this research, the output functions are two dimensional short exposure speckled images. Conversely, deconvolution refers to as determining the input function or the impulse response from the known output function in the off-line processing. If deconvolution is performed for the real-time processing, it is referred to as equalization [40]. In astronomy, since the estimation of the input function is conducted after the image function or the output function is obtained, the problem falls under deconvolution instead of equalization.

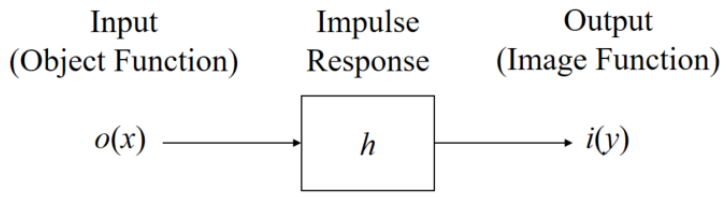


**Figure 2.4. Signal Processing System Diagram.**

There are two categories of deconvolution problems. The first category includes system identification, where the input function and the corresponding output function are known. Alternatively, in some situations, the requirement is to calculate the input function where the system impulse response and the output function are known. The second

category is called blind deconvolution when both the input function and the impulse response are not known, and the only available information is the output function. Since the goal is to estimate the unknown astronomical object function that is passed through some unknown impulse responses to generate the output image function, this research falls into the blind deconvolution category.

**Table 2.2. Signal Processing Categories.**

| **Problem** | **Processing** | **Input** | **Impulse Response** | **Output** |
|---|---|---|---|---|
| Convolution | Any | Known | Known | Unknown |
| Deconvolution | Off-Line | Unknown | Known | Known |
| Deconvolution (System Identification) | Off-Line | Known | Unknown | Known |
| Equalization | Real-Time | Unknown | Known | Known |
| Equalization (System Identification) | Real-Time | Known | Unknown | Known |
| ***Blind Deconvolution*** | ***Off-Line*** | ***Unknown*** | ***Unknown*** | ***Known*** |
| Blind Equalization | Real-Time | Unknown | Unknown | Known |

Even though there are many well-known statistical filtering algorithms, such as the Wiener and Kalman filters, they are known for removing the additive white Gaussian noise (AWGN) that is independent of the signal [41]–[43]. The image processing model of these types can be mathematically and pictorially described as shown in Equation (2.36) and Figure 2.5.

$$i(y) = \sum_{\forall x} o(x)h(y-x) + n(y), \tag{2.36}$$

where $o(x)$ is the object function, $h(x)$ is the PSF, $n(y)$ is the additive white Gaussian noise, $i(y)$ is the resultant image function, $x$ is the two-dimensional (2D) coordinate for the object function, and $y$ is the 2D coordinate for the image function and the noise. The assumption is made that the imaging region of interest is smaller than the isoplanatic patch size, where the system is linear and shift-invariant [44]. Therefore, the image function can be expressed as a convolution between the PSF and the object function.

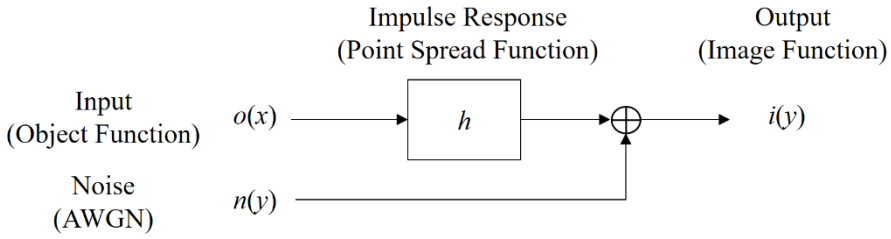


**Figure 2.5. Image Processing Model with Signal Independent Additive White Gaussian Noise.**

In addition to the Wiener and Kalman filters, there have been many researches and studies on the blind deconvolution techniques [45]–[50]. Most assumed the noise is signal-independent. However, this research uses the Poisson noise model, which is signal dependent as shown in the figure below.

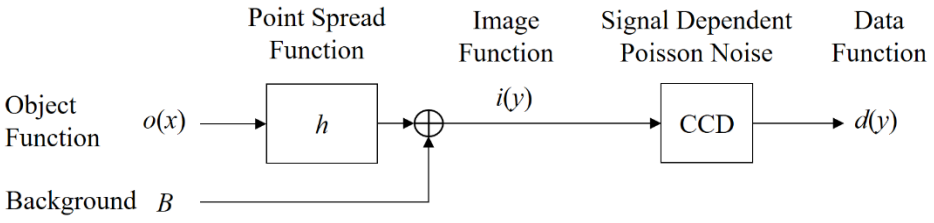


**Figure 2.6. Image Processing Model with Signal Dependent Poisson Noise.**

Le, Chartrand, and Asaki [51] and Jonsson, Huang, and Chang [52] introduced the techniques for filtering signal dependent Poisson noise using the non-linear total variation approach, originally proposed by Rudin, Osher, and Fatemi [53]. However, their approaches were focused on edge-preservation denoising [48], [54], which is different from the primary challenge in this research, which is to detect and estimate point sources. In addition, another challenge with the Total Variation and Gradient Descent Algorithms is that they are dependent on the parameters to control the step size and the smoothness function. These parameters need to be estimated or guessed. The advantage of the blind deconvolution is that it does not require these estimations.

As a result, this research is built on the blind deconvolution algorithm, which is designed to reconstruct the point sources degraded by signal dependent noise in addition to the atmospheric turbulence. Even though the blind deconvolution algorithm has been used and developed for decades, its best known application has been the reconstruction of blurred images from wide-field planetary camera from the Hubble Space Telescope (HST), which was launched on 24 April 1990 [55]. Two decades prior to the HST launch, Richardson (1972) and Lucy (1974) developed a Bayesian-based iterative deconvolution algorithm while working independently in electrical engineering and astronomy [56], [57]. In 1977, Dempster, Laird, and Rubin received widespread attention in the statistics community with their publication of an iterative Expectation-Maximization (EM) algorithm, which optimizes the maximum likelihood function (MLE) functions [58]–[60]. In 1982, Shepp and Vardi introduced the EM based image reconstruction method in emission tomography to the medical imaging community using Poisson distribution [61].

In 1993, Schulz adapted the deconvolution algorithm from Shepp and Vardi and extended it to multi-frame blind deconvolution of astronomical images [62]. The following subsections provide the overviews of these developments.

### 2.6.1. Richardson-Lucy Method

The Fourier transform has been used successfully in image restoration when the noise content is low or moderate; however the Fourier methods fail when the noise content is high [63], [64]. Richardson and Lucy applied a Bayesian-based iterative method to restore noisy degraded images in the spatial domain. In their approaches, they used the Bayes's postulate, also known as the equidistribution of ignorance, which will be explained later in this section.

The object function is defined as $\{o(x): x \in X\}$ where $X$ is the discretized two-dimensional region over which the object function is defined. The object function passes through a PSF, $\{h(y \mid x): x \in X, y \in Y\}$, where $Y$ is the discretized two-dimensional region over which the image function, $i(y)$, is defined as shown in Figure 2.7.

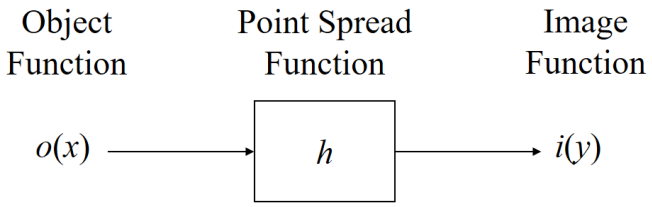


**Figure 2.7. Image Processing System Diagram.**

With the assumption that the imaging region is smaller than isoplanatic patch, the image function can be described as a convolution of the object function with the PSF.

$$i(y) = \sum_{\forall x \in X} h(y-x)o(x), \tag{2.37}$$

where $y$ is a pixel from $Y$ and $x$ is a pixel from $X$. The Bayesian probabilistic model for the object function is shown in Equation (2.38).

$$\begin{aligned} \mathrm{P}\big(o(x) \mid i(y)\big) &= \frac{\mathrm{P}\big(i(y), o(x)\big)}{\mathrm{P}\big(i(y)\big)} \\ &= \frac{\mathrm{P}\big(i(y) \mid o(x)\big)\mathrm{P}\big(o(x)\big)}{\sum_{\forall z \in X} \mathrm{P}\big(i(y) \mid o(z)\big)\mathrm{P}\big(o(z)\big)}. \end{aligned} \tag{2.38}$$

With the goal to estimate the object function, the probability of the object function is expressed as the marginal probability function of the joint distribution function between the object function and the image function as shown in Equation (2.39), which can also be expressed in terms of Bayes's Rule.

$$\begin{aligned} \mathrm{P}\big(o(x)\big) &= \sum_{\forall y} \mathrm{P}\big(o(x), i(y)\big) \\ &= \sum_{\forall y} \mathrm{P}\big(o(x) \big| i(y)\big)\mathrm{P}\big(i(y)\big). \end{aligned} \tag{2.39}$$

Substituting Equation (2.38) into Equation (2.39) results in the following.

$$\mathrm{P}\big(o(x)\big) = \sum_{\forall y} \left[ \left( \frac{\mathrm{P}\big(i(y) \mid o(x)\big)\mathrm{P}\big(o(x)\big)}{\sum_{\forall z \in X} \mathrm{P}\big(i(y) \mid o(z)\big)\mathrm{P}\big(o(z)\big)} \right) \mathrm{P}\big(i(y)\big) \right]. \tag{2.40}$$

In the equation above, the desired solution, which is the object function, appears on both sides of the equation. In many applications of the Bayes's theorem, when the desired solution is not known, an accepted practice is to make the best of a bad situation and use

an estimate of $\mathrm{P}\left(o(x)\right)$ to obtain and approximation of $\mathrm{P}\left(o(x)\middle|i(y)\right)$ [65]. Therefore, the iterative solution becomes the following.

$$\mathrm{P}\left(o^{\mathrm{new}}(x)\right)=\mathrm{P}\left(o(x)\right)\sum_{\forall y\in Y}\frac{\mathrm{P}\left(i(y)\,|\,o(x)\right)\mathrm{P}\left(i(y)\right)}{\sum_{\forall z\in X}\mathrm{P}\left(i(y)\,|\,o(z)\right)\mathrm{P}\left(o(z)\right)}. \tag{2.41}$$

The iterative solution is currently in the form of probabilistic model. Using the Bayes's Postulate or the equidistribution of ignorance, which assumes a uniform distribution for the object, image, and point spread functions, the approach converted from the probabilistic domain to the spatial domain by dividing the value of the individual pixel by the sum of all pixels of its corresponding spatial function.

$$\mathrm{P}\left(o(x)\right)=\frac{o(x)}{\sum_{\forall z\in X}o(z)}=\frac{o(x)}{S_o}, \tag{2.42}$$

$$\mathrm{P}\left(i(y)\right)=\frac{i(y)}{\sum_{\forall v\in Y}i(v)}=\frac{i(y)}{S_i}, \tag{2.43}$$

where $S_i$ is the sum of all pixels of the image function and $S_o$ is the sum of all pixels in the object function. Richardson defined the conditional probability $\mathrm{P}\left(i(y)\,|\,o(x)\right)$ as the following.

$$\mathrm{P}\left(i(y)\,|\,o(x)\right)=\frac{h(y-x)}{S_h}=h(y-x), \tag{2.44}$$

where $S_h$ is the sum of all pixels in the PSF, which is constrained to be one to ensure the process is conservative. As a result, $S_i$ is equal to $S_o$. Substituting Equation (2.42) through Equation (2.44) into Equation (2.41) results in the following equation.

$$\frac{o^{\text{new}}(x)}{S_o} = \frac{o^{\text{old}}(x)}{S_o} \sum_{\forall y \in Y} \frac{h(y-x)\frac{i(y)}{S_i}}{\sum_{\forall z \in X} h(y-z)\frac{o^{\text{old}}(z)}{S_o}}, \tag{2.45}$$

Simplifying the above equation results in the following Richardson-Lucy deconvolution algorithm.

$$o^{\text{new}}(x) = o^{\text{old}}(x) \sum_{\forall y \in Y} \frac{h(y-x)i(y)}{\sum_{\forall z \in X} h(y-z)o^{\text{old}}(z)}. \tag{2.46}$$

In the next section, the EM algorithm is introduced, which is a further improvement over the Bayes's Postulate or the equidistribution of ignorance used by Richardson and Lucy.

### 2.6.2. Expectation-Maximization Algorithm

In this section, the EM is explained in terms the electro-optic terms and variables instead of the statistical terms. In 1976, Dempster, Laird, and Rubin introduced an iterative maximum likelihood from the incomplete data via the generalized EM algorithm. The term "incomplete data" stems from the existence of many-to-one mapping from one space to another space [58]. The observed data is called the "incomplete data" which is formed from the summation of the unobserved data or the "complete data" as shown below.

$$i(y) = \sum_{\forall x} \tilde{i}(y,x), \tag{2.47}$$

where $i(y)$ is the "incomplete" image function and $\tilde{i}(y,x)$ is the "complete" image function. In the subsequent sections, the term "data" also appears as the data functions, which is formed when the photon counting noise is introduced to the image functions. To

avoid confusion, the "incomplete data" defined by Dempster will be simply called "incomplete" for the remaining of this document. The term "data" will be reserved for describing the data function which is the output of the CCD.

Since the image function is the convolution of the PSF, *h*, with the object function, *o*, as shown in Equation (2.37), the complete image function can be expressed as a pixel-by-pixel multiplication of the PSF and the object function as shown in the equation below.

$$\tilde{i}(y,x) = h(y-x)o(x). \tag{2.48}$$

Depending on the situation, either or both of the object function and the PSF are unknown. To estimate the object function or the PSF or both, this approach takes the maximum likelihood estimation (MLE) of the complete image function. For deconvolution problems, the MLEs of the object function and the PSF are shown in Equation (2.49) and Equation (2.50). For blind deconvolution problems, the MLE of both is shown in Equation (2.51).

$$L(o) = \prod_{\forall x}\prod_{\forall y} \mathrm{P}\left(\tilde{i}(y,x)\right), \tag{2.49}$$

$$L(h) = \prod_{\forall x}\prod_{\forall y} \mathrm{P}\left(\tilde{i}(y,x)\right), \tag{2.50}$$

$$L(o,h) = \prod_{\forall x}\prod_{\forall y} \mathrm{P}\left(\tilde{i}(y,x)\right). \tag{2.51}$$

The EM algorithm requires maximizing the MLE functions. Instead of maximizing them directly, it is much simpler to maximize the logarithm (log) of the MLE functions. Due to the fact that log is a real ($\mathbb{R}$), one-to-one, and strictly monotonic function for any real ($\mathbb{R}$) argument greater than zero, maximizing the log of a function will give the same

maximum location as the maximizing the function itself. Therefore, the log is applied on the MLE function to obtain the log-likelihood function, $\ell$, as shown below.

$$\begin{aligned}\ell(o) &= \log L(o) \\ &= \sum_{\forall x}\sum_{\forall y} \log\left[\mathrm{P}\left(\tilde{i}(y,x)\right)\right].\end{aligned} \tag{2.52}$$

Until converges, the EM algorithm iteratively calculates the conditional expectation of the log-likelihood function in the Expectation Step (E-Step) and maximizes the resultant conditional expectation function in the Maximization Step (M-Step) to estimate the object function or the PSF or both. The E-Step and the M-Step are shown in Equation (2.53) and Equation (2.54) for estimating the object function. The system diagram for the EM algorithm is shown in Figure 2.8.

$$\text{E-Step: } Q(o) = \mathrm{E}\left[\ell(o)\middle|i(y)\right], \tag{2.53}$$

$$\text{M-Step: } \hat{o}(x) = \arg\max_{o(x)} Q(o). \tag{2.54}$$

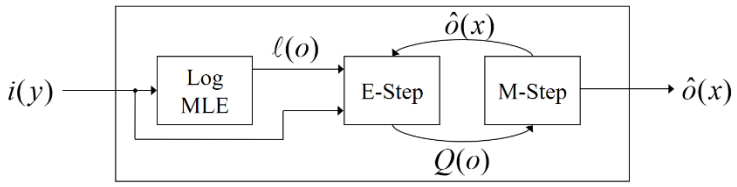


**Figure 2.8. Generalized Expectation-Maximization Algorithm for Estimating the Object Function.**

For blind deconvolution, the conditional expectation function from the E-Step will be maximized with respect to both the true object function, *o*, and the PSF, *h*. Even though

there is no requirement on the type of probability distribution function to be used for the image function, Dempster, Laird, and Rubin used an exponential random variable whose MLE function is easily computed, resulting in an easy computation for the E-Step and the M-Step. Therefore, in this section, the image function, *i*, is treated as an exponential random variable, which is not the case for the rest of this document. In the next section, the EM algorithm is derived for the data functions with Poisson distribution since the Poisson noise model is used for medical imaging devices and CCDs [22].

### 2.6.3. Expectation-Maximization in Medical Imaging

In 1982, Shepp and Vardi applied the EM algorithm for emission tomography, with a Poisson random variable. This section also uses the terms and variables that are consistent with the electro-optics instead of those from the medical imaging community. The electro-optics equivalent of Shepp and Vardi's problem is shown in Figure 2.9.

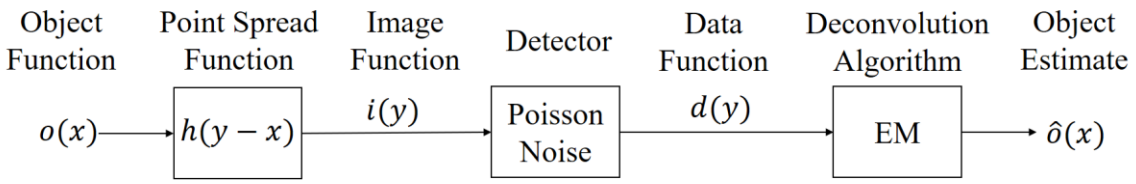


**Figure 2.9. Medical Image Reconstruction System Diagram.**

From the figure above, the data function is formed by introducing the photon counting noise to the image function. As a result, the relationship between the image function and the data function can be described as the following.

$$\mathrm{E}\left[d(y)\right] = i(y). \tag{2.55}$$

Since the sum of independent Poisson distributions is also a Poisson distribution, the complete data function, $\tilde{d}(y,x)$, and its probability mass function in terms of the complete image function, $\tilde{i}(y,x)$, are defined as the following.

$$d(y) = \sum_{\forall x} \tilde{d}(y,x), \tag{2.56}$$

$$\mathrm{E}\left[\tilde{d}(y,x)\right] = \tilde{i}(y,x), \tag{2.57}$$

$$\mathrm{P}\left(\tilde{d}(y,x)\right) = e^{-\tilde{i}(y,x)} \frac{\tilde{i}(y,x)^{\tilde{d}(y,x)}}{\tilde{d}(y,x)!}. \tag{2.58}$$

The MLE function and the log-likelihood functions are shown in Equation (2.59) and Equation (2.60). Since the Poisson distribution uses an exponential function, the natural logarithm (ln) is used in the log-likelihood function.

$$L(o) = \prod_{\forall x}\prod_{\forall y} \mathrm{P}\left(\tilde{d}(y,x)\right), \tag{2.59}$$

$$\begin{aligned} \ell(o) &= \sum_{\forall x}\sum_{\forall y} \ln\left[\mathrm{P}\left(\tilde{d}(y,x)\right)\right] \\ &= \sum_{\forall x}\sum_{\forall y} \ln\left[e^{-\tilde{i}(y,x)} \frac{\tilde{i}(y,x)^{\tilde{d}(y,x)}}{\tilde{d}(y,x)!}\right] \\ &= \sum_{\forall x}\sum_{\forall y}\left[-\tilde{i}(y,x) + \tilde{d}(y,x)\ln\tilde{i}(y,x) + \mathrm{A.T.}\right], \end{aligned} \tag{2.60}$$

where A.T. represents the additional term that does not affect the maximization [62]. In the subsequent equations, this A.T. term will be dropped. With the log-likelihood function derived, the conditional expectation given the incomplete data, $d(y)$, can be calculated as the following.

$$
\begin{aligned}
Q(o) &= \mathrm{E}\left[\ell(o)\middle|d(y)\right] \\
&= \mathrm{E}\left[\sum_{\forall x}\sum_{\forall y}\left[-\tilde{i}(y,x)+\tilde{d}(y,x)\ln\left(\tilde{i}(y,x)\right)-\ln\left(\tilde{d}(y,x)!\right)\right]\middle|d(y)\right] \\
&= -\sum_{\forall x}\sum_{\forall y}\mathrm{E}\left[\tilde{i}(y,x)\middle|d(y)\right]+\sum_{\forall x}\sum_{\forall y}\mathrm{E}\left[\tilde{d}(y,x)\ln\left(\tilde{i}(y,x)\right)\middle|d(y)\right].
\end{aligned}
\tag{2.61}
$$

Substituting $\tilde{i}(y,x)=h(y-x)o(x)$ from Equation (2.48) into the equation above gives us the following conditional expectation function.

$$
\begin{aligned}
Q(o) &= -\sum_{\forall x}\sum_{\forall y}\mathrm{E}\left[h(y-x)o(x)\middle|d(y)\right] \\
&\quad +\sum_{\forall x}\sum_{\forall y}\mathrm{E}\left[\tilde{d}(y,x)\ln\left(h(y-x)o(x)\right)\middle|d(y)\right] \\
&= -\sum_{\forall x}\sum_{\forall y}h(y-x)o(x)+ \\
&\quad +\sum_{\forall x}\sum_{\forall y}\ln\left(h(y-x)o(x)\right)\mathrm{E}\left[\tilde{d}(y,x)\middle|d(y)\right].
\end{aligned}
\tag{2.62}
$$

Since the object function and the PSF do not have probability distributions, their conditional expectation functions are just themselves. The complete data function, $\tilde{d}(y,x)$, also takes on the Poisson distribution [62]. In Equation (2.7), the conditional expectation function of a Poisson random variable given a sum of Poisson random variables including itself, $\mathrm{E}\left[X_k\middle|Y=X_1+\ldots+X_N\right]$, was derived. Since the incomplete data is the sum of complete data as shown in Equation (2.47) and both the incomplete data and the complete data are Poisson random variables, the conditional expectation function of a complete data given the incomplete data is given by the following equation.

$$
\begin{aligned}
\mathrm{E}\left[\tilde{d}(y,x)\middle|d(y)\right] &= \mathrm{E}\left[\tilde{d}(y,x)\middle|\sum_{\forall z\in X}\tilde{d}(y,z)\right] \\
&= \frac{\mathrm{E}\left[\tilde{d}(y,x)\right]}{\mathrm{E}\left[\sum_{\forall z\in X}\tilde{d}(y,z)\right]}d(y) \\
&= \frac{h(y-x)o(x)}{\sum_{\forall z\in X}h(y-z)o(z)}d(y) \\
&= \frac{h(y-x)o(x)}{i(y)}d(y).
\end{aligned}
\tag{2.63}
$$

In the EM algorithm, the new functions are iteratively updated from the estimations of the old functions. Thus, Equation (2.63) represents the old terms. Substituting Equation (2.63) into Equation (2.62) gives the following conditional expectation function, which is the E-Step in the EM algorithm.

$$
\begin{aligned}
Q(o) = &-\sum_{\forall x}\sum_{\forall y}h(y-x)o(x) \\
&+\sum_{\forall x}\sum_{\forall y}\left[\ln\left(h(y-x)o(x)\right)\left(\frac{h^{\mathrm{old}}(y-x)o^{\mathrm{old}}(x)}{i^{\mathrm{old}}(y)}d(y)\right)\right].
\end{aligned}
\tag{2.64}
$$

Next, the conditional expectation function is maximized with respect $o(x)$ by taking the derivative and setting it to zero.

$$
\begin{aligned}
\frac{\partial}{\partial o(x)}Q(o) &= 0 \\
-\sum_{\forall y}h(y-x)+\sum_{\forall y}\left[\frac{h(y-x)}{h(y-x)o(x)}\left(\frac{h^{\mathrm{old}}(y-x)o^{\mathrm{old}}(x)}{i^{\mathrm{old}}(y)}d(y)\right)\right] &= 0 \\
-1+\frac{1}{o(x)}\sum_{\forall y}\left(\frac{h^{\mathrm{old}}(y-x)o^{\mathrm{old}}(x)}{i^{\mathrm{old}}(y)}d(y)\right) &= 0 \\
o^{\mathrm{old}}(x)\sum_{\forall y}\left(\frac{h^{\mathrm{old}}(y-x)}{i^{\mathrm{old}}(y)}d(y)\right) &= o(x).
\end{aligned}
\tag{2.65}
$$

Therefore, the update equation for the new estimate for the object function becomes

$$o^{\text{new}}(x) = \arg\max_{o(x)} Q(o)$$
$$= o^{\text{old}}(x) \sum_{\forall y} \left( \frac{h^{\text{old}}(y-x)}{i^{\text{old}}(y)} d(y) \right). \tag{2.66}$$

Next, the techniques from this section are adapted and expanded for blind deconvolution when the data functions from multiple unknown PSFs are available.

### 2.6.4. Multi-Frame Blind Deconvolution of Astronomical Images

The statistical model used by Shepp and Vardi for the medical imaging detector took on a Poisson distribution, which is consistent with the statistical model of CCDs used in electro-optics [22]. In 1993, Schulz adapted Shepp and Vardi's algorithm and expanded for multi-frame blind deconvolution for astronomical images.

With multiple atmospheric PSFs, $\{h_k(x) : k = 1, ..., K\}$, where $K$ is the total number of PSFs, the corresponding image functions, $i_k(y)$, can be defined as the following.

$$i_k(y) = \sum_{x \in X} h_k(y-x) o(x) = \sum_{x \in X} \tilde{i}_k(y, x). \tag{2.67}$$

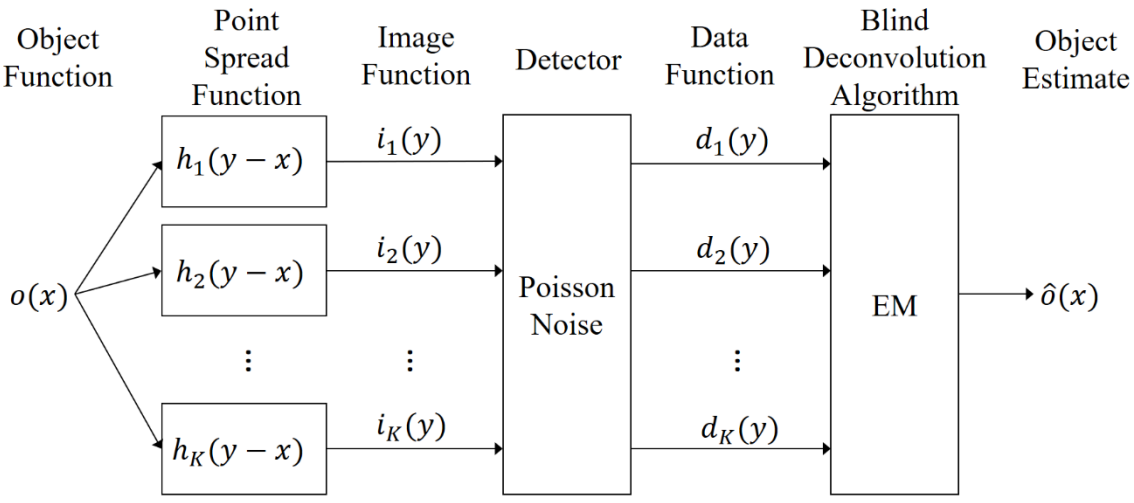


**Figure 2.10. System Diagram for Multi-Frame Blind Deconvolution.**

Using similar concepts as Shepp and Vardi, the incomplete image function for each frame, $i_k(y)$, can be expressed as a sum of a complete image function for each frame, $\tilde{i}_k(y,x)$.

$$i_k(y) = \sum_{x \in X} \tilde{i}_k(y,x), \tag{2.68}$$

$$\tilde{i}_k(y,x) = h_k(y-x)o(x). \tag{2.69}$$

The data functions for the complete data and incomplete data for each of the frames also assume Poisson distribution.

$$\mathrm{E}\left[d_k(y)\right] = i_k(y), \tag{2.70}$$

$$\mathrm{E}\left[\tilde{d}_k(y,x)\right] = \tilde{i}_k(y,x), \tag{2.71}$$

$$d_k(y) = \sum_{x \in X} \tilde{d}_k(y,x). \tag{2.72}$$

With multiple PSFs, the joint likelihood of the complete data, *L*, and the joint log-likelihood, $\ell$, can be expressed as the following.

$$L(o,h_k) = \prod_{\forall k}\prod_{\forall y}\prod_{\forall x} \mathrm{P}\left(\tilde{d}_k(y,x)\right), \tag{2.73}$$

$$\ell(o,h_k) = \ln L(o,h_k). \tag{2.74}$$

Next, the *Q*-function for the E-Step is derived similar to that shown in the previous subsection. See Equation (2.61) through Equation (2.64) for the detailed derivation of the *Q*-function for a single frame image processing.

$$
\begin{aligned}
Q(h_k, o) &= \mathrm{E}\left[ \ell(h_k, o) \middle| d_k(y) \right] \\
&= -\sum_{\forall k}\sum_{\forall x}\sum_{\forall y} h_k(y-x)o(x) \\
&\quad + \sum_{\forall k}\sum_{\forall x}\sum_{\forall y}\left[ \ln\left(h_k(y-x)o(x)\right)\left( \frac{h_k^{old}(y-x)o^{old}(x)}{i^{old}(y)} d_k(y) \right) \right].
\end{aligned}
\tag{2.75}
$$

Similar to Equation (2.65), the M-step maximizes the *Q*-function with respect to the objection function as shown in Equation (2.76). In addition, since this is a blind deconvolution problem, the M-step also maximizes the PSF functions as shown in Equation (2.77).

$$
\begin{aligned}
o^{\mathrm{new}}(x) &= \arg\max_{o(x)} Q(h_k, o) \\
&= \frac{o^{\mathrm{old}}(x)}{K} \sum_{\forall y}\left( \frac{h^{\mathrm{old}}(y-x)}{i^{\mathrm{old}}(y)} d_k(y) \right),
\end{aligned}
\tag{2.76}
$$

$$
\begin{aligned}
h_k^{\mathrm{new}}(x) &= \arg\max_{h_k(x)} Q(h_k, o) \\
&= \frac{1}{U_k^{\mathrm{new}}} h_k^{\mathrm{old}}(x) \sum_{\forall y}\left( \frac{o^{\mathrm{old}}(y-x)}{i^{\mathrm{old}}(y)} d_k(y) \right),
\end{aligned}
\tag{2.77}
$$

where $U_k^{\mathrm{new}}$ is some constant to ensure the $k^{\mathrm{th}}$ PSF sums up to unity. The two update equations above serve as the fundamental equations for this research. In the next section, the Gerchberg-Saxton algorithm is introduced as a phase retrieval method.

## 2.7. Gerchberg-Saxton Phase Retrieval Overview

Since the optical systems cannot measure the phase directly, the phase retrieval refers to recovering the two-dimensional phase of the optical system using only the intensity data. It has a wide variety of applications in optical engineering [66]. Even though there are many phase retrieval techniques available [67]–[69], this research uses the

iterative method proposed by Gerchberg and Saxton (GS) in 1971 [70]. Any phase retrieval method can be independently integrated with the blind deconvolution algorithm. The integration between the two is explained in the next chapter.

When using the GS algorithm, the assumption is made that the aperture function is known, *A*. In addition, the magnitude of $\mathscr{H}$ is known and calculated from Equation (2.33) and Equation (2.34) as shown in the equation below.

$$\left|\mathscr{H}(x)\right| = \sqrt{h(x)}\,, \tag{2.78}$$

The Fourier Transform of any function can be expressed in terms of its magnitude and phase as shown in the equation below.

$$\mathscr{H}(x) = \left|\mathscr{H}(x)\right| \exp\left(j\phi_{\mathscr{H}}(x)\right), \tag{2.79}$$

where $\phi_{\mathscr{H}}$ is the phase of the Fourier Transform associated with $\mathscr{H}$. Initially, there is no knowledge of the phases, $\phi_{\mathscr{H}}$ or $\phi$. However, the magnitude of $\mathscr{H}$ and the aperture function, *A*, are known. By iteratively going back-and-forth between the Fourier Transform and the spatial domain of the aperture, where the atmospheric turbulence introduces its effect, the GS algorithm estimates the atmospheric wavefront from the intensity of the PSF. A pseudocode and a visual representation as well as of the GS algorithm are shown in Equation (2.80) and Figure 2.11 respectively. The initialization and the number of iterations are explained in detail in the next chapter.

$$
\begin{aligned}
&\hat{\phi}(u) = \phi_{initial}(u) \\
&\text{iterate } \{ \\
&\quad \hat{\mathscr{H}}(x) = \mathfrak{F}\left\{ A(u)\exp\left( j\hat{\phi}(u) \right) \right\} \\
&\quad \hat{\phi}_{\mathfrak{F}}(x) = \text{angle}\left\{ \hat{\mathscr{H}}(x) \right\} \\
&\quad a(u) = \mathfrak{F}^{-1}\left\{ \sqrt{h(x)}\exp\left( j\hat{\phi}_{\mathfrak{F}}(x) \right) \right\} \\
&\quad \hat{\phi}(u) = \text{angle}\left\{ a(u) \right\} \\
&\}
\end{aligned}
\tag{2.80}
$$

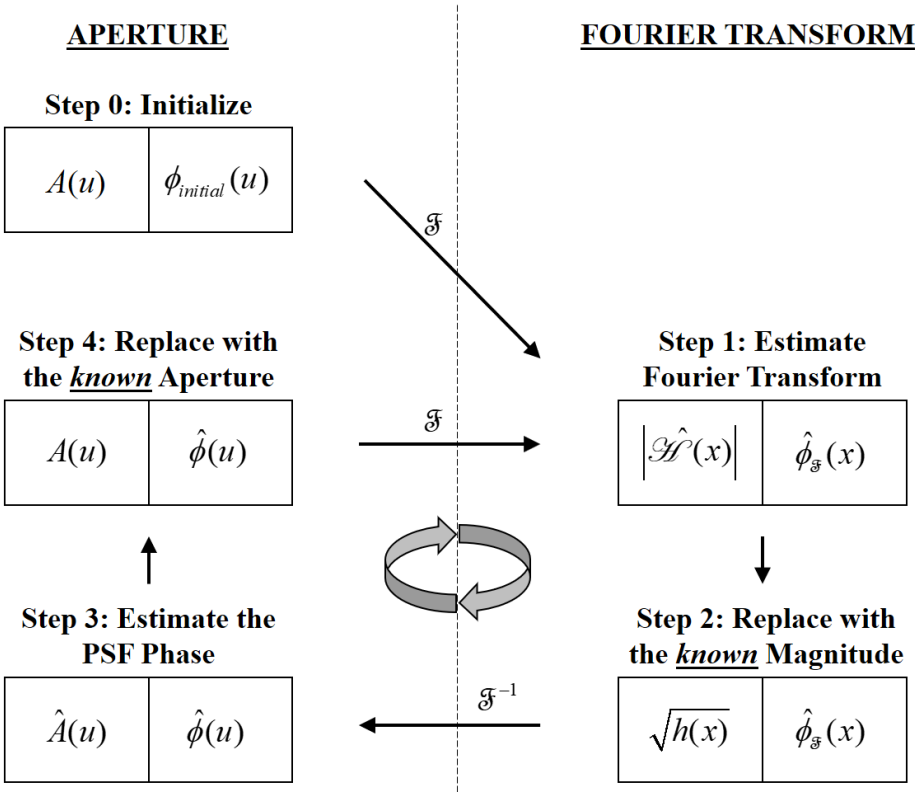


**Figure 2.11. A Visual Representation of the Gerchberg-Saxton Algorithm.**

## 2.8. Spatial Undersampling

Unless the narrow field-of-view (FOV) optical systems are used, the data functions from the CCDs could be undersampled. This section explains the process of spatial undersampling using a mathematical model and the steps that the object function passes through to generate an undersampled data function.

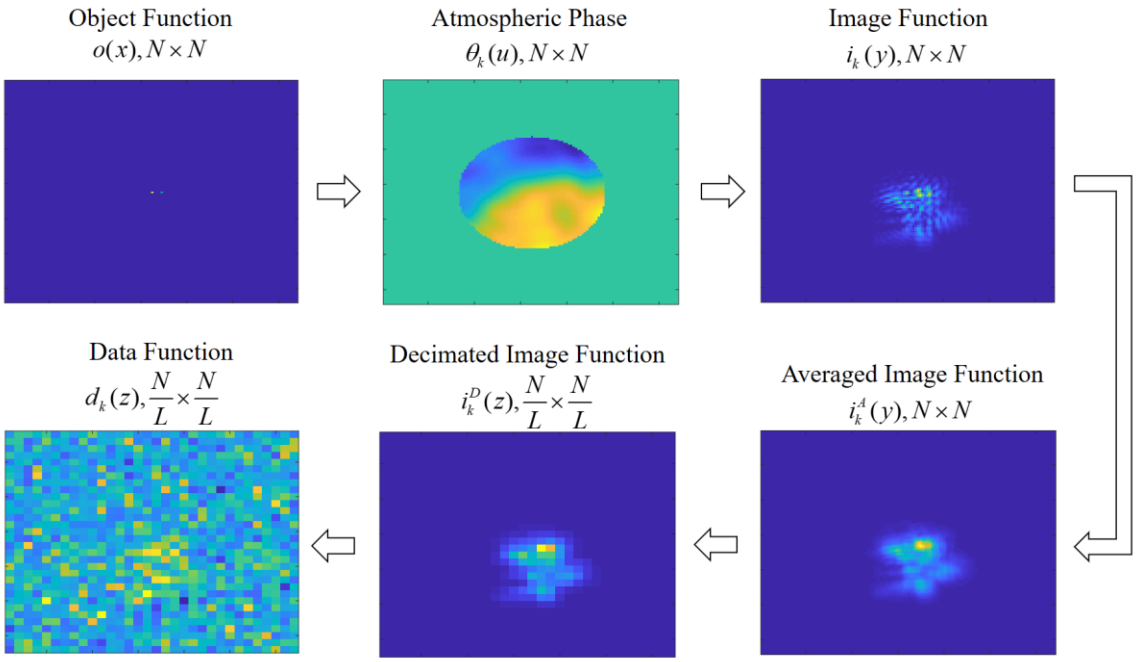


**Figure 2.12. The process of undersampling, where *N*×*N* represents the number of pixels in the functions sampled at the Nyquist rate and *L* is the undersampled factor.**

As shown in Figure 2.12, assume that the object function, $o(x)$, is sampled at the Nyquist rate, with $N \times N$ pixels in size. It passes through an atmospheric PSF, $h_k$, to

generate an image function, $i_k(y)$, which is also sampled at the Nyquist rate. The Nyquist sampling size, $\Delta$, is defined as shown in Equation (2.81), where $\lambda$ is the wavelength, $l$ is the distance between the exit pupil and the detector, and $D$ is the diameter of the source.

$$\Delta = \frac{\lambda l}{2D}. \tag{2.81}$$

The downsampling process starts with spatial averaging. The concept behind the averaging filter is to reduce "sharp" intensities by replacing the value of every pixel in the image function by the average of the intensity levels in the neighborhood defined by the filter [71]. The averaging effects of the square CCD sensors modeled as a rectangle average function, $r(y)$, as shown in the equation below [24], [68].

$$r(y) = \begin{cases} \frac{1}{L^2} & y \in \{(1,1),...,(L,L)\}, \\ 0 & \text{otherwise,} \end{cases} \tag{2.82}$$

where $L$ is the size of the neighborhood pixels. In addition, $L$ is the decimating factor. It is also assumed that $N/L$ is an integer. From Figure 2.12, the averaged image function, $i_k^A(y)$, is obtained from convolving the image function with the rectangle function as described in Equation (2.83). Note that if there are better models available to represent the averaging effects of particular CCDs, they should be used instead.

$$i_k^A(y) = \sum_{\forall \upsilon} r(\upsilon) i_k(y - \upsilon). \tag{2.83}$$

Next, the decimated image function, $\{i_k^D(z) : z \in Z\}$, is obtained by decimating the averaged image function as shown in the equation below, where $Z$ is the 2D discretized region over which the undersampled functions are defined.

$$i_k^D(z) = i_k^A(Lz) = \sum_{\forall \upsilon}\sum_{\forall x} o(x)r(\upsilon)h_k(Lz - x - \upsilon). \tag{2.84}$$

The undersampled data function, $d_k(z)$, can then be obtained by adding signal-dependent photon counting noise. The relationship can be described such that the Poisson mean of the data function is equal to the decimated image function.

$$\mathrm{E}\left[d_k(z)\right] = i_k^D(z). \tag{2.85}$$

In addition to the photon counting noise, the CCD also introduces signal-independent additive white Gaussian noise (AWGN). Because the AWGN is modeled as a zero-mean with some variance, $\sigma_n^2$, it does not affect the equation above. Since a wide FOV optical system will produce undersampled data, understanding the spatial undersampling process is important in compensating the effects.

## 2.9. Summary

In summary, this chapter introduced the background for the tools and methods that are used throughout this research. It explained the three major sources of image degradation: (1) the photon counting noise associated with the CCDs, (2) the atmospheric turbulence, and (3) the spatial undersampling. The chapter provided an overview of astronomical detectors and the Poisson noise associated with the CCDs. Then, the atmospheric model using the Zernike polynomials was introduced and the technique to simulate the atmospheric wavefronts was explained. Afterwards, this chapter provided the iterative Expectation-Maximization based Multi-Frame Blind Deconvolution to overcome the signal dependent Poisson noise and the Gerchberg-Saxton phase retrieval to recover

the atmospheric wavefronts from the intensity of the PSF. Finally, this chapter explained the process for spatial undersampling so that it can be taken into account for processing undersampled data functions.

# 3. Methodology

## 3.1. Chapter Overview

This chapter presents the methodology that is used throughout this research. It is built on the tools and techniques described in the previous chapter. It builds a model for each component of the imaging system, starting with an object function that contains the stellar objects and ending with the estimation and detection of the objects after the object function passes through an atmospheric turbulence, an imaging system, a detector, and a blind deconvolution algorithm with phase retrieval. This chapter also describes the eight assumptions made about the source, the detector, the point spread function (PSF), and the data.

## 3.2. Assumptions

***Assumption 1: The Fresnel far-field propagation criteria is met.***

In this research, the statistical image processing is conducted using two-dimensional (2D) Fourier transform. It is because the Fourier transforms are computationally efficient. In addition, the concept of light propagation can be easily simplified, understood, and implemented with the Fresnel propagation as a Fourier transform, compared to the Rayleigh-Somerfield propagation. However, in order for us to use the Fourier transform approach, the Fresnel criteria shown in Equation (3.1) must be met [39].

$$\frac{\pi}{4\lambda}\frac{D^4}{z^3} \ll 1, \tag{3.1}$$

where $z$ is the propagation distance, $D$ is the diameter of the object at the source plane, and $\lambda$ is the wavelength of the light.

For any digital imaging system, it is necessary that the Fresnel propagation is met for the propagation between the stellar object and the aperture of the telescope as well as that between the aperture of the telescope and the charge-coupled device (CCD) as shown Figure 3.1. This allows each propagation to be implemented as a Fourier transform, which in turn allows the entire propagation to be implemented as a Fourier transform.

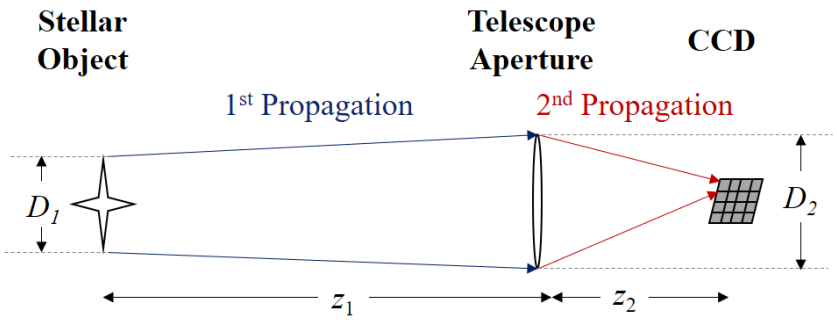


**Figure 3.1. The light propagation from the stellar object to the telescope aperture and then to the CCD.**

In Table 3.1, the Fresnel criteria is calculated for a 10 meter diameter stellar object that is in different orbital regimes, using an average visible wavelength of 550 nanometers. From the table, it can be seen that the Fresnel criteria is met for all cases.

For the second propagation, it is dependent on the application. The aperture diameters of a majority of the Space Domain Awareness (SDA) telescopes at the Starfire Optical Range (SOR) in New Mexico and the Air Force Maui Optical & Supercomputing Site (AMOS) in Hawaii are 1.6 meters or smaller, even though the SOR has a 3.5 meter telescope and the AMOS has 3.6 meter telescope respectively [72], [73]. Even though this

research does not use the data from these SDA telescopes, the Fresnel criteria is verified for each application.

**Table 3.1. The Fresnel Propagation Criteria for a 10 Meter Diameter Object**

| **Orbital Regime** | **Altitude (kilometer)** | **Fresnel Criteria** $\left(\frac{\pi}{4\lambda}\frac{D^4}{z^3}\right)$ |
|---|---|---|
| Low Earth Orbit | 180 – 2,000 | $2.5\times10^{-6} - 1.8\times10^{-9}$ |
| Medium Earth Orbit | 2,000 – 35,780 | $1.8\times10^{-9} - 3.1\times10^{-13}$ |
| High Earth and Geosynchronous Orbit | $\geq$ 35,780 | $\leq 3.1\times10^{-13}$ |

***Assumption 2: The imaging region is smaller than the isoplanatic patch size, where the imaging system is linear and shift-invariant.***

Imaging systems are seldom isoplanatic over their entire object field. However, it is possible to divide the field into small regions, the isoplanatic patches, such that the system is approximately linear and shift-invariant [39]. This allows the image function to be represented as the convolution between the object function and the PSF.

***Assumption 3: The effects of the atmospheric turbulence is frozen for short exposure images.***

In a digital imaging system, the CCD converts the photons into the photoelectrons by converting the intensity of the electromagnetic field that is incident on the CCD detector for some integration time as defined in the equation below [74].

$$I_{det}(y,t) = \frac{1}{T}\int_{t}^{t+T} \left|\mathbf{u}_{det}(y,\tau)\right|^2 d\tau\,, \tag{3.2}$$

where $I_{det}$ is the intensity functions, $\mathbf{u}_{det}$ is the field incident on the detector plane, $T$ is the integration time, $t$ is the time at which the intensity measurement begins, and $y$ is the two-dimensional region over which the intensity is defined.

In this research, the integration time, $T$, is assumed to be much less than one second so that the effects of the atmospheric turbulence is effectively frozen. Since the exposure time depends on the effective wind velocity, the definition of the short exposure time varies from 10 milliseconds or less as explained by Goodman [74], to 15 milliseconds as stated by Hirsch, Harmeling, Sra, and Scholkopf [75], and to 60 milliseconds as used by Howell and Horch to image and resolve a binary star system [76]. In Chapter 5, the short exposure time of 30 milliseconds is used in collected the laboratory data to be tested and validated for this research.

***Assumption 4: The atmospheric point spread functions are uncorrelated.***

The intensity functions are recorded at various times and they are simply called the image functions as defined in the equation below.

$$i_k(y) \triangleq I_{det}(y,t_k)\,, \tag{3.3}$$

where $i_k$ is the $k^{\text{th}}$ image function obtained from the CCD at time $t_k$ and $y$ is the 2D region over which the intensity is defined.

In this research, the assumption is made that enough time has elapsed since the previous image function was taken so that the atmospheric PSFs are no longer correlated.

A time separation of one second between the two consecutive images ($\Delta t_k \geq 1$ second) is considered to be sufficient to generate uncorrelated PSFs.

$$\Delta t_k = t_{k+1} - t_k \geq 1 \text{ sec}, \ \forall k = 1, ..., K-1. \tag{3.4}$$

***Assumption 5: The CCD introduces signal dependent Poisson noise.***

As mentioned in Section 2.3 and Section 2.6, the CCD introduces a Poisson photon counting noise, in such a way that the statistical mean of each output pixel is equal to the intensity of the same pixel on the image function. The resultant output function with a Poisson noise is called the data function, which is shown in the equation below.

$$\mathrm{E}_{\mathrm{Poisson}}\left[ d_k^{star}(y) \right] = i_k^{star}(y), \tag{3.5}$$

where $d_k^{star}$ is the noise-free data function, $i_k^{star}$ is the noise-free image function, and $y$ is the 2D region over which both the data function and the image function are defined.

***Assumption 6: The background noise also follows Poisson distribution.***

In addition, the CCD also introduces noise from the background light and dark current in the form of Poisson distribution [22], [77]. The noise data function, $d_k^{noise}$, from the background noise, $i_k^{noise}$, is shown in Equation (3.6).

$$\mathrm{E}_{\mathrm{Poisson}}\left[ d_k^{noise}(y) \right] = i_k^{noise}(y). \tag{3.6}$$

Since the sum of two Poisson distributions results in another Poisson distribution as explained in Section 2.3, the data function, $d_k$, is formed from the summation of the noise-free data function and the noise data function as shown in the equation below.

$$\begin{aligned} \mathrm{E}_{\mathrm{Poisson}}\left[d_k(y)\right] &= \mathrm{E}_{\mathrm{Poisson}}\left[d_k^{star}(y)\right] + \mathrm{E}_{\mathrm{Poisson}}\left[d_k^{noise}(y)\right] \\ &= i_k^{star}(y) + i_k^{noise}(y) \\ &= i_k(y). \end{aligned} \tag{3.7}$$

***Assumption 7: The light incident upon the CCD is temporally incoherent.***

The next assumption is that the light incident on the CCD is temporally incoherent even though it may be spatially coherent. Temporal coherence is related to the intrinsic spectrum bandwidth of the light source, while spatial coherence is affected by the size of the light source and the propagation distance [78]. The light from the sun is temporally incoherent and therefore the sun light reflecting from the object is also temporally incoherent.

Next, the spatial coherence of the sun light will be considered. According to the Van Cittert-Zernike theorem, the area of coherence, $A_c$, on the satellite body from the sun's spatially incoherent light is given by the following equation [74], [79].

$$A_c = \frac{\left(\bar{\lambda} z\right)^2}{A_s}, \tag{3.8}$$

where $\bar{\lambda}$ is the average wavelength, $z$ is the distance between the source and the destination of the light, and $A_s$ is the uniformly bright source area.

With the average distance between the sun and the earth is about 150 million kilometers [80] and the surface area of the sun's hemisphere is about $3.04\times10^{18}$ square meters [81], the area of coherence on the satellite body is calculated to be $4.16\times10^{-9}$ square

meter or a square with 0.473 microns on each side. An average visible wavelength of 550 nanometers is used in the calculation.

$$A_{c_1} = \frac{\left(550\times10^{-9}\,\mathrm{m}\times150\times10^{9}\,\mathrm{m}\right)^2}{3.04\times10^{18}\,\mathrm{m}^2} = 2.24\times10^{-9}\,\mathrm{m}^2 = \left(0.473\times10^{-6}\,\mathrm{m}\right)^2. \qquad (3.9)$$

In the calculation above, the distance between the earth and the geosynchronous orbit (GEO) of 35,780 kilometers is ignored since it is very small compared to that between the sun and the earth [82]. Since the area of coherence on the satellite body is very small compared to the size of the satellite, the light reflecting from the satellite body is assumed to be spatially incoherent.

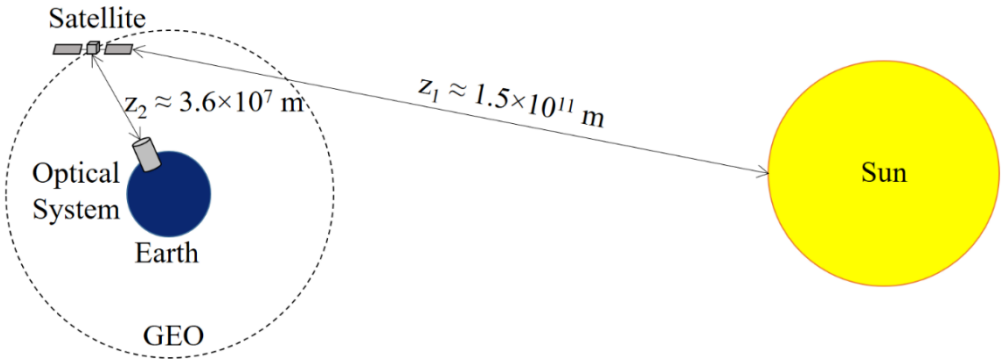


**Figure 3.2. Geometries among the Sun, the Earth, and the Optical System.**

Next, the area of coherence on the telescope aperture from the satellite body will be calculated. For this calculation, the propagation distance between the earth surface and the GEO orbit of 35,780 kilometers, an average visible light wavelength of 550 nanometer, and the satellite size of 100 square meters are used. With such parameters, the area of

coherence is calculated to be an area with about 1.1 meters in radius or 2.2 meters in diameter.

$$A_{c_2} = \frac{\left(550\times10^{-9}\,\text{m}\times35,780\times10^{3}\,\text{m}\right)^2}{10\ \text{m}^2} = 3.9\ \text{m}^2 = \pi\left(1.11\ \text{m}\right)^2. \tag{3.10}$$

Therefore, the whole satellite body becomes spatially coherent in the aperture plane even though it is temporally incoherent.

***Assumption 8: The intensities of the data function pixels are statistically independent.***

To justify the assumption that the pixels of the data function are statistically independent, the joint intensity function between two points will be calculated. The joint intensity function, **J**, is defined as the mutual coherence function, **Γ**, with no time difference as shown in the equation below [74].

$$\mathbf{J}\left(Q_1,Q_2\right) = \mathbf{\Gamma}\left(Q_1,Q_2,t=0\right), \tag{3.11}$$

where $Q_1$ and $Q_2$ are any two points on the receiver plane as shown in Figure 3.3.

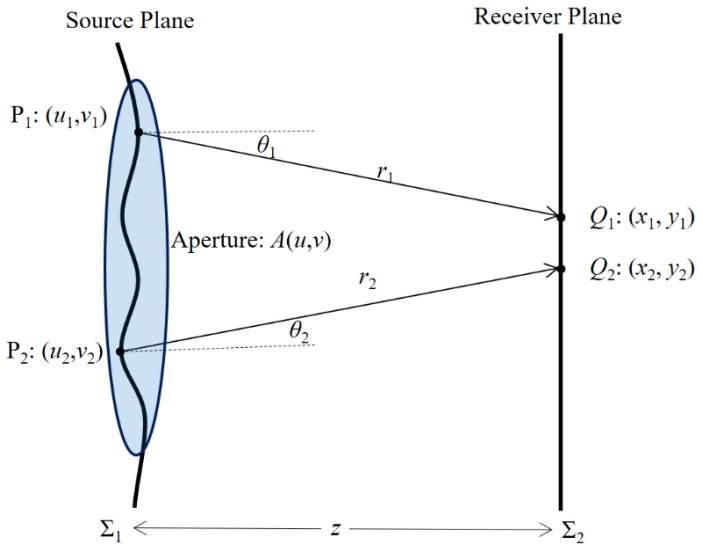


**Figure 3.3. Geometry for Propagation of Mutual Coherence.**

The mutual coherence function can be calculated using the following equation [74].

$$\mathbf{\Gamma}\left(Q_1,Q_2,0\right)=\iint_{\Sigma_1}\iint_{\Sigma_1}\mathbf{\Gamma}\left(P_1,P_2,\frac{r_2-r_1}{c}\right)\frac{\chi\left(\theta_1\right)}{\bar{\lambda}r_1}\frac{\chi\left(\theta_2\right)}{\bar{\lambda}r_2}dS_1dS_2, \tag{3.12}$$

where $P_1$ and $P_2$ are any two points on the source plane, $r_1$ is the distance between $Q_1$ and $P_1$, $r_2$ is the distance between $Q_2$ and $P_2$, $c$ is the speed of light, $\bar{\lambda}$ is the average wave length, $\chi$ is the obliquity factor with value between 0 and 1, $\theta_1$ and $\theta_2$ represent the angle between the normal of the surface and the propagation path, $\Sigma_1$ represents the area where $P_1$ and $P_2$ can exist, and $S_1$ and $S_2$ represent the integration variables for the source and receiver plane respectively. To make the calculation simple, the obliquity factors along with their denominators from Equation (3.12) are omitted since they will only scale the whole joint intensity function by a factor of some constant. The simplified equation can be written as the following equation.

$$\begin{aligned}\mathbf{J}(Q_1,Q_2) &= \iint_{\Sigma_1}\iint_{\Sigma_1}\boldsymbol{\Gamma}\left(P_1,P_2,\frac{r_2-r_1}{c}\right)dS_1dS_2 \\ &= \iint_{\Sigma_1}\iint_{\Sigma_1}\mathrm{E}\left[\mathbf{u}\left(P_1,t+\frac{r_2-r_1}{c}\right)\mathbf{u}^*(P_2,t)\right]dS_1dS_2,\end{aligned} \tag{3.13}$$

where **u** and **u*** are the field and its complex conjugate and *t* is the time variable. The field can be expressed in terms of its phasor amplitude and phase. Since the source field is confined to an artificial aperture placed on the source, the field is multiplied by the aperture transmittance function as shown in the equation below.

$$\mathbf{u}(P,t) = A(P)U(P,t)\exp(-j2\pi\bar{v}t), \tag{3.14}$$

where *A* is the aperture transmittance function as described in Equation (2.30), *U* is the phasor amplitude, and $\bar{v}$ is the average frequency. Using Equation (3.14), the correlation function from Equation (3.13) can be simplified as the following.

$$\begin{aligned}\mathrm{E}\left[\mathbf{u}\left(P_1,t+\frac{r_2-r_1}{c}\right)\mathbf{u}^*(P_2,t)\right] &= \mathrm{E}\left[A(P_1)U\left(P_1,t+\frac{r_2-r_1}{c}\right)e^{-j2\pi\bar{v}\left(t+\frac{r_2-r_1}{c}\right)}A^*(P_2)U^*(P_2,t)e^{j2\pi\bar{v}t}\right] \\ &= \mathrm{E}\left[U\left(P_1,t+\frac{r_2-r_1}{c}\right)U^*(P_2,t)\right]A(P_1)A^*(P_2)e^{-j2\pi\bar{v}\left(\frac{r_2-r_1}{c}\right)}.\end{aligned} \tag{3.15}$$

Recognizing that $P_1$ and $P_2$ are drawn from a spatially coherent field as described in Assumption 7, the correlation between $P_1$ and $P_2$ becomes a constant, which will be omitted to simplify the derivation. The average frequency, $\bar{v}$, divided by the speed of light, *c*, is also replaced with an average wavelength, $\bar{\lambda}$.

$$\mathrm{E}\left[\mathbf{u}\left(P_1,t+\frac{r_2-r_1}{c}\right)\mathbf{u}^*(P_2,t)\right] = A(P_1)A^*(P_2)e^{-j2\pi\bar{v}\left(\frac{r_2-r_1}{c}\right)}. \tag{3.16}$$

Substituting Equation (3.16) into Equation (3.13), the joint intensity function can be simplified as the following. Note that the scale factor is omitted to simplify the derivation.

$$\mathbf{J}(Q_1,Q_2)=\iint_{\Sigma_1}\iint_{\Sigma_1} A(P_1)A^*(P_2)e^{-j2\pi\left(\frac{r_2-r_1}{\bar{\lambda}}\right)}dS_1dS_2. \tag{3.17}$$

In the previous sections and chapters, single variables, such as *x* and *y*, were used to represent 2D regions. For the following equations from Equation (3.18) to Equation (3.28), the following variable pairs will be used to represent the coordinates on the source and detector planes as shown in Table 3.2.

With the two variable coordinates, Equation (3.17) can be expressed as shown in Equation (3.18).

$$\mathbf{\Gamma}(x_1,y_1,x_2,y_2)=\iint_{\Sigma_1}\iint_{\Sigma_1} A(P_1)A(P_2)\exp\left[-j\frac{2\pi}{\bar{\lambda}}(r_2-r_1)\right]du_1dv_1du_2dv_2. \tag{3.18}$$

**Table 3.2. Mapping Between Single Variable Coordinates and Two Variable Coordinates.**

| Single Variable Coordinate | Corresponding Two Variable Coordinate |
|---|---|
| $P_1$ | $(x_1,y_1)$ |
| $P_2$ | $(x_2,y_2)$ |
| $Q_1$ | $(u_1,v_1)$ |
| $Q_2$ | $(u_2,v_2)$ |

Next, the difference between the two distances will calculated as shown below.

$$
\begin{aligned}
r_2 - r_1 &= \|P_1 - Q_1\| - \|P_2 - Q_2\| \\
&= \sqrt{z^2 + (u_1 - x_1)^2 + (v_1 - y_1)^2} - \sqrt{z^2 + (u_2 - x_2)^2 + (v_2 - y_2)^2} \\
&= z\sqrt{1 + \frac{(u_1 - x_1)^2 + (v_1 - y_1)^2}{z^2}} - z\sqrt{1 + \frac{(u_2 - x_2)^2 + (v_2 - y_2)^2}{z^2}}.
\end{aligned} \tag{3.19}
$$

Using the Taylor series expansion, the square root terms can be approximated as the following.

$$
\begin{aligned}
r_2 - r_1 &\approx z\left(1 + \frac{(u_1 - x_1)^2 + (v_1 - y_1)^2}{2z^2}\right) - z\left(1 + \frac{(u_2 - x_2)^2 + (v_2 - y_2)^2}{2z^2}\right) \\
&= \frac{(u_1 - x_1)^2 + (v_1 - y_1)^2}{2z} - \frac{(u_2 - x_2)^2 + (v_2 - y_2)^2}{2z} \\
&= \frac{u_1^2 - 2u_1x_1 + x_1^2 + v_1^2 - 2v_1y_1 + y_1^2}{2z} - \frac{u_2^2 - 2u_2x_2 + x_2^2 + v_2^2 - 2v_2y_2 + y_2^2}{2z} \\
&= \frac{u_1^2 - u_2^2 + v_1^2 - v_2^2 + x_1^2 - x_2^2 + y_1^2 - y_2^2 - 2u_1x_1 - 2v_1y_1 + 2u_2x_2 + 2v_2y_2}{2z}.
\end{aligned} \tag{3.20}
$$

Next, the substitutions $x_2 = \Delta_x + x_1$ and $y_2 = \Delta_y + y_1$ are used to simplify Equation (3.20).

$$
\begin{aligned}
r_2 - r_1 \approx{}& \frac{u_1^2 - u_2^2 + v_1^2 - v_2^2}{2z} + \frac{x_1(u_2 - u_1) + y_1(v_2 - v_1) + u_2\Delta_x + v_2\Delta_y}{z} \\
&- \frac{x_1\Delta_x + y_1\Delta_y}{z} - \frac{\Delta_x^2 + \Delta_y^2}{2z}.
\end{aligned} \tag{3.21}
$$

If the lenses are used to focus the light, the quadratic term will vanish. In addition, since the mutual coherence function is integrated over the surface of the source plane, the receiver plane variables can be pulled out of the integral. Since these terms represent some constant value, they are omitted to simplify the derivation.

$$
[r_2 - r_1]_{\text{integral}} = \frac{x_1(u_2 - u_1) + y_1(v_2 - v_1) + u_2\Delta_x + v_2\Delta_y}{z}. \tag{3.22}
$$

Next, the substitutions $\Delta_u = u_2 - u_1$ and $\Delta_v = v_2 - v_1$ are used to further simplify the difference between the two distances.

$$\left[r_2 - r_1\right]_{\text{integral}} = \frac{x_1\Delta_u + y_1\Delta_v + u_2\Delta_x + v_2\Delta_y}{z}. \tag{3.23}$$

When the simplified version from Equation (3.23) is substituted into Equation (3.18), the joint intensity function can be expressed as shown in the equation below.

$$\mathbf{J}\left(\Delta_x, \Delta_y\right) = \iint\iint A(u_2 - \Delta_u, v_2 - \Delta_v) A(u_2, v_2) e^{-j\frac{2\pi}{\bar{\lambda}}\left(\frac{x_1\Delta_u + y_1\Delta_v + u_2\Delta_x + v_2\Delta_y}{z}\right)} d\Delta_u d\Delta_v du_2 dv_2 \tag{3.24}$$

Next, the above equation is regrouped and rearranged the integral with respect to the integration variables.

$$\mathbf{J}\left(\Delta_x, \Delta_y\right) = \iint\left(\iint A\left(-\left(\Delta_u - u_2\right), -\left(\Delta_v - v_2\right)\right) e^{-j\frac{2\pi}{\bar{\lambda}}\left(\frac{x_1\Delta_u + y_1\Delta_v}{z}\right)} d\Delta_u d\Delta_v\right) A(u_2, v_2) e^{-j\frac{2\pi}{\bar{\lambda}}\left(\frac{u_2\Delta_x + v_2\Delta_y}{z}\right)} du_2 dv_2. \tag{3.25}$$

Recognizing that the inner integral is a Fourier transform of the aperture function, the integral is calculated as the following.

$$\begin{aligned}\mathbf{J}\left(\Delta_x, \Delta_y\right) &= \iint\left(\mathscr{F}\left(-\frac{x_1}{\bar{\lambda}z}, -\frac{y_1}{\bar{\lambda}z}\right) e^{-j\frac{2\pi}{\bar{\lambda}}\left(\frac{x_1u_2 + y_1v_2}{z}\right)}\right) A(u_2, v_2) e^{-j\frac{2\pi}{\bar{\lambda}}\left(\frac{u_2\Delta_x + v_2\Delta_y}{z}\right)} du_2 dv_2 \\ &= \mathscr{F}\left(-\frac{x_1}{\bar{\lambda}z}, -\frac{y_1}{\bar{\lambda}z}\right) \iint A(u_2, v_2) e^{-j\frac{2\pi}{\bar{\lambda}}\left(\frac{u_2\left(\Delta_x + x_1\right) + v_2\left(\Delta_y + y_1\right)}{z}\right)} du_2 dv_2,\end{aligned} \tag{3.26}$$

where $\mathscr{F}$ is the Fourier transform of the aperture function, *A*. Also, recognizing that the remaining integral is also another Fourier transform of the aperture function, the joint intensity function can be simplified as shown in the equation below.

$$\mathbf{J}\left(\Delta_x, \Delta_y\right) = \mathscr{F}\left(-\frac{x_1}{\bar{\lambda}z}, -\frac{y_1}{\bar{\lambda}z}\right) \mathscr{F}\left(\frac{\Delta_x + x_1}{\bar{\lambda}z}, \frac{\Delta_y + y_1}{\bar{\lambda}z}\right). \tag{3.27}$$

If the aperture function is an even function, then its discrete Fourier transform is also an even function. Therefore, the above equation can be written as the following.

$$\mathbf{J}\left(\Delta_x, \Delta_y\right) = \mathscr{H}\left(\frac{x_1}{\lambda z}, \frac{y_1}{\lambda z}\right) \mathscr{H}\left(\frac{x_1 + \Delta_x}{\lambda z}, \frac{y_1 + \Delta_y}{\lambda z}\right). \tag{3.28}$$

From Equation (3.28), it can be easily seen that the joint intensity function is a product of the Fourier transform of the aperture function with the frequency shifted version of itself. As a visual illustration, the cross section of the Fourier transform of a circular aperture, the frequency shifted versions of the Fourier transform, and the joint intensity function are shown in Figure 3.4 and Figure 3.5. From these two figures, when the two points from the same scene are close together, there exists joint intensity between the two points. However, when the two points from the same scene are located far apart, their joint intensity becomes very small, meaning the two points are not correlated. In addition, as explained in Sections 2.4 and 2.5, since statistically independent random variables are used in generating uncorrelated PSFs, the assumption is made that different pixels in the data function are statistically independent.

$$\mathrm{P}\left(d_k(x_1, y_1) d_{k'}(x_2, y_2)\right) = \mathrm{P}\left(d_k(x_1, y_1)\right) \mathrm{P}\left(d_{k'}(x_2, y_2)\right), \forall k \neq k', (x_1, y_1) \neq (x_2, y_2). \tag{3.29}$$

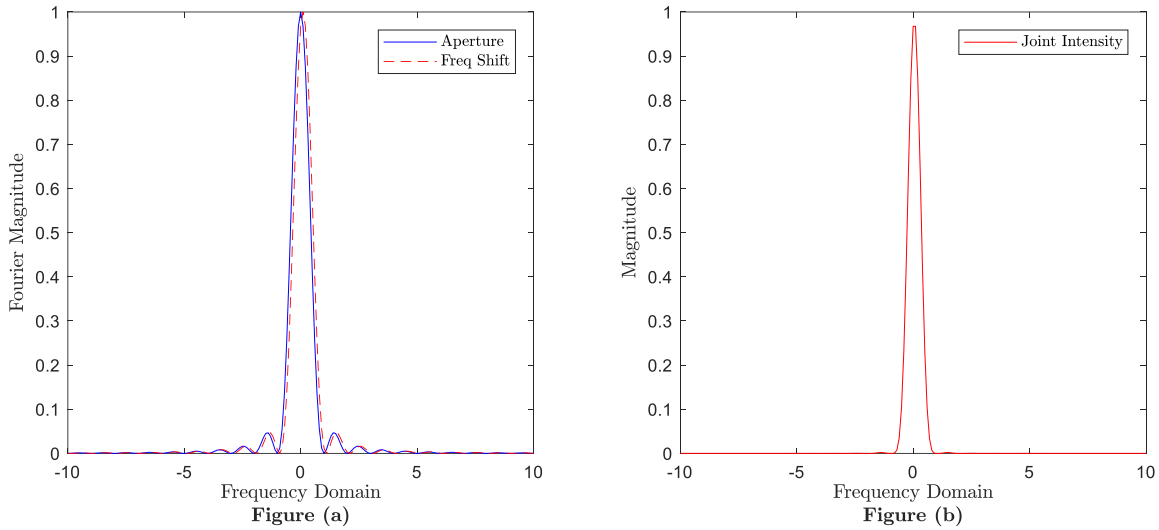


**Figure 3.4. The joint intensity function is shown in Figure (a) as the product of the Fourier transform of an aperture with a small frequency shift from Figure (b).**

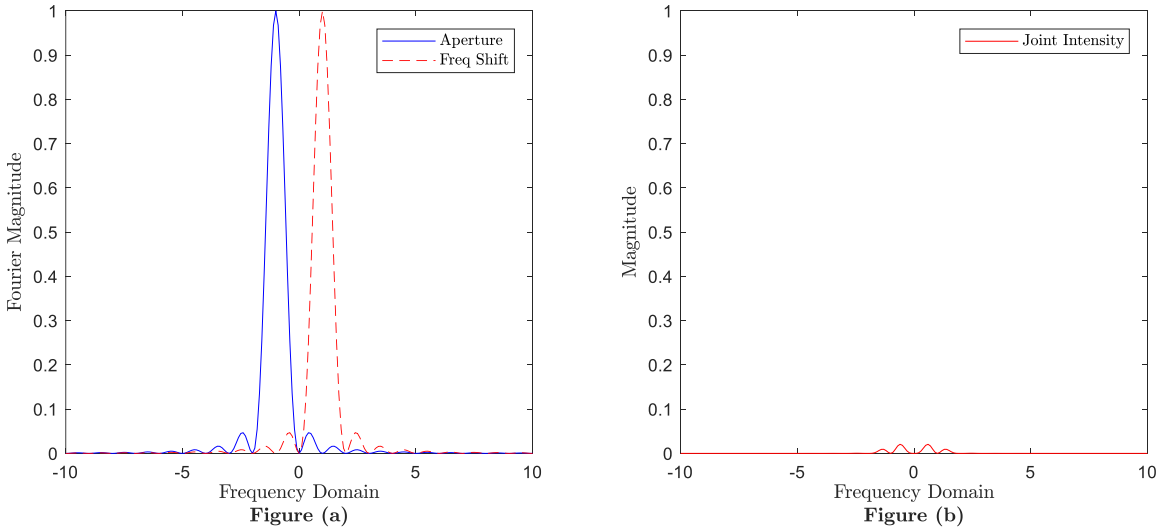


**Figure 3.5. The joint intensity function is shown in Figure (a) as the product of the Fourier transform of an aperture with a large frequency shift from Figure (b).**

### 3.3. Computer Generated Data Model

In this research, the algorithms are tested and validated with computer simulated data first. All implementations are completed in the MATLAB programming language on a standard desktop. The MATLAB built-in functions are used throughout this research.

#### 3.3.1. Detector Model

For modeling and simulation, the size of the detector is chosen to be a square $N \times N$ pixel plane. For the computer generated data, $N = 64$ is chosen. Even though the number chosen is arbitrary, it provides a size large enough to simulate the atmospheric PSF while being small enough to test and validate the iterative blind deconvolution algorithm without overtaxing the computational resources. In this research, the object function, the image function, the PSF, and the data function have the same size.

#### 3.3.2. Atmospheric Point Spread Function

In order to create an atmospheric PSF, an aperture function is first generated on the $N \times N$ plane. The diameter of aperture is set to be $N/2$ pixels. The resultant aperture function, $A(u)$, is shown in Figure 3.6. The diffraction limited PSF, $h_{DL}$, is generated without any phase aberration as shown in Equation (3.30). The diffraction limited optical transfer function (OTF), $H_{DL}$, is the Fourier transform of the diffraction limited PSF, as shown in Equation (3.31). The diffraction limited OTF and PSF are shown in Figure 3.7.

$$h_{DL}(x) = \left| \mathfrak{F}\{A(u)\} \right|^2, \tag{3.30}$$

$$H_{DL}(f) = \mathfrak{F}\{h_{dl}(x)\}. \tag{3.31}$$

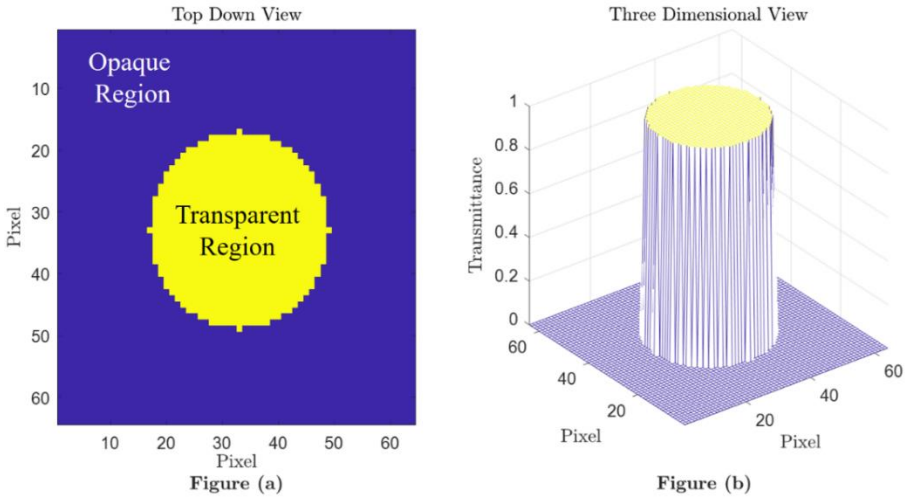


**Figure 3.6. Aperture Transmittance Function (*N* = 64).**

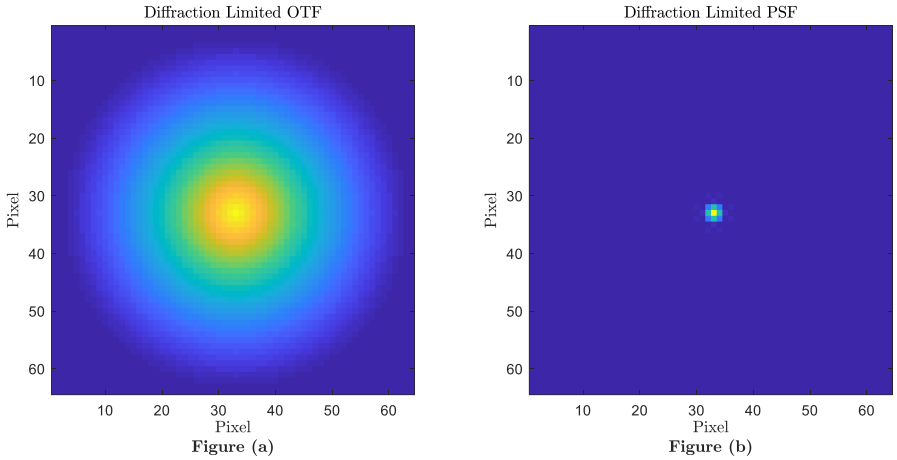


**Figure 3.7. Diffraction Limited OTF and PSF (*N* = 64).**

For the atmospheric PSF, the phase from atmospheric turbulence is combined with the aperture function. To generate an atmospheric turbulence model, the turbulence strength is needed to be defined, which the ratio between the diameter of the telescope aperture, *D*, and the Fried's seeing parameter, $r_0$ [83], [84]. For the simulation, the turbulence strength is selected to be two, $(D/r_0 = 2)$. With a typical median $r_0$ value of

0.16 meters at the best astronomical sites [85], the chosen strength can be achieved with any telescope with a 0.32 meter diameter. An atmospheric wavefront is generated using the Zernike method as described in Section 2.4. Since an infinitely many Zernike coefficients cannot be generated for modeling and simulation, a total of $p = 100$ Zernike coefficients are used in generating each wavefront. Even though this choice of $p$ is sufficient for $(D/r_0 = 2)$, higher number of Zernike coefficients should be used for stronger turbulence [83], [84]. Using these parameters, a Zernike covariance matrix, **C**, is generated using Equation (2.20). Then, a set of $p = 100$ independent zero-mean, unit-variance Gaussian random numbers are generated for each atmospheric wavefront. A set of random numbers, $\mathbf{N}_{100}$, is first generated from independent and identically distributed zero-mean, unit-variance Gaussian distribution.

$$\mathbf{N}_{100} = \left[n_2 \; n_3 \; \ldots \; n_{100}\right]^T, \text{where } n_k \sim \text{Gaussian}\left(\mu = 0, \sigma = 1\right), \forall k \in \left\{2, \ldots, 100\right\}, \tag{3.32}$$

where $\mu$ and $\sigma$ are the mean and variance of the Gaussian random variable. As described in Section 2.4, the set of random numbers is multiplied with the Cholesky decomposition, **U**, of the Zernike covariance matrix, **C**, to generate a set of Zernike coefficients, $\mathbf{A}_{100}$, as shown in the equation below.

$$\mathbf{A}_{100} = \mathbf{U}^T \mathbf{N}_{100} \text{ where } \mathbf{C} = \mathbf{U}^T \mathbf{U}, \; \mathbf{A}_{100} = \left[a_2 \; a_3 \; \ldots \; a_{100}\right]^T. \tag{3.33}$$

Next, a linear combination of the Zernike polynomials with the Zernike coefficients generates one atmospheric wavefront as explained in Section 2.4 and in Equation (3.34). Since the first Zernike polynomial is a constant or piston term, which represents a time delay, it is not used in calculation. The atmospheric wavefront is then combined with the

aperture function to generate the atmospheric PSF as explained in Section 2.5. For the convenience for the readers, the two main equations from Sections 2.4 and 2.5 are shown again in the equations below.

$$\phi(u) = \sum_{k=2}^{100} a_k Z_k(u), \tag{3.34}$$

$$h(x) = \frac{1}{U}\left|\mathfrak{F}\left\{A(u)e^{j\phi(u)}\right\}\right|^2, \tag{3.35}$$

where $\phi$ is the atmospheric wavefront, and $a_k$ and $Z_k$ is the $k^{\text{th}}$ Zernike coefficient and polynomial, $u$ is the two-dimensional discrete region over which the wavefront is defined, $h$ is the atmospheric PSF, $A$ is the aperture function, $\mathfrak{F}$ is the Fourier transform, $x$ is the 2D discrete region over which the PSF is defined, and $U$ is some constant to ensure that the PSF is a conservative process and sums up to unity. Figure 3.8 shows a comparison among an aberration free PSF and a set of five atmospheric PSFs generated with the atmospheric turbulence strength of $D/r_0 = 2$.

**Table 3.3. Parameters for Computer Simulated Data for a Scenario.**

| **Parameter** | **Value** |
|---|---|
| Aperture Diameter to Fried's Seeing Parameter Ratio $(D/r_0)$ | 2 |
| Number of Zernike Polynomials to Generate Atmosphere PSF | 100 |
| Number of Photoelectrons for the Bright Object | 10,000 |
| Number of Photoelectrons for the Dim Object | 1,000 – 25 |
| Number of Photoelectrons for the Background | 10 |
| Number of Pixels in Detector | 64×64 |
| Number of Pixels Separation Between the Bright and Dim Objects | 1 |

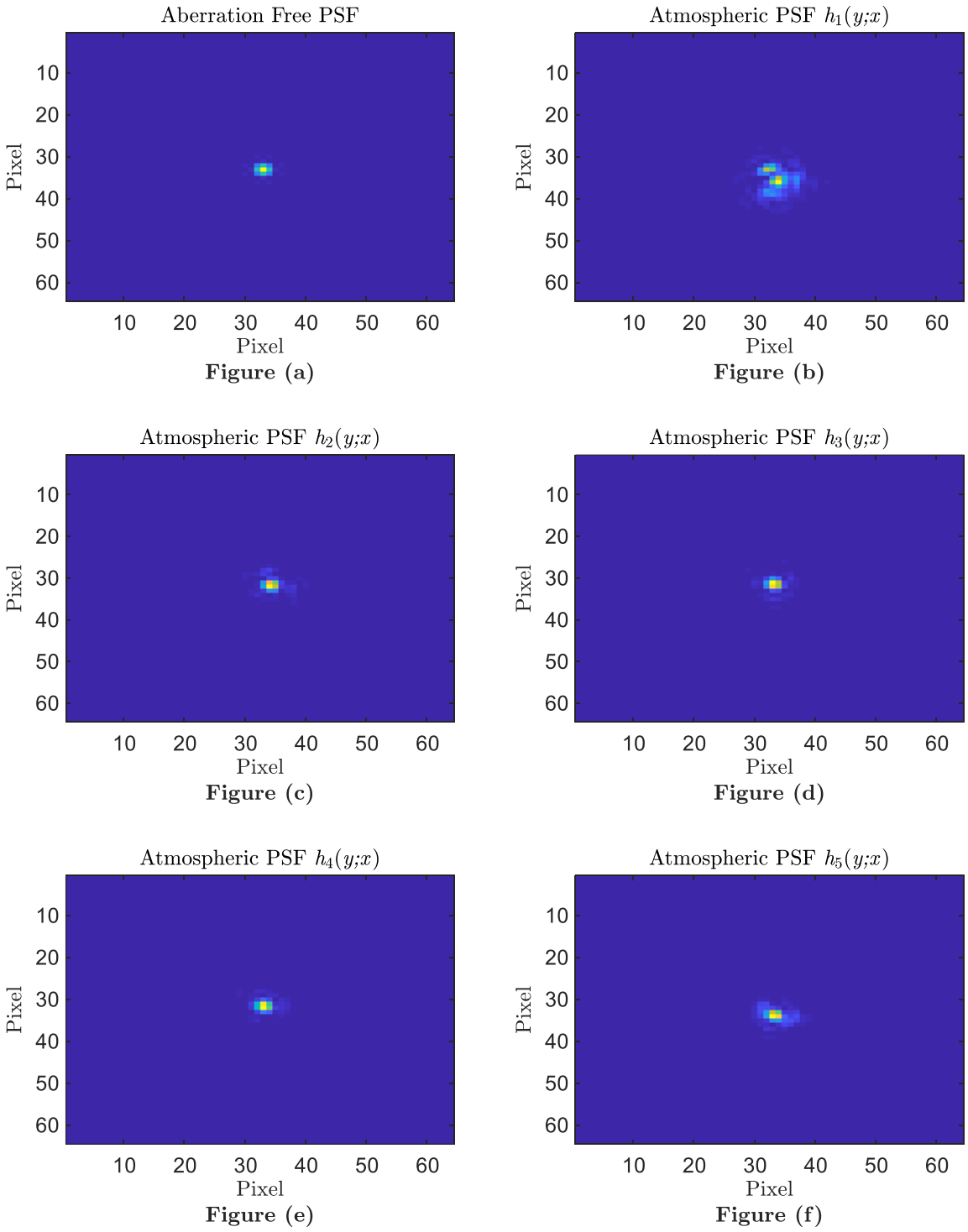


**Figure 3.8. Comparison among aberration free PSF (Fig. a) and the five atmospheric PSFs (Figs. b – f).**

### 3.3.3. Mean Square Residual Phase Error

The mean square residual phase error for using a finite number of Zernike coefficients, $\Delta_p$, is given by Equation (3.24) [37]. For $p = 100$, the mean square residual phase error is calculated to be $\Delta_{100} \approx 0.0176 \text{ rad}^2$.

$$\Delta_p \approx 0.2944 p^{-\sqrt{3}/2} \left( D / r_0 \right)^{5/3}. \tag{3.36}$$

The standard deviation of the phase error is calculated to be the following.

$$\sqrt{\Delta_{100}} = \sqrt{0.0176 \text{ rad}^2} \left( \frac{\lambda}{2\pi \text{ rad}} \right) = 0.0211\lambda. \tag{3.37}$$

This value is much less than the surface irregularities reported by lens manufacturer at $\lambda / 4$ [86].

### 3.3.4. Object Function

Using the 64×64 pixel detector, the bright object is placed at the center of the detector at pixel (32, 32). The dim object is placed two pixels away or with one pixel separation at (32, 34) as shown in Figure 3.9. The intensity of the bright object is set to be 10,000 photoelectrons. The intensity of the dim object is varied from 1,000 to 25 photoelectrons so that the algorithm could be first tested and validated in an easier scenario before using in more challenging scenarios as shown in Table 3.3. The scenarios for the object function are shown in Table 3.4. The background level of 10 photoelectrons is added to all pixels including the bright object and the dim object.

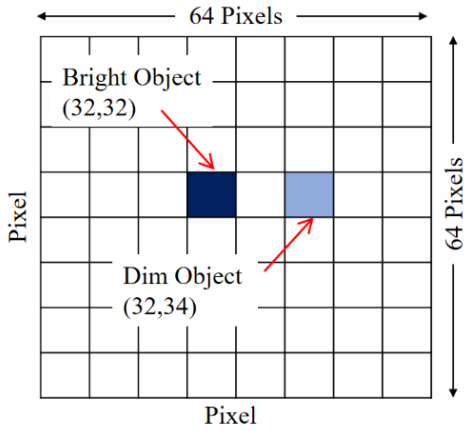


**Figure 3.9. An Object Function.**

**Table 3.4. Scenarios for the Object Function in Photoelectron Count.**

| Scenario | Bright Object | Dim Object | Background |
|---|---|---|---|
| Scenario 1 | 10,000 | 1,000 | 10 |
| Scenario 2 | 10,000 | 500 | 10 |
| Scenario 3 | 10,000 | 250 | 10 |
| Scenario 4 | 10,000 | 125 | 10 |
| Scenario 5 | 10,000 | 100 | 10 |
| Scenario 6 | 10,000 | 75 | 10 |
| Scenario 7 | 10,000 | 50 | 10 |
| Scenario 8 | 10,000 | 25 | 10 |

### 3.3.5. Image Function

The image function is generated by the convolution of the object function with the atmospheric PSF as described in Section 2.6. With the five atmospheric PSFs for each dataset, five image functions are generated from each object function.

### 3.3.6. Data Function

The data function is formed from introducing signal dependent Poisson noise the image function in such a way that the mean of the data function is equivalent to the image function as shown in Equation (3.7). Figure 3.10 shows the comparison between two object functions that pass through two random different atmospheric PSFs to form two different data functions. The first and second columns show the data functions formed from an object functions using the object intensities from Scenario 1 and Scenario 2 respectively.

### 3.3.7. Dataset Generation

As shown in Figure 2.10, an object function passes through five atmospheric PSFs to generate five image functions, which in turn generates five data functions. These five data functions make up a dataset. In order to generate statistics on the performance of the blind deconvolution algorithm, a total of 500 datasets are generated for each scenario. A scenario is defined as a unique combination of the intensities of the object. Table 3.4 shows the scenarios for the object functions, along with the intensities for the bright object, the dim object, and the background. All scenarios in Table 3.4 are generated using the atmospheric turbulence strength of $D/r_0 = 2$ and $p = 100$ Zernike polynomials to generate the atmospheric PSFs.

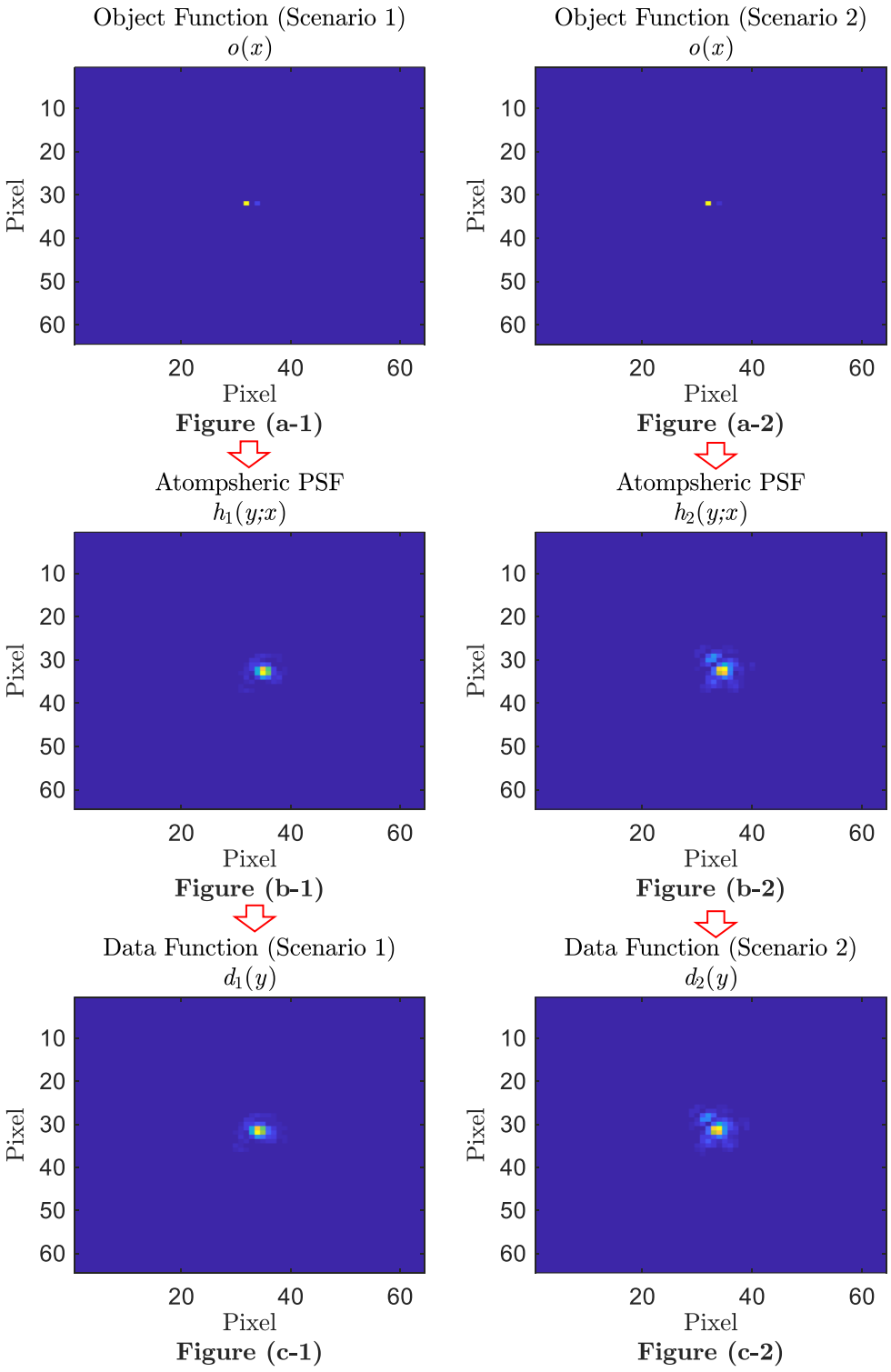


**Figure 3.10. Two data functions formed from two different object functions, passing through two different atmospheric PSFs.**

### 3.4. Signal Processing Model

For this research, the combination of the Multi-Frame Blind Deconvolution (MFBD) approach by Schulz and the Gerchberg-Saxton (GS) phase retrieval algorithm is used as the baseline. It is because the MFBD has been adopted by the United States Air Force for its Space Domain Awareness applications [12], [13]. In addition, it has been used for comparison in the electro-optics society by other scholars [14], [15]. The Expectation-Maximization (EM) based MFBD algorithm and the GS phase retrieval algorithm are integrated such that the Expectation Step (E-Step) calculates the conditional expectation of the log-likelihood of the object function and the PSFs. The conditional expectation is maximized in the Maximization Step (M-Step) to iteratively update the object function and the PSFs. The PSFs from the M-Step is fed into the GS algorithm to recover their phases. Afterwards, the object function from the M-Step and the PSFs and their phases from the GS algorithm are fed back into the E-Step. The system diagram in Figure 3.11 shows one iteration of the blind deconvolution algorithm with the phase retrieval algorithm. For additional information, the blind deconvolution and phase retrieval algorithms are described in Section 2.6.4 and Section 2.7 respectively. Even though the blind deconvolution algorithm and the phase retrieval algorithm are used together, their operations are independent from each other. Therefore, a different phase retrieval algorithm can be used instead of the GS phase retrieval algorithm.

A total of 1,000 iterations are completed in the blind deconvolution algorithm. For each blind deconvolution iteration, a total of 100 phase retrieval iterations are completed. The iteration numbers are arbitrary but large enough for the update functions to converge.

The diffraction limited PSF with no phase aberration, as described in Section 3.3.2, is used as the initial PSF estimates, $\hat{h}$. The initial estimated object function, $\hat{o}$, is formed from the average of all the data functions. Prior to averaging, all the data functions are aligned using the cross-correlation technique.

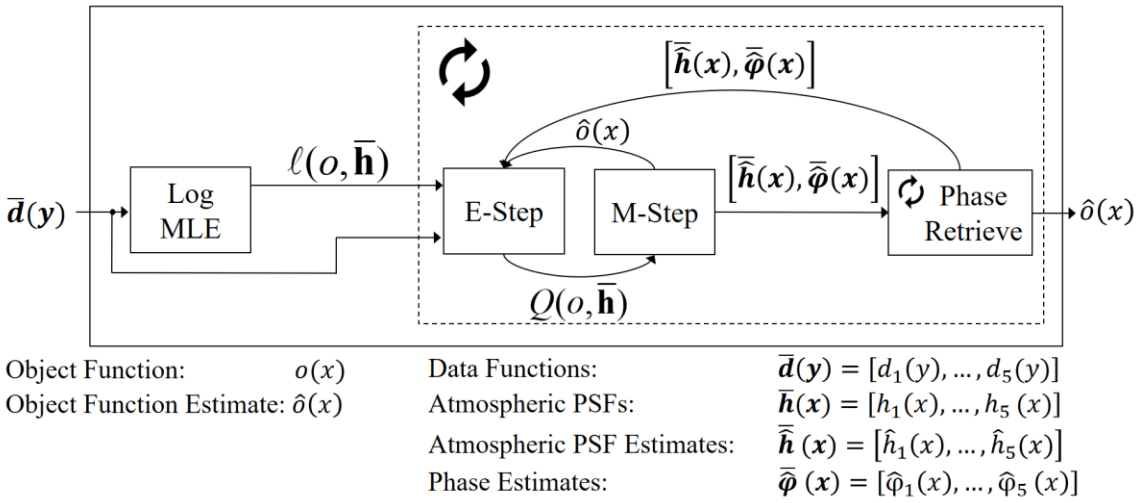


**Figure 3.11. The Blind Deconvolution Algorithm with Phase Retrieval.**

## 3.5. Performance Metrics

Each scenario, which consists of 500 datasets, produces 500 estimated object functions after being processed through the blind deconvolution algorithm with phase retrieval algorithm. In order to quantify performance, two performance metrics are used: (1) dim object detection and (2) dim object average.

### 3.5.1. Dim Object Detection

From the 500 estimated object functions, $\{[\hat{o}(x)]_n, n=1,...,N\}$ where $N=500$, the intensities of the dim object from the $n^{th}$ dataset, $[\hat{i}_{det}]_n$, and that of the false alarm location from the same dataset, $[\hat{i}_{det}]_n$, are calculated using a correlation method with a 2D Gaussian mask function with a 0.5 pixel standard deviation in both X- and Y-axes. The Gaussian mask function is centered at the locations of the dim object and the false alarm pixel to calculate the intensities as shown in the equation below. The standard deviation value of 0.5 pixels provides good detection rates by averaging out the noise near the dim object and the false alarm pixel.

$$\left[\hat{i}_{det}\right]_n = \sum_{\forall x \in X}\left(\left[\hat{o}(x)\right]_n \times \frac{1}{2\pi\sigma^2}\exp\left(-\frac{\left(x-x_{det}\right)^T\left(x-x_{det}\right)}{2\sigma^2}\right)\right), \tag{3.38}$$

$$\left[\hat{i}_{fa}\right]_n = \sum_{\forall x \in X}\left(\left[\hat{o}(x)\right]_n \times \frac{1}{2\pi\sigma^2}\exp\left(-\frac{\left(x-x_{fa}\right)^T\left(x-x_{fa}\right)}{2\sigma^2}\right)\right), \tag{3.39}$$

where $x_{det}$ and $x_{fa}$ are the locations for the dim star and the false alarm pixel, $\sigma$ is the standard deviation of the Gaussian mask function, and × represents a pixel-by-pixel multiplication between two functions.

As shown in Figure 3.12, the false alarm location is selected to be on the opposite side of the dim object at the same distance from the bright object. This location is chosen because it does not contain any object intensity. There are two other pixels that can be used as the alternate false alarm location. These two are not optimal because they are located at $\sqrt{2}$ pixels away from the dim object even though they are located two pixels away from the bright object.

**Table 3.5. Dim Object and False Alarm Locations of an Estimated Object Function.**

| Description | Values/Coordinates |
|---|---|
| Detector Size | 64×64 |
| Bright Object Location | (32, 32) |
| Dim/Secondary Object Location ($x_{det}$) | (32, 34) |
| False Alarm Pixel Location ($x_{fa}$) | (32, 30) |

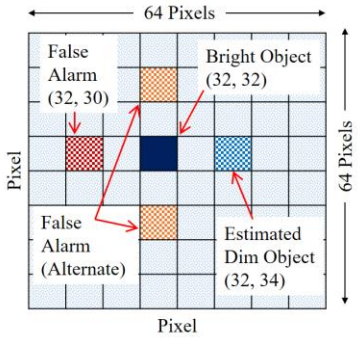


**Figure 3.12. Estimated Object Function.**

Using the correlation method with the Gaussian masks, a set of 500 intensities from the dim object and that from the false alarm pixel are then fitted to the gamma function. See Appendix A for the technique to estimate the gamma distribution parameters. Figure 3.13 through Figure 3.20 show the probability distribution functions and the cumulative distribution functions for the intensities of the dim object and the false alarm pixels for each scenario, along with their corresponding gamma distribution fits. Except for the intensities for the dim objects in Scenario 1 and Scenario 2, the intensities follow the shape of the gamma distribution. Even though other functions, such as the Weibull function, might be better fits for these two first scenarios, the separation between the dim object

distribution and the false alarm pixel distribution is well separated, obviating the need to fit to other types of functions.

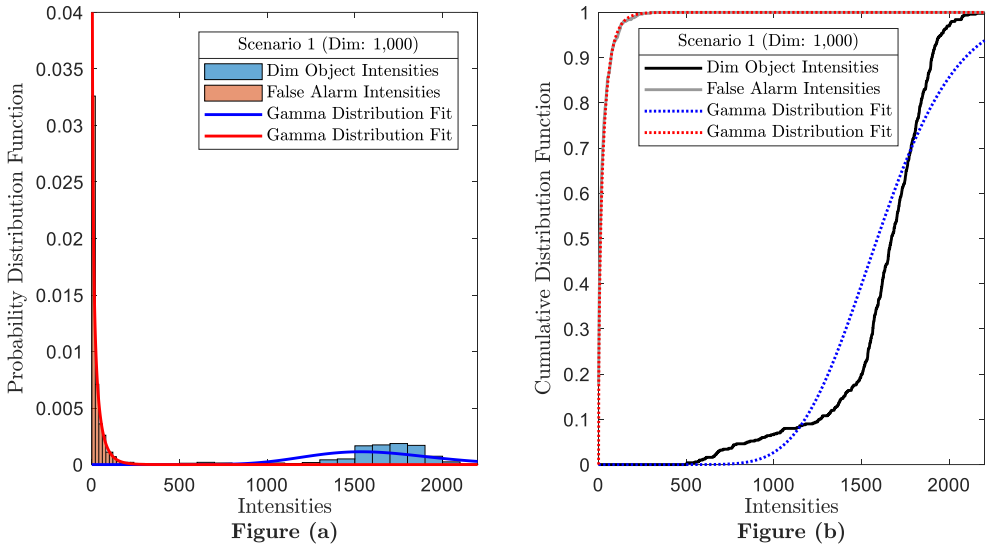


**Figure 3.13. The distributions of the intensities of the dim object and the false alarm pixel for Scenario 1 using the MFBD approach.**

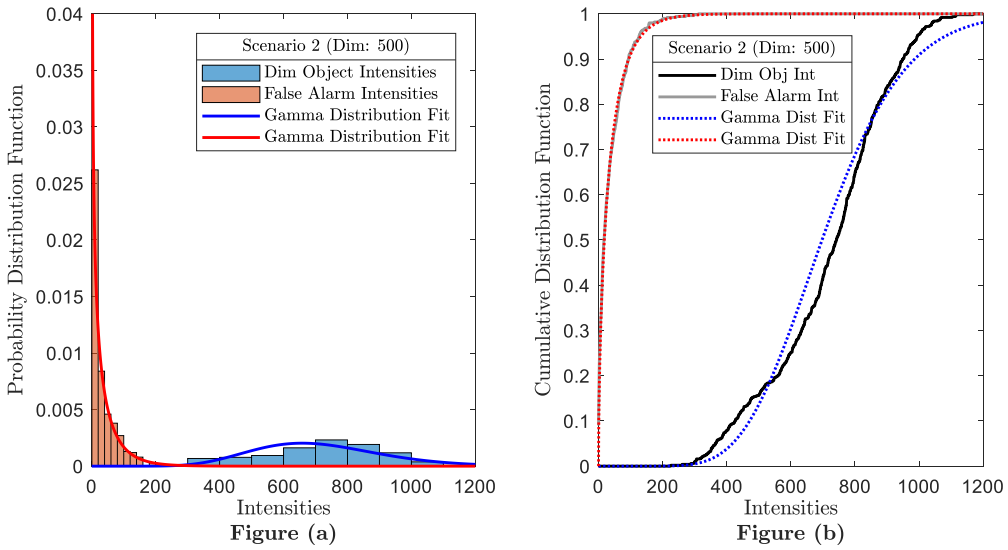


**Figure 3.14. The distributions of the intensities of the dim object and the false alarm pixel for Scenario 2 using the MFBD approach.**

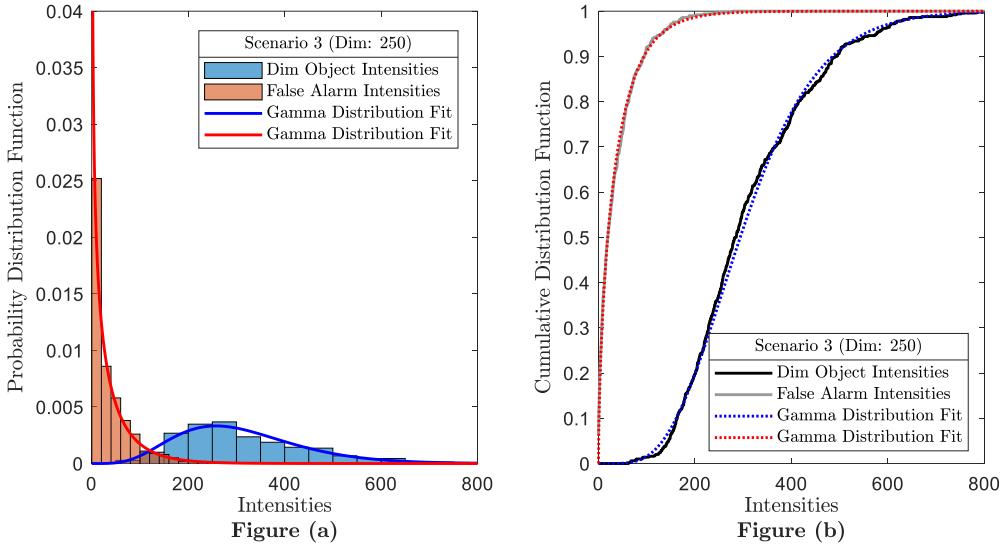


**Figure 3.15. The distributions of the intensities of the dim object and the false alarm pixel for Scenario 3 using the MFBD approach.**

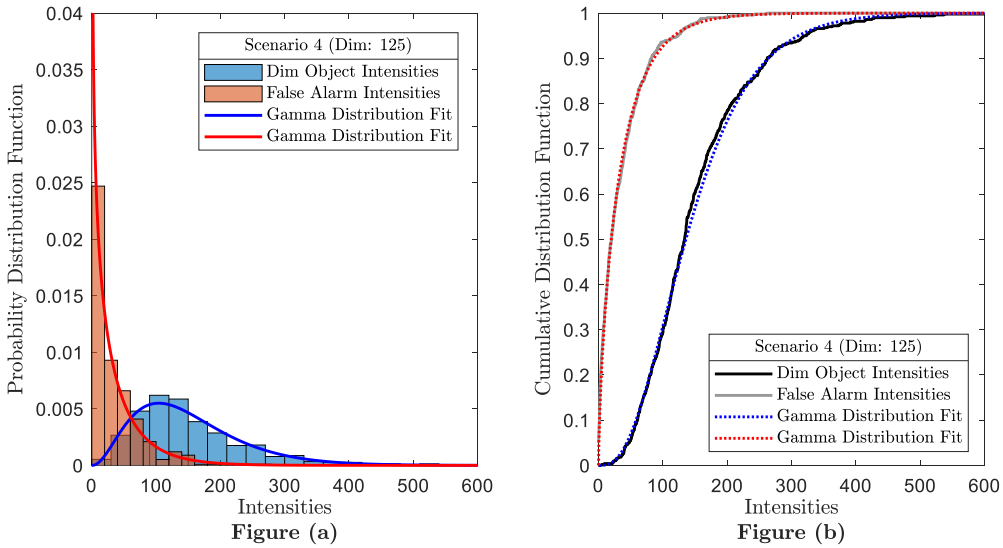


**Figure 3.16. The distributions of the intensities of the dim object and the false alarm pixel for Scenario 4 using the MFBD approach.**

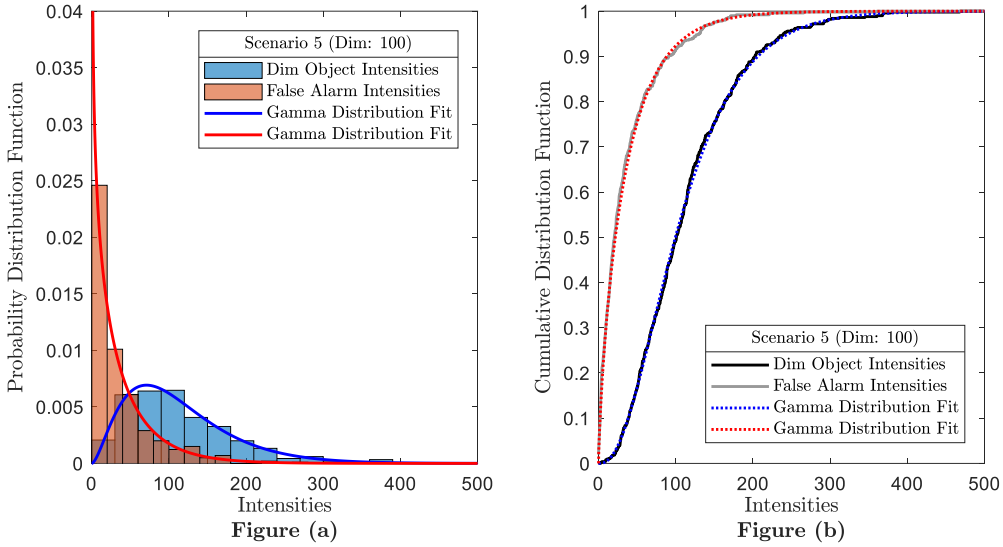


**Figure 3.17. The distributions of the intensities of the dim object and the false alarm pixel for Scenario 5 using the MFBD approach.**

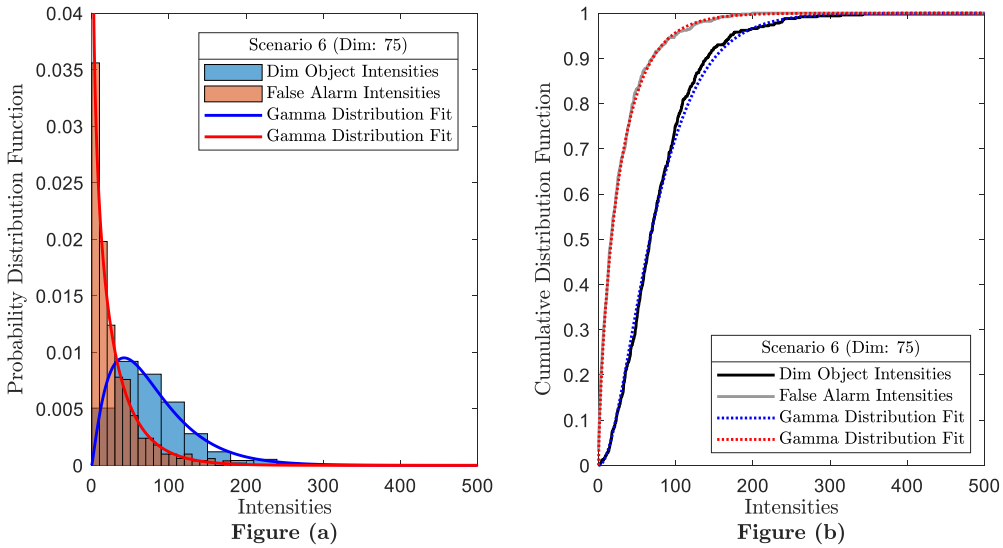


**Figure 3.18. The distributions of the intensities of the dim object and the false alarm pixel for Scenario 6 using the MFBD approach.**

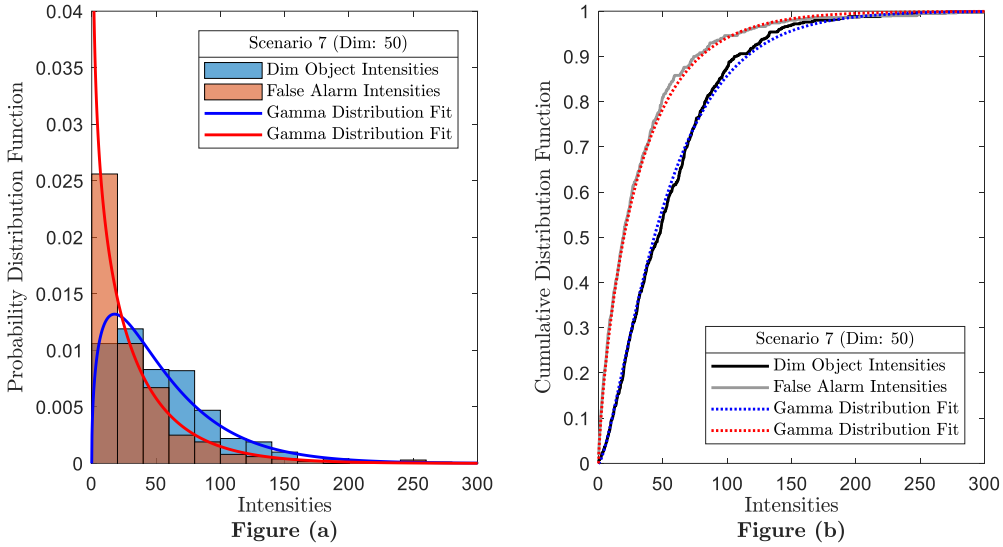


**Figure 3.19. The distributions of the intensities of the dim object and the false alarm pixel for Scenario 7 using the MFBD approach.**

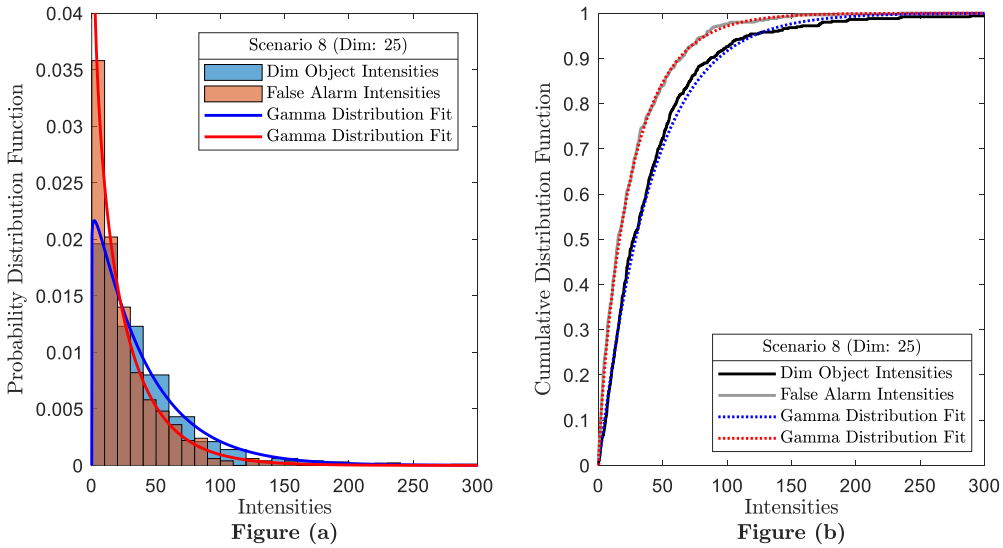


**Figure 3.20. The distributions of the intensities of the dim object and the false alarm pixel for Scenario 8 using the MFBD approach.**

After the gamma fits are obtained, the survival function (SF) of the false alarm intensities is compared with that of the dim object intensities for each scenario to generate the receiver operating characteristic (ROC) functions. The survival function is defined as the complement of the cumulative distribution function (CDF) as shown in the equation below [87], [88].

$$\mathrm{SF} = 1 - \mathrm{CDF}. \tag{3.40}$$

The ROC compares how well the distribution of the dim object is separated from that of the false alarm. This metric is used because it plays a key role in the signal detection communities [89], medical diagnostics [90], and machine learning [91].

The ROC curve for all eight scenarios are shown in Figure 3.21. As the intensity of the dim object is reduced, the ability of the algorithm to detect the dim object also decreases. In addition, Figure 3.21 also shows the performance of the blind deconvolution algorithm using 2,000 iterations. A total of 100 phase retrieval iterations is used for each blind deconvolution algorithm. From the results, it can be seen that performing additional blind deconvolution iterations does not further improve the performance.

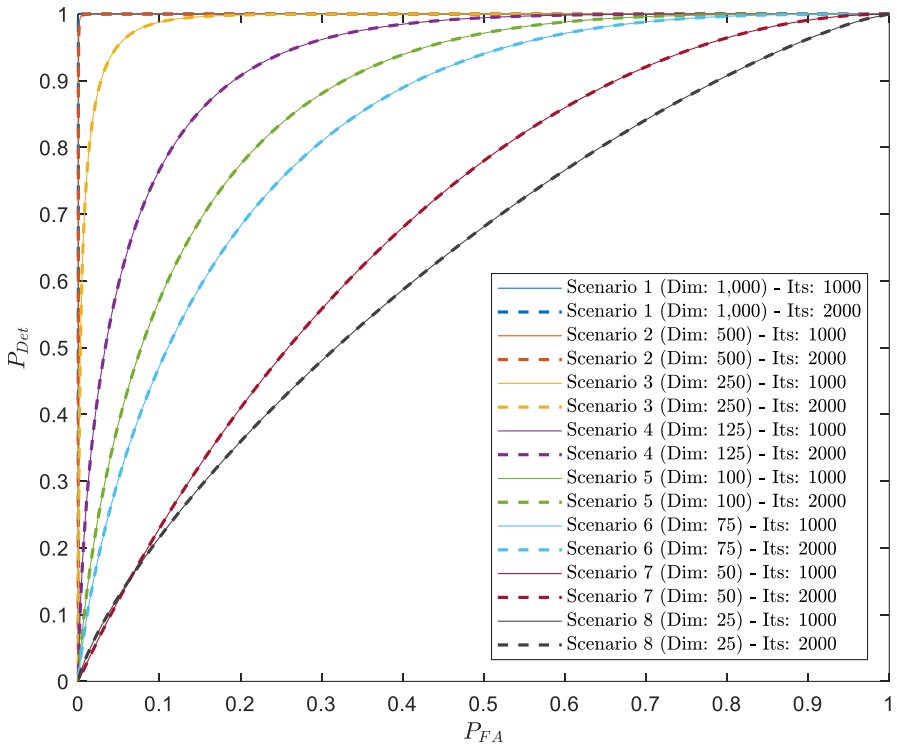


**Figure 3.21. The dim object detection performance of the MFBD algorithm for all 8 scenarios with 1,000 and 2,000 iterations.**

### 3.5.2. Dim Object Average

The second performance metric is the average function, $\hat{o}_{avg}(x)$, which is defined as the mean of all estimated object functions for each scenario. This metric complements the first one because this one provides the average information, while the first one provides the information about how well the two distributions are separated.

$$\hat{o}_{avg}(x) = \frac{1}{500}\sum_{n=1}^{500}\left[\hat{o}(x)\right]_n . \tag{3.41}$$

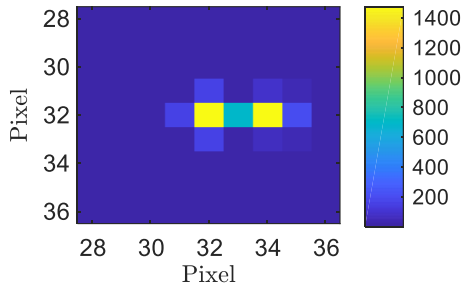


Fig. (a): Scenario 1 (Dim: 1,000)

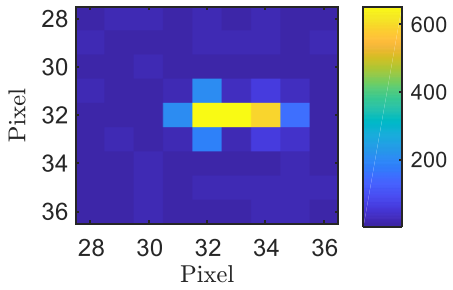


Fig. (b): Scenario 2 (Dim: 500)

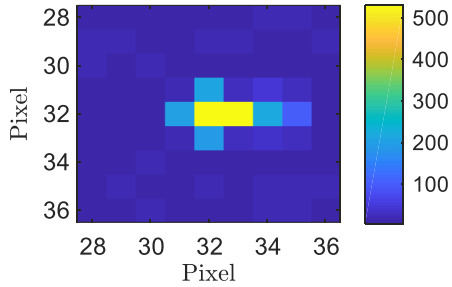


Fig. (c): Scenario 3 (Dim: 250)

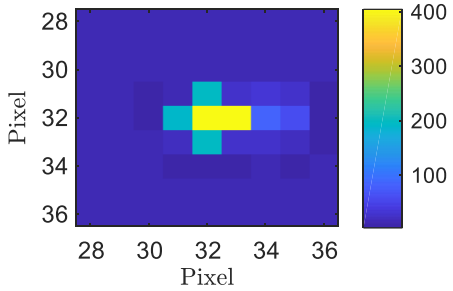


Fig. (d): Scenario 4 (Dim: 125)

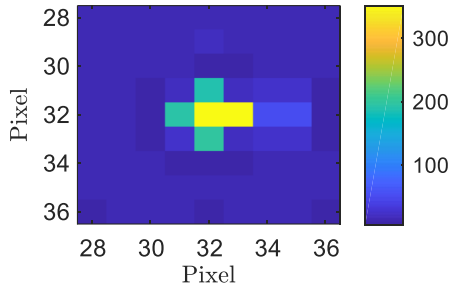


Fig. (e): Scenario 5 (Dim: 100)

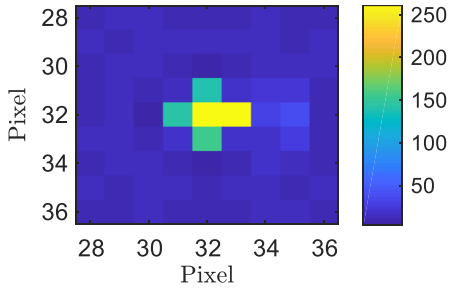


Fig. (f): Scenario 6 (Dim: 75)

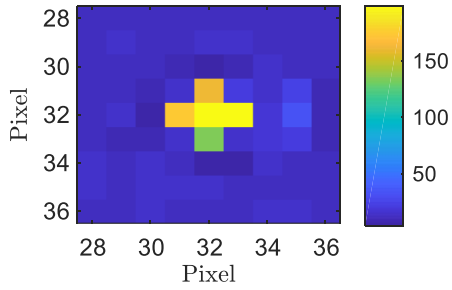


Fig. (g): Scenario 7 (Dim: 50)

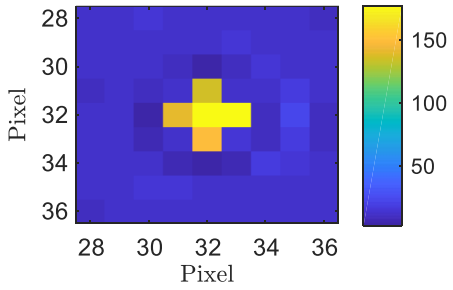


Fig. (h): Scenario 8 (Dim: 25)

**Figure 3.22. The dim object average performance of the MFBD algorithm for all eight scenarios.**

**Important Note**: The intensities of the bright object in Figure 3.22 are matched to the second brightest pixel to reduce the contrast ratio for all scenarios.

The average functions for all eight scenarios are shown in Figure 3.22. As the intensity of the dim object is reduced, the ability to resolve the dim object from the bright object is reduced. The system level diagram for obtaining the two metrics is shown in Figure 3.23.

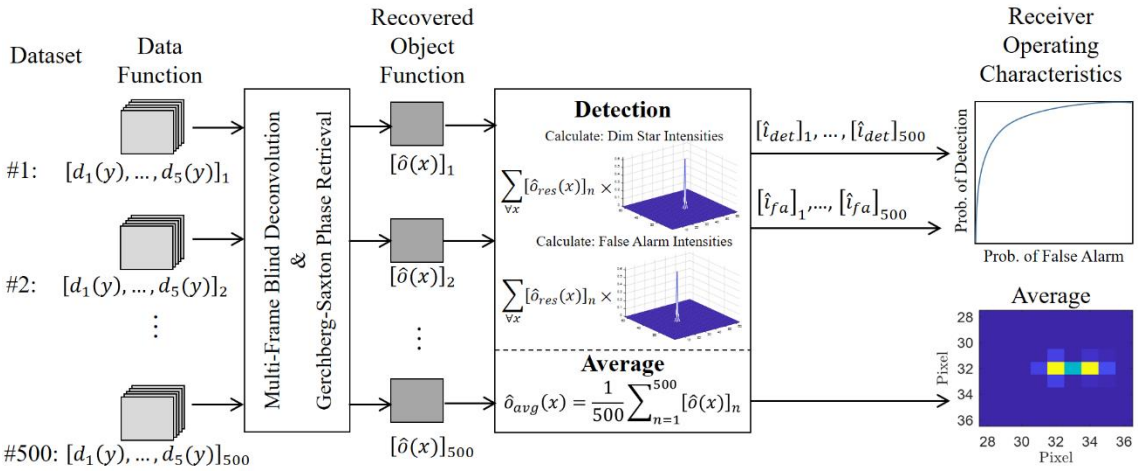


**Figure 3.23. System Level Diagram of the MFBD Algorithm as the Baseline.**

## 3.6. Summary

This chapter explained the methodology that is used throughout this research. It provided the detail implementations of the tools and techniques described in the previous chapter. First, it explained the eight main assumptions made throughout this research. Afterwards, the implementation for the detector model, the atmospheric PSF, the object function, the image function, and the data function were explained. Then, it explained how the Gerchberg-Saxton phase retrieval algorithm is integrated into the Multi-Frame Blind Deconvolution algorithm. Next, this chapter explained the data functions for different

scenarios for the object function, where the bright object and background intensities are kept the same but the intensity of the dim object is gradually decreased. The turbulence strength was kept the same for generating random atmospheric phase screens. Lastly, this chapter showed the performance of the Multi-Frame Blind Deconvolution algorithm with the Gerchberg-Saxton phase retrieval with two metrics, which are (1) dim object detection and (2) dim object average. The results for the eight scenarios were also shown, using these two metrics. The results also showed that the performance of the Multi-Frame Blind Deconvolution saturated after some iteration. To gain additional performance, the next chapter introduces the Neighborhood System Blind Deconvolution.

# 4. Neighborhood System Blind Deconvolution

## 4.1. Chapter Overview

This chapter introduces a new Neighborhood System Blind Deconvolution (NSBD) algorithm to further improve the dim object detection performance over the Multi-Frame Blind Deconvolution (MFBD) algorithm. The performance results of the new algorithm are compared with those of the MFBD algorithm using two metrics which are (1) the ability to detect the dim object using the receiver operating characteristic (ROC) functions and (2) the average intensity of the dim object as described in Section 3.5. The datasets used in this chapter are the same as those explained in the previous chapter.

## 4.2. Algorithm Development

In Section 3.5, it has been shown that the performance of the MFBD algorithm saturates after a certain point, meaning additional iterations of the blind deconvolution algorithm no longer further improves the performance. Therefore, a new algorithm is derived, which is built on the method proposed by Cain [77]. In this chapter, it is assumed that the data is sampled at the Nyquist rate or higher.

The NSBD algorithm separates the object function into three functions, which are a function of unknown neighborhood system around the bright object, $o_s(x)$, the amplitude of the primary bright object that is known to exist, $o_p(x)$, and the background light and dark current measured during the acquisition process, *B*. The assumption is made the intensities of these three functions are statistically independent.

$$o_p(x) = \gamma\delta\left(x - x_p\right), \tag{4.1}$$

$$o_s(x) = \sum_{m=1}^{M} \mu_m \delta\left(x - x_m\right), \tag{4.2}$$

$$o_b(x) = B, \tag{4.3}$$

where $x_p$ and $\gamma$ are the location and the magnitude of the bright object, $\mu_m$ is the number of photoelectrons emitted from the $m^{\text{th}}$ star in the system, $M$ is the total number of pixels in the system, and $\delta$ is the Dirac Delta function. In this chapter, $x_p$ and $\gamma$ are obtained from either the results of the MFBD algorithm or the unprocessed data function. However, if there exists some prior information about them, the known information should be used.

Since the image function is a convolution between the object function and the PSF, the image functions for the primary bright object, $i_k^p$, the neighborhood system, $i_k^s$, and the background, $i_k^b$, for the $k^{\text{th}}$ observation are described by the following equations.

$$i_k^p(y) = \gamma h_k\left(y - x_p\right). \tag{4.4}$$

$$i_k^s(y) = \sum_{\forall x} h_k(y - x) o_s(x). \tag{4.5}$$

$$i_k^b(y) = \sum_{\forall x} h_k(y - x) o_b(x). \tag{4.6}$$

The data functions from the CCD can be expressed as a sum of all the three data functions formed from the three image functions mentioned above. In addition, as described in Section 3.3.6, the CCD introduces signal dependent Poisson noise to the image functions as shown in Equation (4.7) through Equation (4.10).

$$d_k(y) = d_k^p(y) + \sum_{\forall x} d_k^s(y,x) + d_k^b(y), \tag{4.7}$$

$$\mathrm{E}\left[d_k^p(y)\right] = \gamma h_k\left(y - x_p\right), \tag{4.8}$$

$$\mathrm{E}\left[d_k^s(y \mid x)\right] = o_s(x) h_k(y-x), \tag{4.9}$$

$$\mathrm{E}\left[d_k^b(y)\right] = B \tag{4.10}$$

Since the three functions are statistically independent, the likelihood function will be at the maximum when the probability mass functions of the primary object function, the neighborhood system function, and the background function are at their maximums. Therefore, the likelihood function of the three sets of the data functions are expressed as shown in Equation (4.11).

$$L(o,h_k) = \prod_{\forall k}\prod_{\forall y}\prod_{\forall x}\left[\mathrm{P}\left(d_k^p(y)\right)\mathrm{P}\left(d_k^s(y \mid x)\right)\mathrm{P}\left(d_k^b(y)\right)\right]. \tag{4.11}$$

Using the Expectation-Maximization (EM) algorithm on Equation (4.11), the update equations for the new algorithm are obtained as shown in Equation (4.12) through Equation (4.17). The derivations for the EM algorithm is similar to those described in Section 2.6.3. The update equation for the PSF remains the same as the MFBD algorithm. For the convenience for the readers, the update equation for the PSF is shown again in Equation (4.16).

$$\gamma^{\text{new}} = \gamma^{\text{old}} \frac{1}{K} \sum_{\forall k}\sum_{\forall y} \frac{h_k^{\text{old}}(y - x_p) d_k(y)}{i_k^{\text{old}}(y)}, \tag{4.12}$$

$$o_s^{\text{new}}(x) = o_s^{\text{old}}(x) \frac{1}{K} \sum_{\forall k}\sum_{\forall y} \frac{h_k^{\text{old}}(y - x) d_k(y)}{i_k^{\text{old}}(y)}, \tag{4.13}$$

$$o_b^{\text{new}}(x) = \frac{o_b^{\text{old}}(x)}{K} \sum_{\forall k} \sum_{\forall y} \frac{h_k^{\text{old}}(y-x)\, d_k(y)}{i_k^{\text{old}}(y)}, \tag{4.14}$$

$$o^{\text{new}}(x) = o_p^{\text{new}}(x) + o_s^{\text{new}}(x) + o_b^{\text{new}}(x), \tag{4.15}$$

$$h_k^{\text{new}}(x) = h_k^{\text{old}}(x) \sum_{\forall y} \frac{o^{\text{old}}(y-x) d_k(y)}{i_k^{\text{old}}(y)}, \tag{4.16}$$

$$i_k^{\text{new}}(y) = \sum_{x \in X} h_k^{\text{new}}(y-x)\, o^{\text{new}}(x). \tag{4.17}$$

Even though the background function is assumed to be constant across all pixels as shown in Equation (4.3), it is not constrained to be constant across all pixels at each iteration.

### 4.3. System Implementation and Initialization

This section discusses about two ways to implement the system. They are: (1) the coupled approach where the outputs of the MFBD algorithm is fed into the NSBD algorithm and (2) the decoupled approach where the NSBD algorithm is processed independently of the MFBD algorithm.

#### 4.3.1. Coupled Approach

In the first approach, the MFBD algorithm and the NSBD algorithm are coupled, where the NSBD algorithm uses the object function estimate and the PSF estimates from the MFBD algorithm. Figure 4.1 shows the overall system, along with the methods for modeling and simulation. The same 500 datasets for each scenario used in the MFBD algorithm are also used in the NSBD algorithm. This approach was used for the results that were presented at the 2019 AMOS Conference [92].

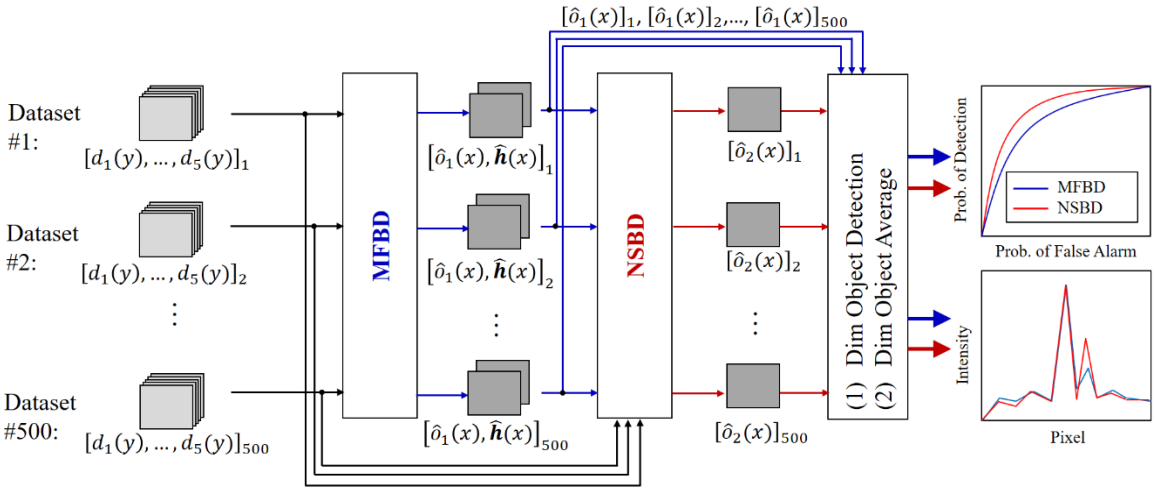


**Figure 4.1. The Overall System Implementation for the Coupled Approach.**

For the MFBD algorithm, the object function is initialized with a 64×64 pixel plane with a constant intensity of one. For the NSBD algorithm, the initial object functions are also separated into three functions. For the primary object function, the estimates on the brightness and location of the primary object are obtained from the MFBD algorithm. For the neighborhood system function, a circle with radius of 4 pixels outside the bright object and is centered at the bright object. The initial value of 10 is used for all scenarios. The pixels of the primary object function and the neighborhood system function are mutually exclusive so that there are no overlapping pixels shared by both functions. The neighborhood system function is shown in Figure 4.2. For the background function, the whole 64×64 pixel plane is initialized with the value of one. Therefore, the background function overlaps with both the primary object function and the neighborhood system function. These exact initial intensities are not required for the algorithm. As long as the intensity of the primary object function is higher than that of the neighborhood system

function, which in turn is higher than that of the background function, the NSBD will provide consistent results.

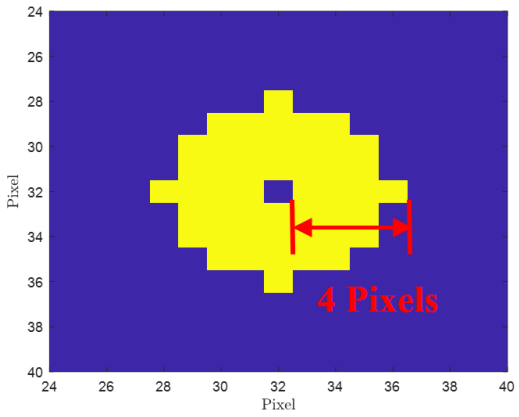


**Figure 4.2. The Neighborhood System Function.**

For the PSFs, the MFBD algorithm is initialized with the diffraction-limited PSFs. However, the PSF estimates from the MFBD algorithm are used as the initial PSFs for the NSBD algorithm.

### 4.3.2. Decoupled Approach

In the second approach, the MFBD algorithm and the NSBD algorithm are decoupled, where the NSBD algorithm does not use the estimates from the MFBD algorithm. The systems diagram for this method is shown in Figure 4.3.

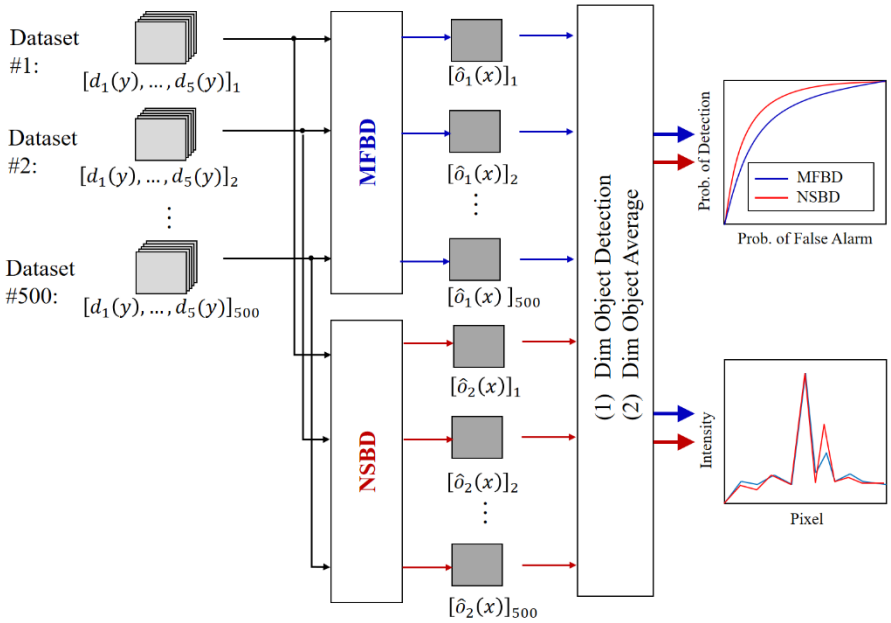


**Figure 4.3. The Overall System Implementation for the Decoupled Approach.**

For the NSBD algorithm, the intensity of the initial primary bright object is calculated from the average of the pixel with the highest intensity from all data functions within the dataset. The initial neighborhood system function and the background functions are the same as those used in the coupled approach.

For the initialization for the PSFs, both the MFBD algorithm and the NSBD algorithm are initialized with the diffraction limited PSF. Therefore, the estimates of the atmospheric phases start at zero across the whole aperture.

## 4.4. Performance Comparison

In this section, the performance of the NSBD algorithm is compared with that of the MFBD algorithm. Similar to the MFBD algorithm, a total of 1,000 EM iterations are used for the NSBD algorithm for both coupled and decoupled approaches. For each EM

iteration, 100 phase retrieval iterations are completed using the Gerchberg-Saxton method for both coupled and decoupled approaches.

### 4.4.1. Dim Object Detection

Similar to the MFBD algorithm, the metric for the ability to detect the dim object is accomplished using the ROC function as described in Section 3.5.1. In this section, the same Gaussian masks from Equation (3.38) and Equation (3.39) are also used in calculating the intensities of the dim object and that of the false alarm pixel respectively. The intensities are also fitted with Gamma distributions. See Appendix B for the Gamma fitted distributions for the coupled and decoupled approaches.

After the fitted gamma distribution functions for the intensities of the dim object and the false alarm pixel are obtained, the survival function, which is the complement of the cumulative distribution function, of the dim object is plotted against that of the false alarm pixel to generate the ROC function for each scenarios.

In addition, the confidence interval bounds are added to both the MFBD algorithm and the NSBD algorithm. The upper bound of the MFBD algorithm is compared with the lower bound of the NSBD algorithm. In other words, the best case of the MFBD algorithm is compared with the worst case of the NSBD algorithm for the given level of significance, $\alpha$. Figure 4.4 provides a graphical explanation for generating the confidence interval for the ROC functions used in this research.

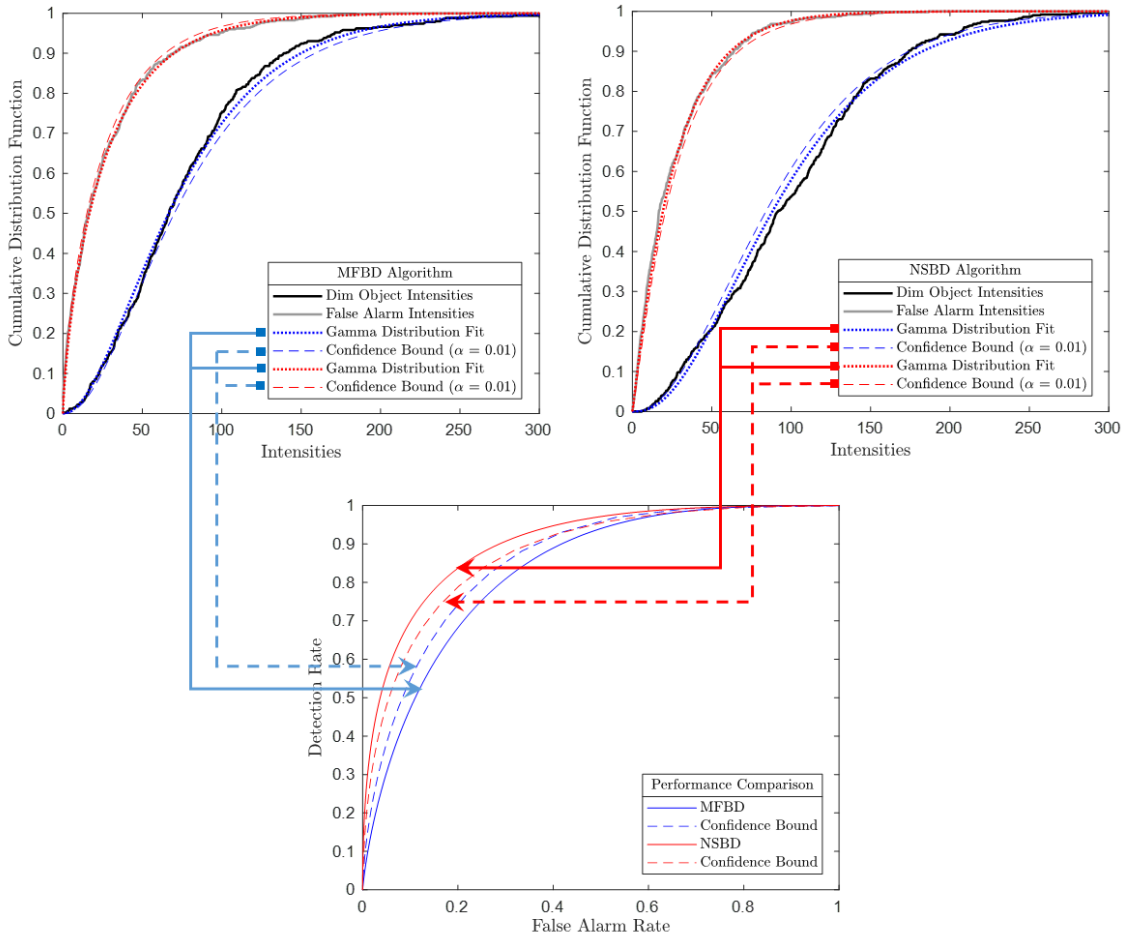


**Figure 4.4. The Process for Generating ROC Functions with Confidence Intervals.**

Figure 4.5 through Figure 4.20 compare the performance comparison between the MFBD algorithm and the NSBD algorithm for all scenarios for both coupled and decoupled approaches using the ROC functions. The level of significance of $\alpha = 0.10$ is used for all scenarios.

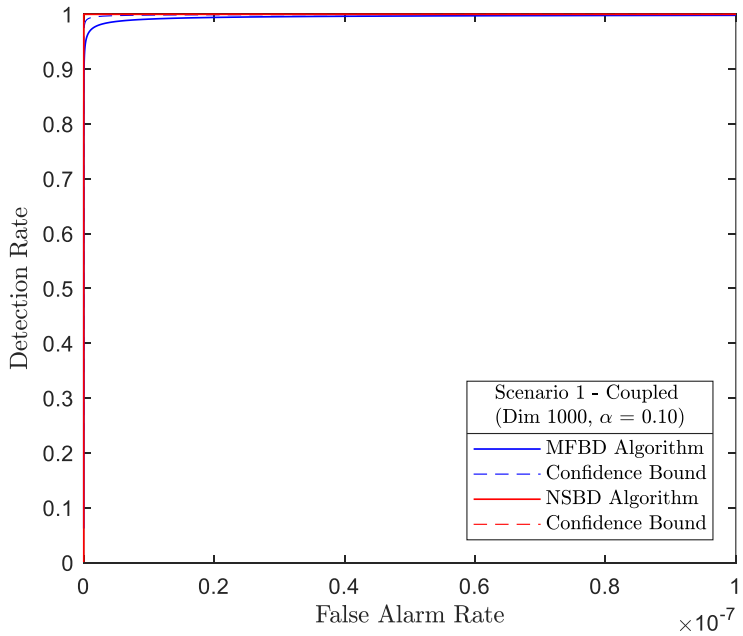


**Figure 4.5. The ROC Comparison for Scenario 1 for the Coupled Approach for Very Low False Alarm Rate.**

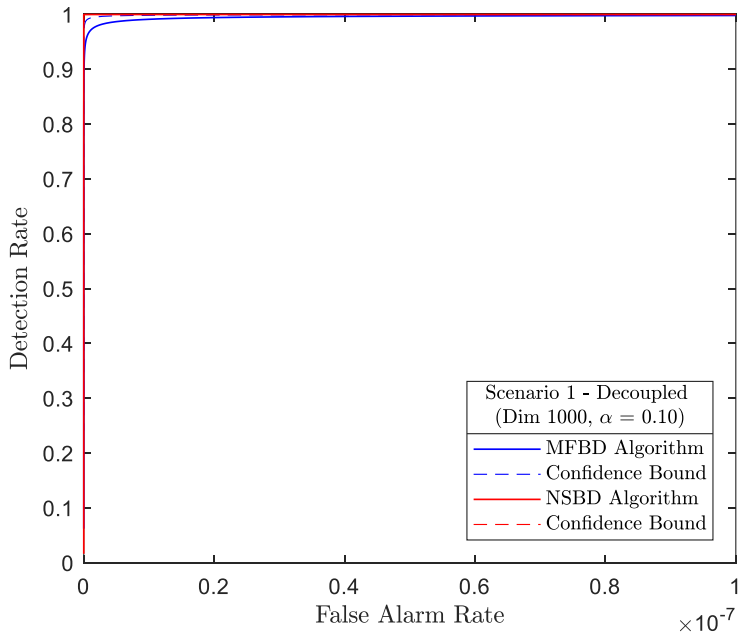


**Figure 4.6. The ROC Comparison for Scenario 1 for the Decoupled Approach for Very Low False Alarm Rate.**

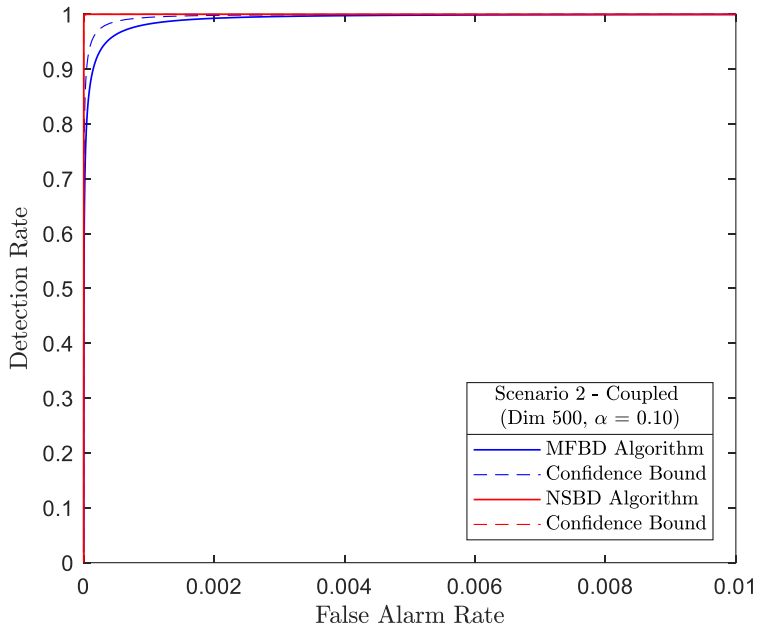


**Figure 4.7. The ROC Comparison for Scenario 2 for the Coupled Approach for Low False Alarm Rate.**

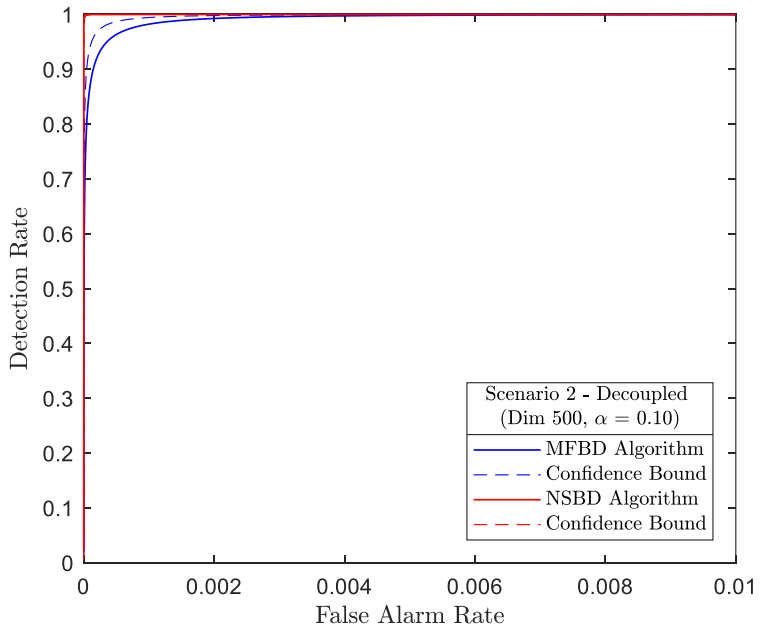


**Figure 4.8. The ROC Comparison for Scenario 2 for the Decoupled Approach for Low False Alarm Rate.**

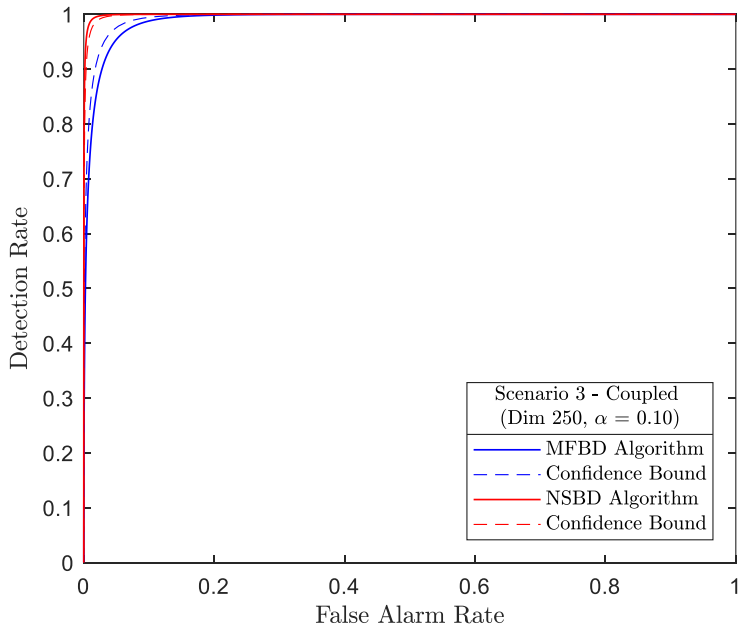


**Figure 4.9. The ROC Comparison for Scenario 3 for the Coupled Approach.**

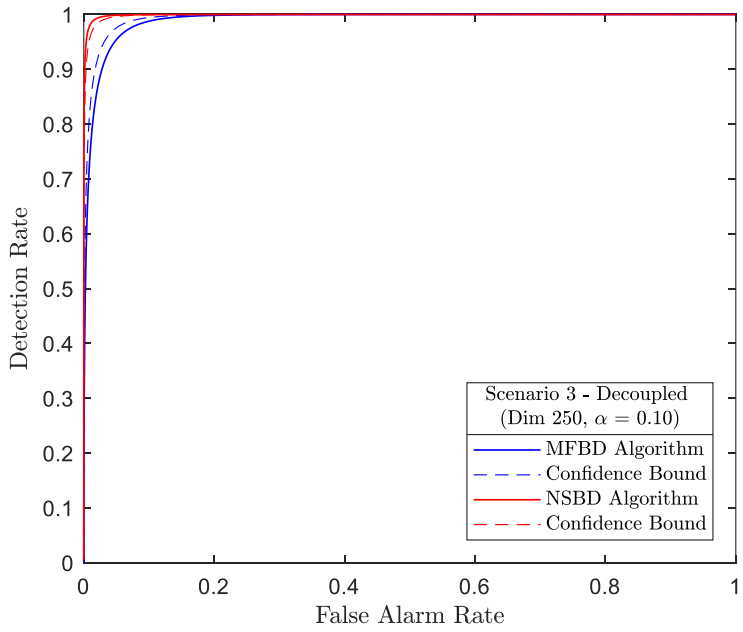


**Figure 4.10. The ROC Comparison for Scenario 3 for the Decoupled Approach.**

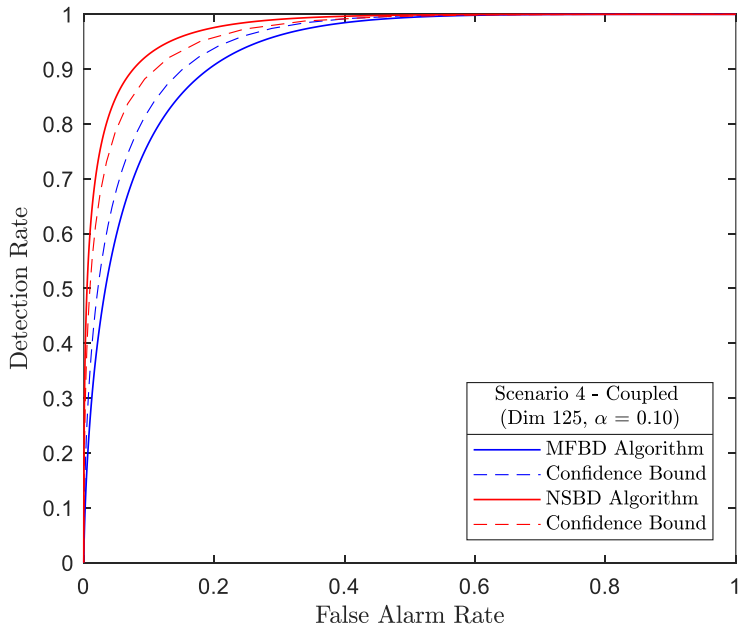


**Figure 4.11. The ROC Comparison for Scenario 4 for the Coupled Approach.**

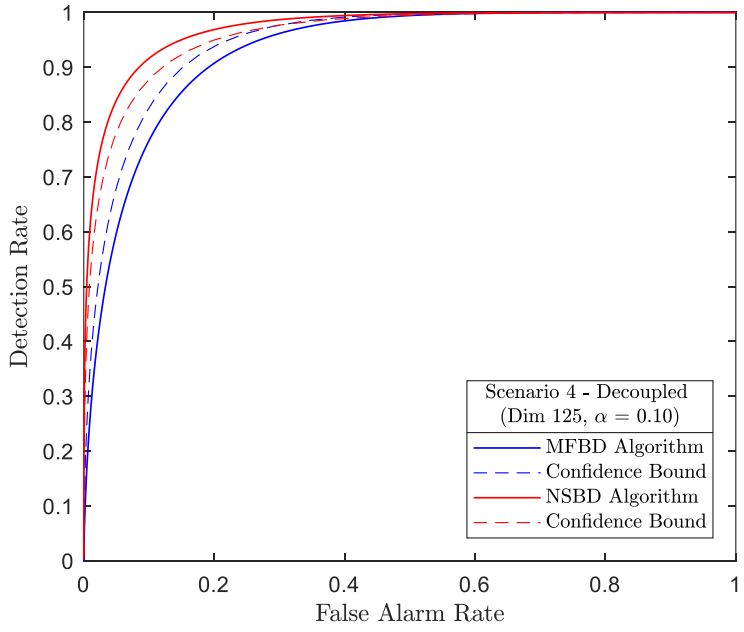


**Figure 4.12. The ROC Comparison for Scenario 4 for the Decoupled Approach.**

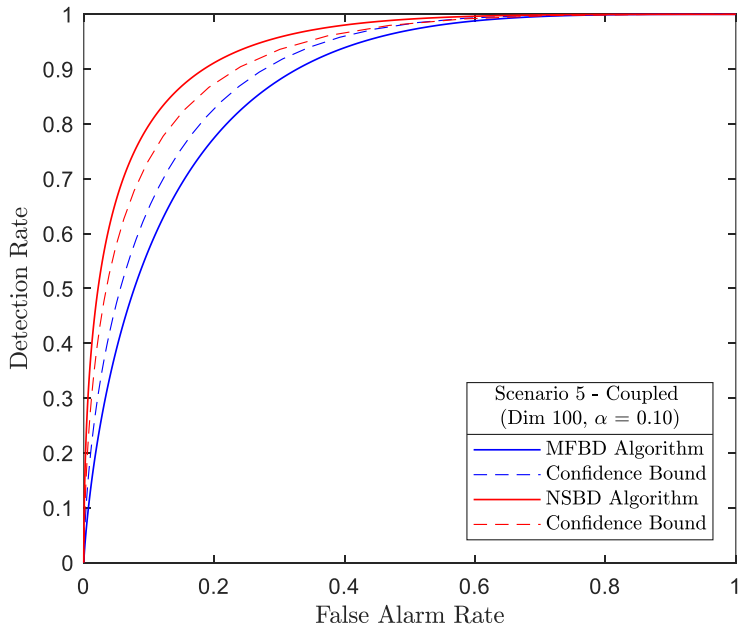


**Figure 4.13. The ROC Comparison for Scenario 5 for the Coupled Approach.**

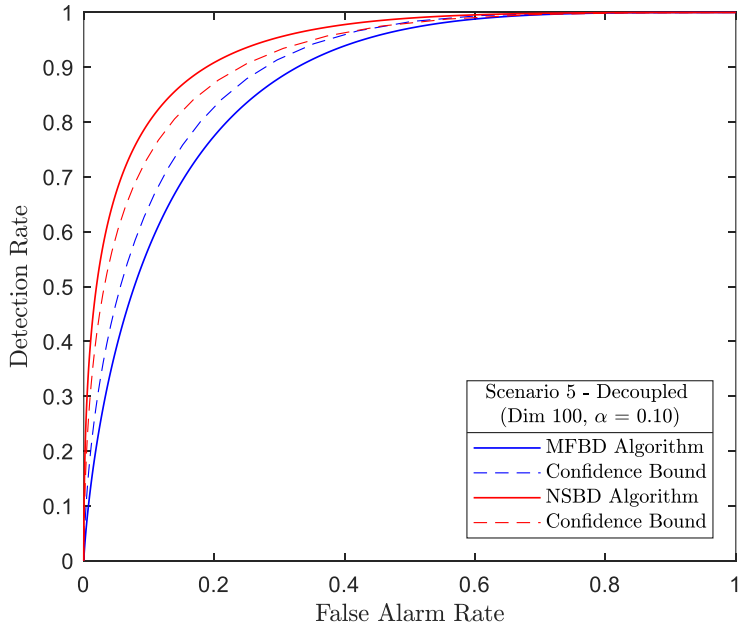


**Figure 4.14. The ROC Comparison for Scenario 5 for the Decoupled Approach.**

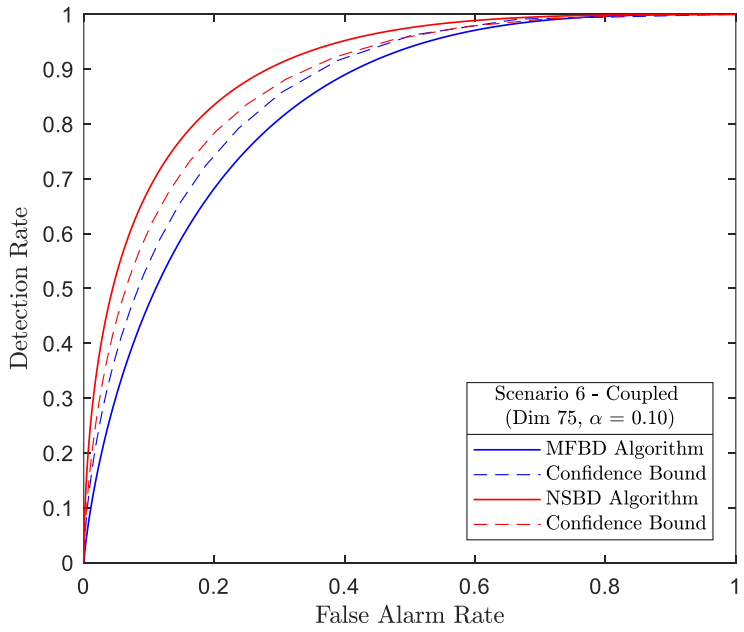


**Figure 4.15. The ROC Comparison for Scenario 6 for the Coupled Approach.**

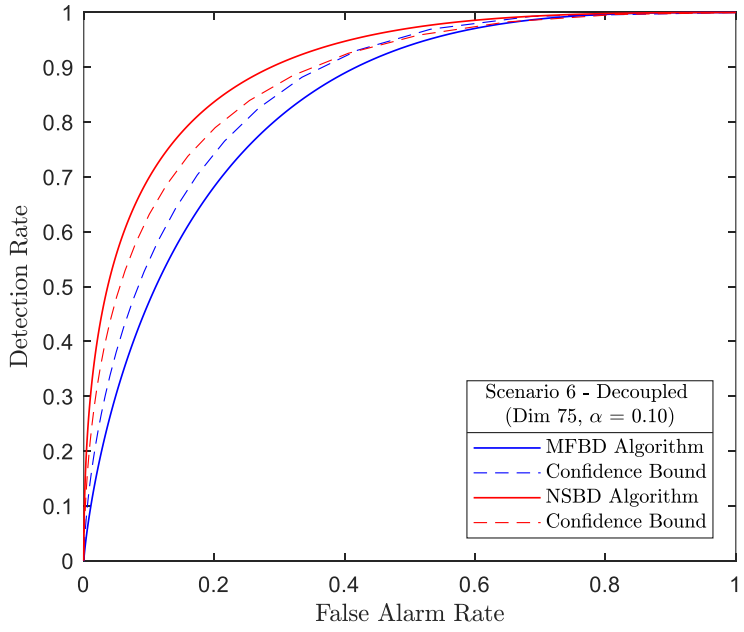


**Figure 4.16. The ROC Comparison for Scenario 6 for the Decoupled Approach.**

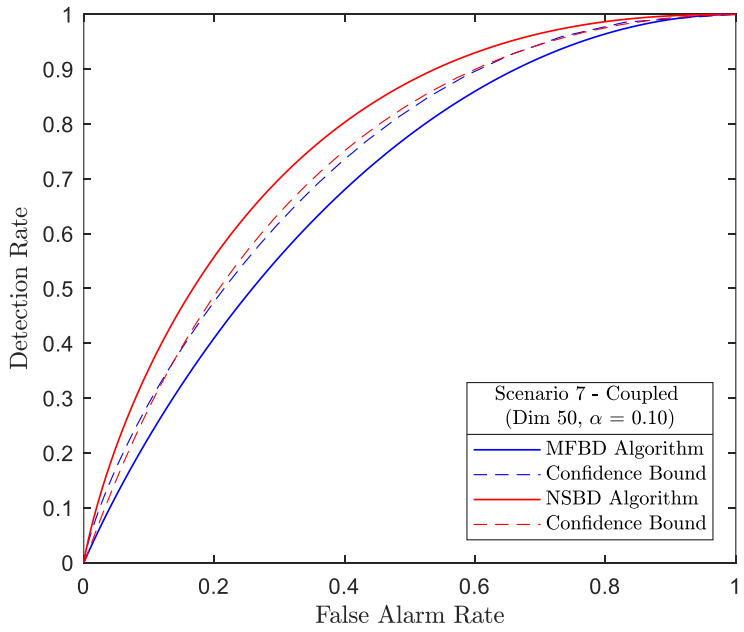


**Figure 4.17. The ROC Comparison for Scenario 7 for the Coupled Approach.**

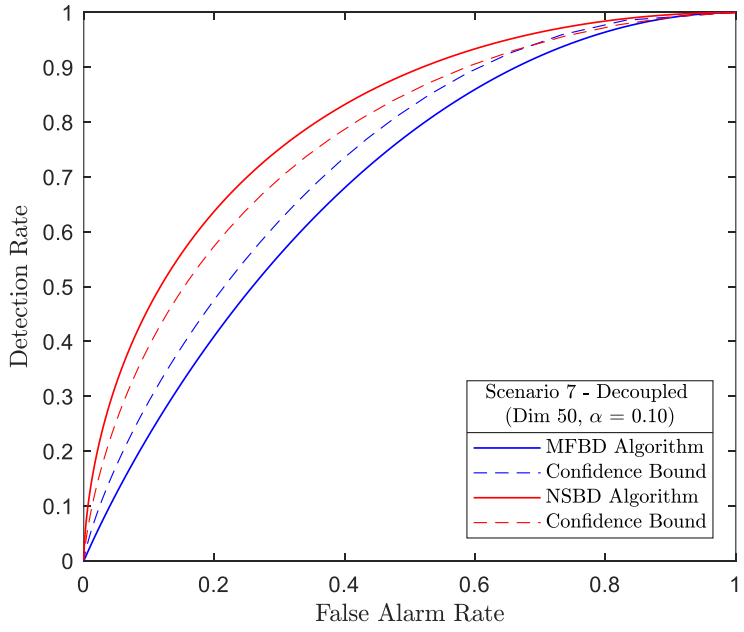


**Figure 4.18. The ROC Comparison for Scenario 7 for the Decoupled Approach.**

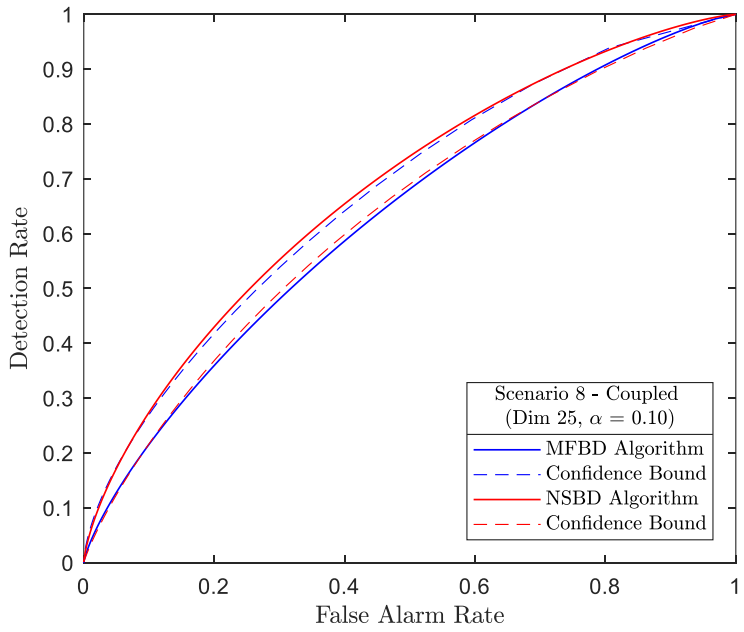


**Figure 4.19. The ROC Comparison for Scenario 8 for the Coupled Approach.**

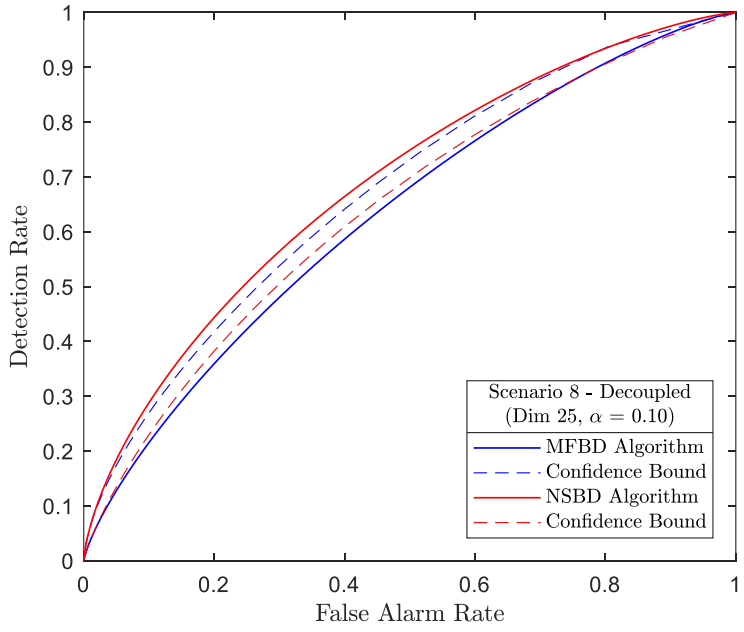


**Figure 4.20. The ROC Comparison for Scenario 8 for the Decoupled Approach.**

From the results, it can be seen that the NSBD algorithm outperforms the MFBD algorithm for all scenarios. In addition, the performance is significantly different for $\alpha = 0.10$ for most of the operating regions for Scenario 1 through Scenario 7 for both coupled and decoupled approaches. The performance is not significantly different for $\alpha = 0.10$ for both coupled and decoupled approaches, even though the NSBD algorithm outperforms the MFBD algorithm. The performance between the coupled and decoupled approaches are consistent. Therefore, the decoupled approach is better in a sense that it requires less computational time and resources without degradation in performance.

### 4.4.2. Dim Object Average

Next, the performance of the NSBD algorithm is compared to the MFBD algorithm using the average function as described in Section 3.5.2. The results are shown in Figure 4.22 through Figure 4.37. From the results, it can be seen that the NSBD algorithm outperforms the MFBD algorithm in this performance metric for all eight scenarios, for both coupled and decoupled approaches. A cross-sectional view of the average function along the axis, $y = 32$, is shown in the figures.

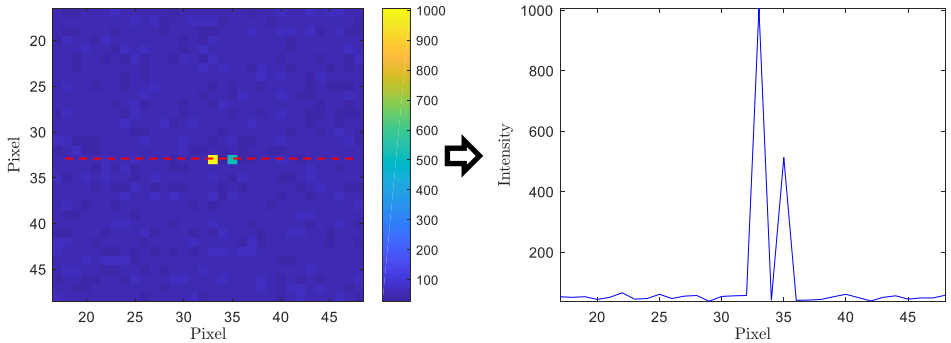


**Figure 4.21. Top Down vs Cross-Sectional View of the Average Function.**

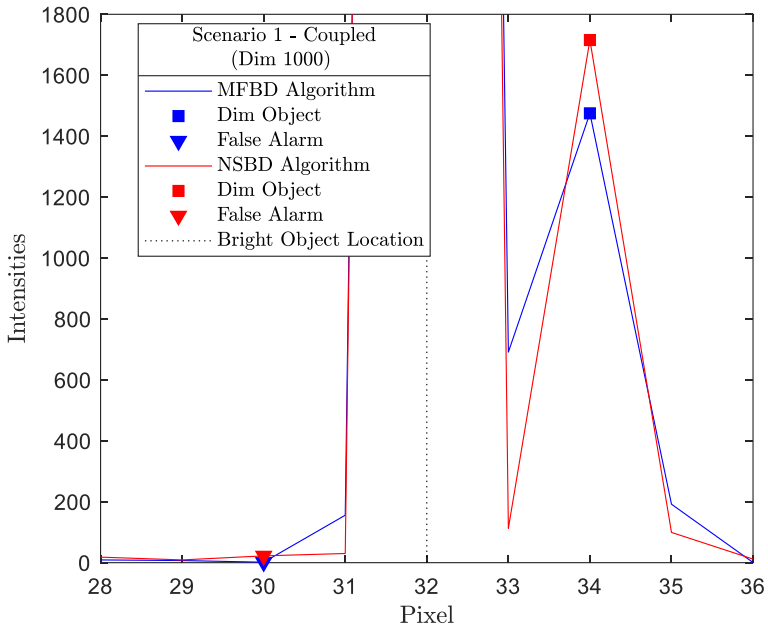


**Figure 4.22. The Average Functions for Scenario 1 for the Coupled Approach.**

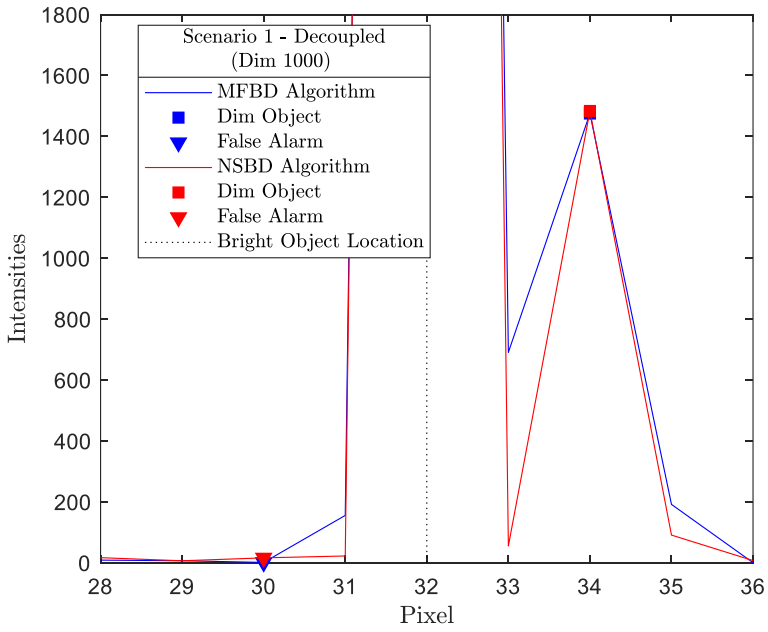


**Figure 4.23. The Average Functions for Scenario 1 for the Decoupled Approach.**

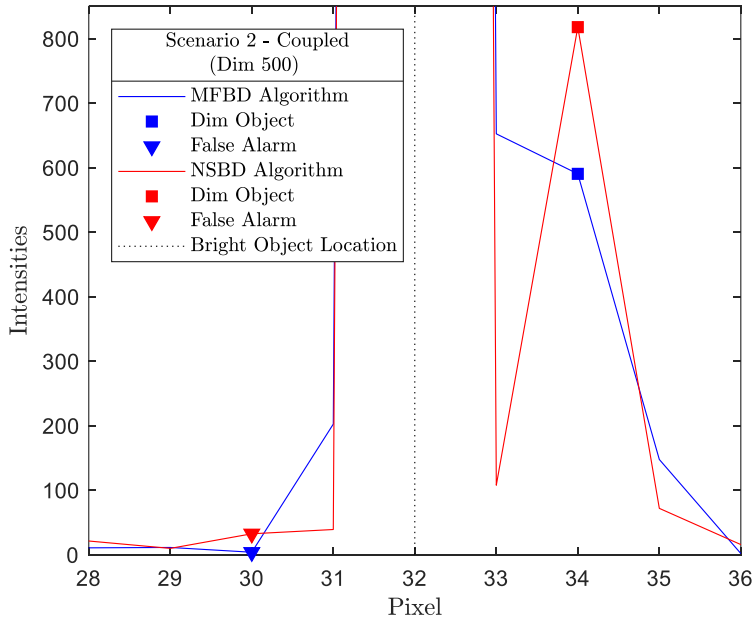


**Figure 4.24. The Average Functions for Scenario 2 for the Coupled Approach.**

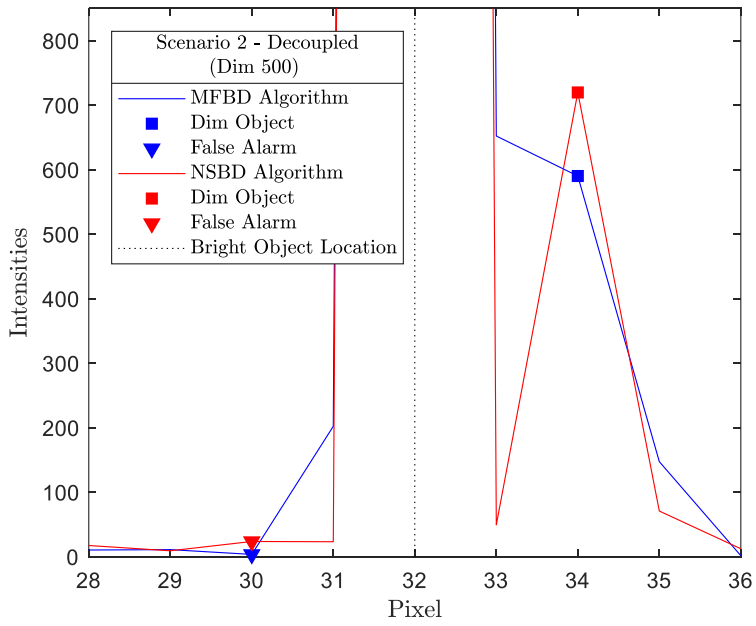


**Figure 4.25. The Average Functions for Scenario 2 for the Decoupled Approach.**

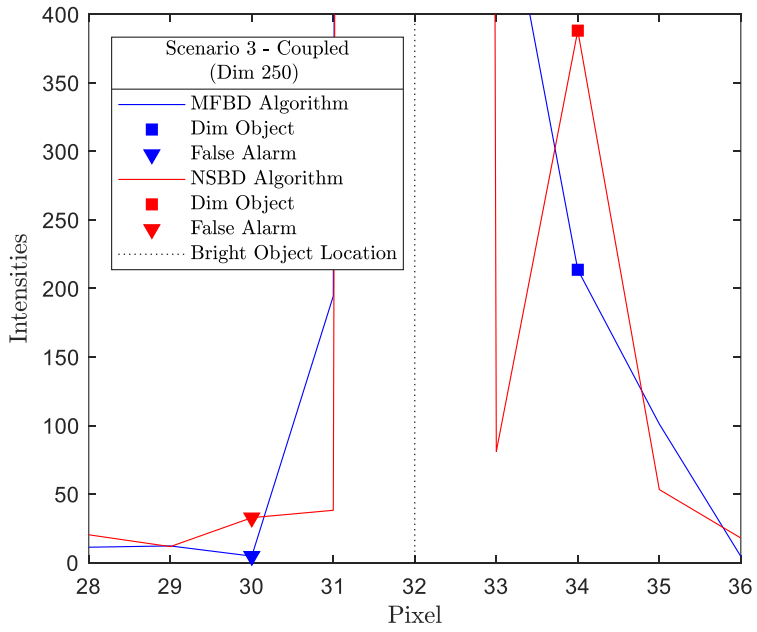


**Figure 4.26. The Average Functions for Scenario 3 for the Coupled Approach.**

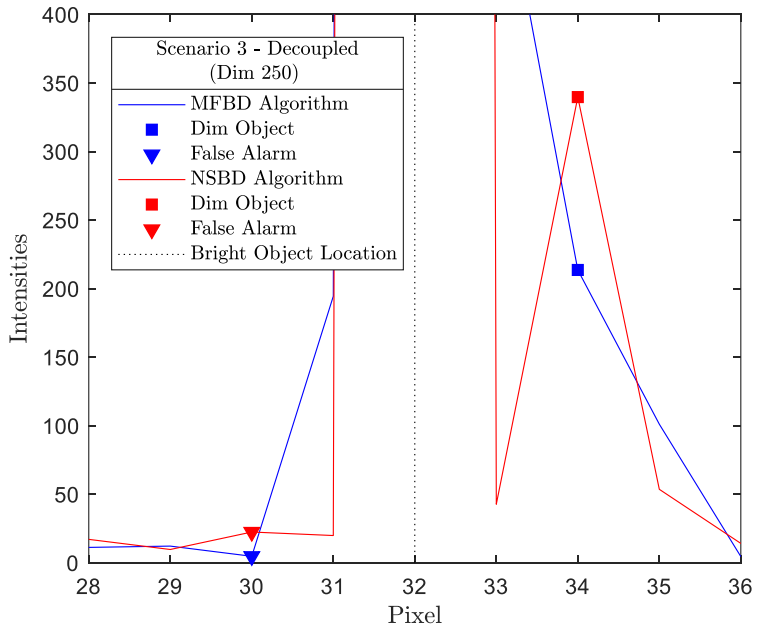


**Figure 4.27. The Average Functions for Scenario 3 for the Decoupled Approach.**

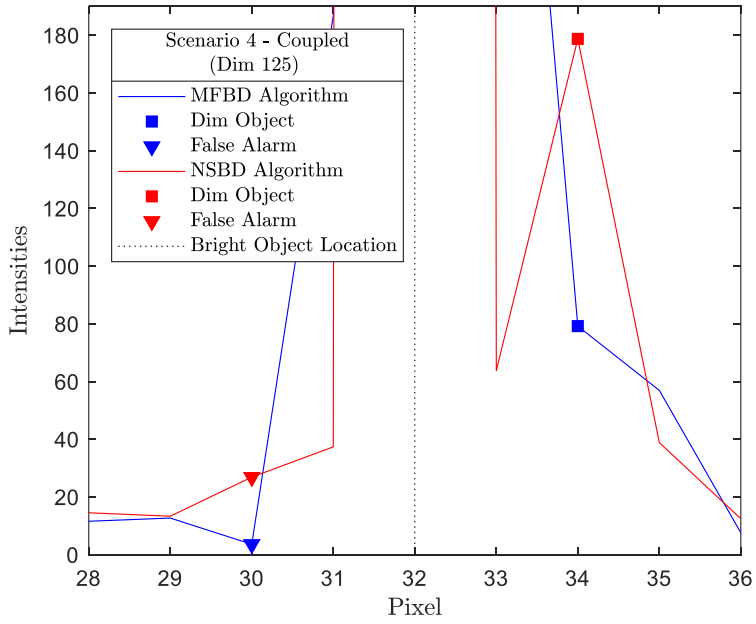


**Figure 4.28. The Average Functions for Scenario 4 for the Coupled Approach.**

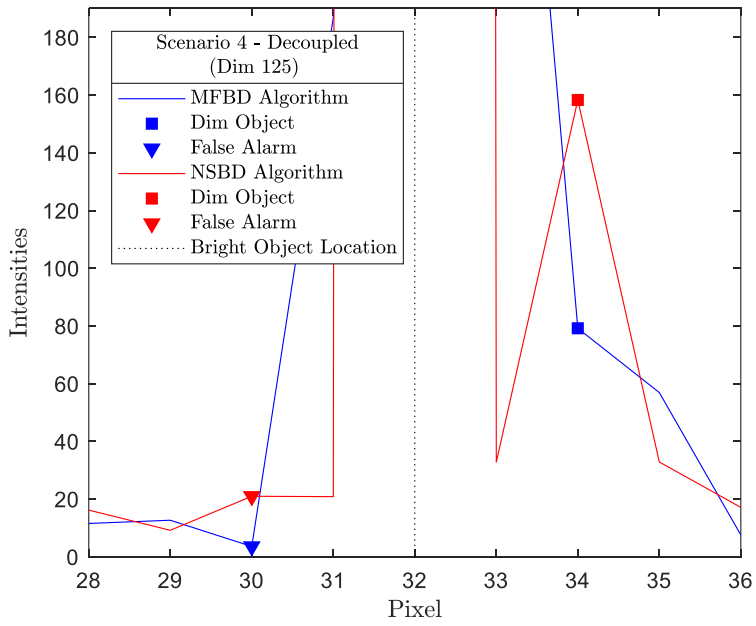


**Figure 4.29. The Average Functions for Scenario 4 for the Decoupled Approach.**

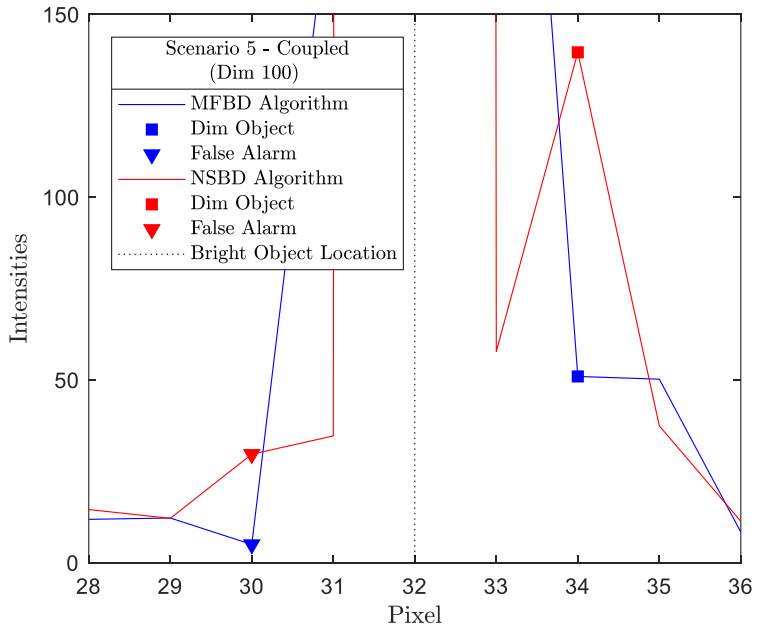


**Figure 4.30. The Average Functions for Scenario 5 for the Coupled Approach.**

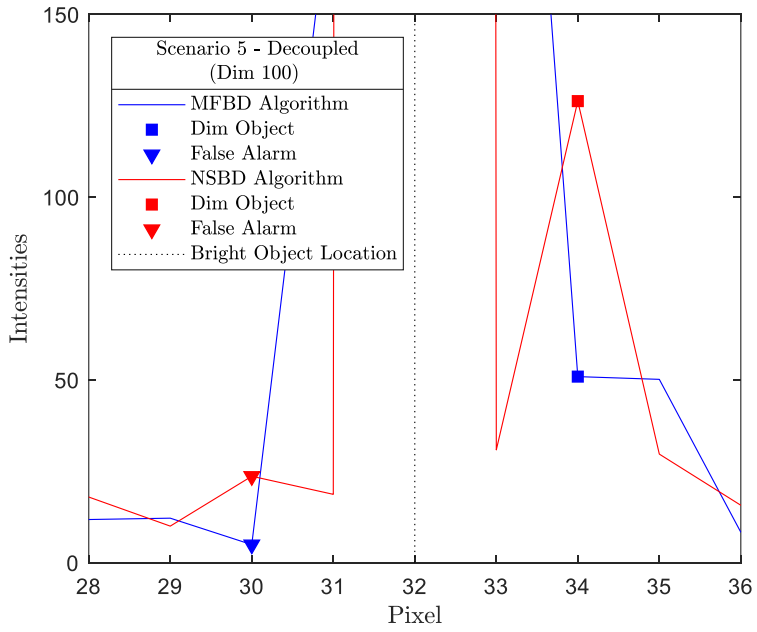


**Figure 4.31. The Average Functions for Scenario 5 for the Decoupled Approach.**

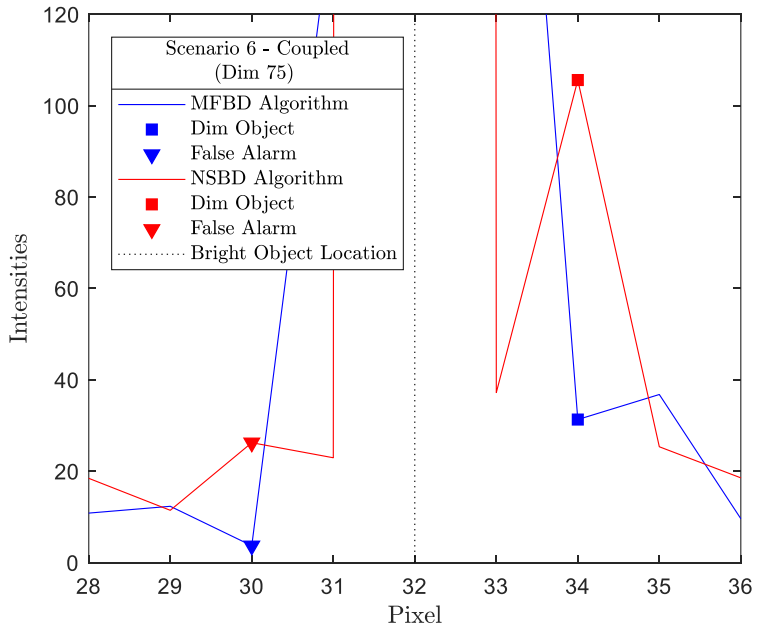


**Figure 4.32. The Average Functions for Scenario 6 for the Coupled Approach.**

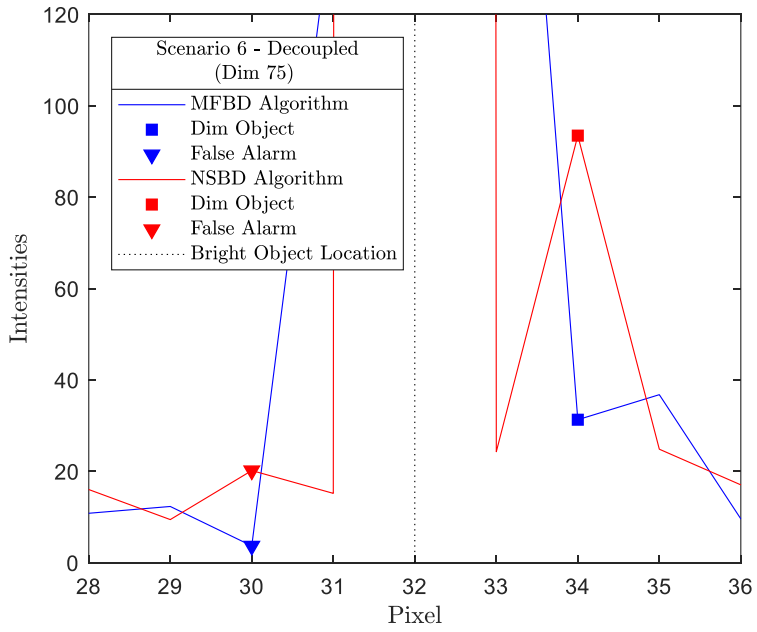


**Figure 4.33. The Average Functions for Scenario 6 for the Decoupled Approach.**

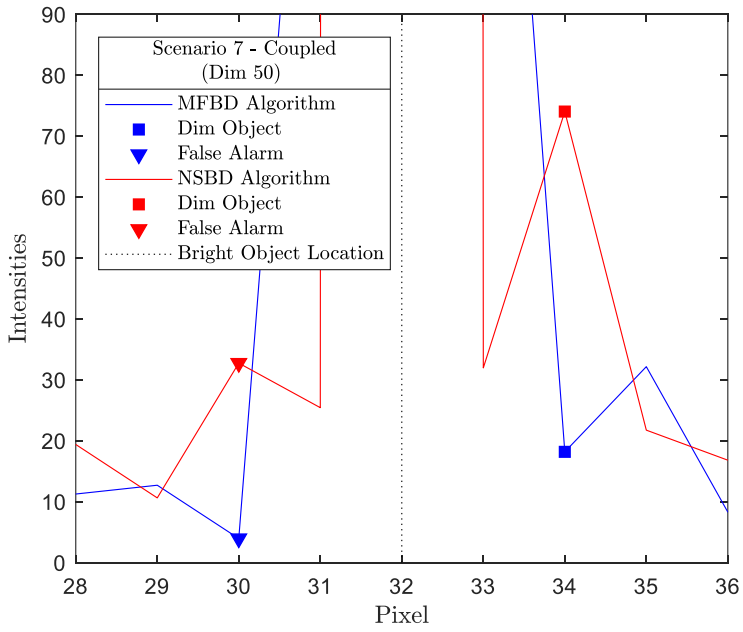


**Figure 4.34. The Average Functions for Scenario 7 for the Coupled Approach.**

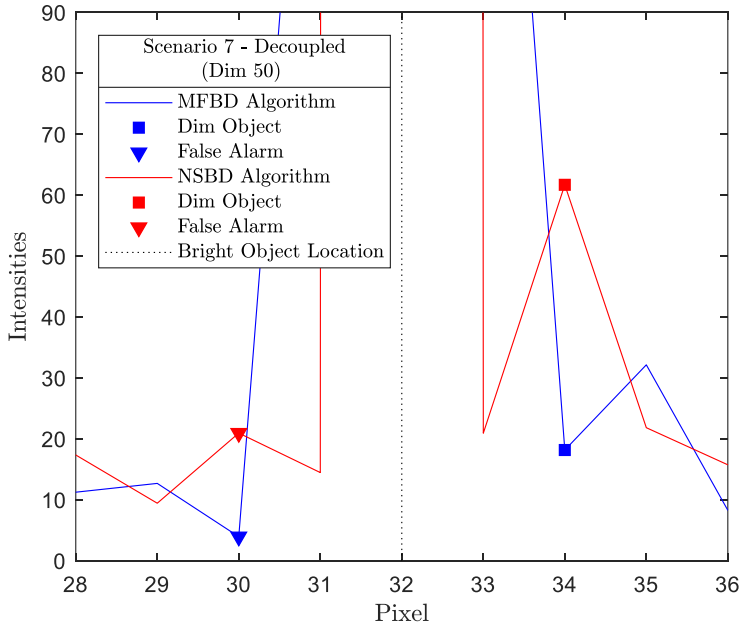


**Figure 4.35. The Average Functions for Scenario 7 for the Decoupled Approach.**

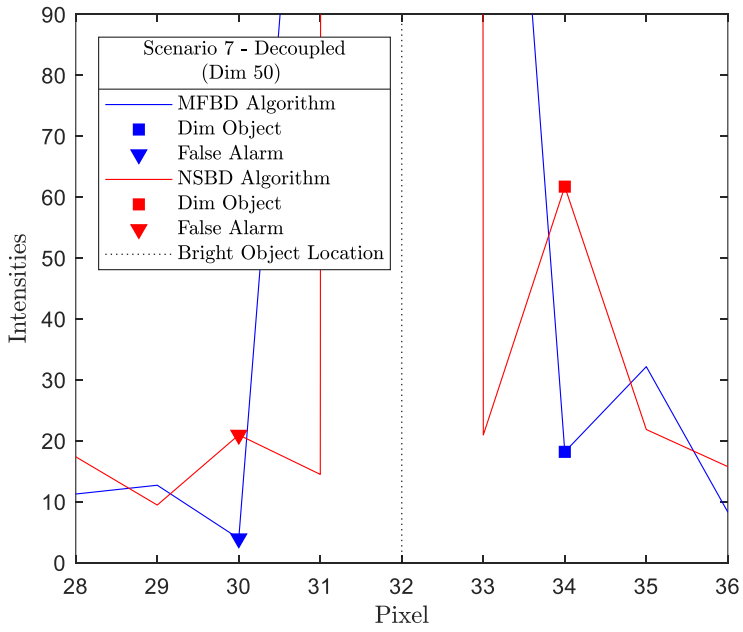


**Figure 4.36. The Average Functions for Scenario 8 for the Coupled Approach.**

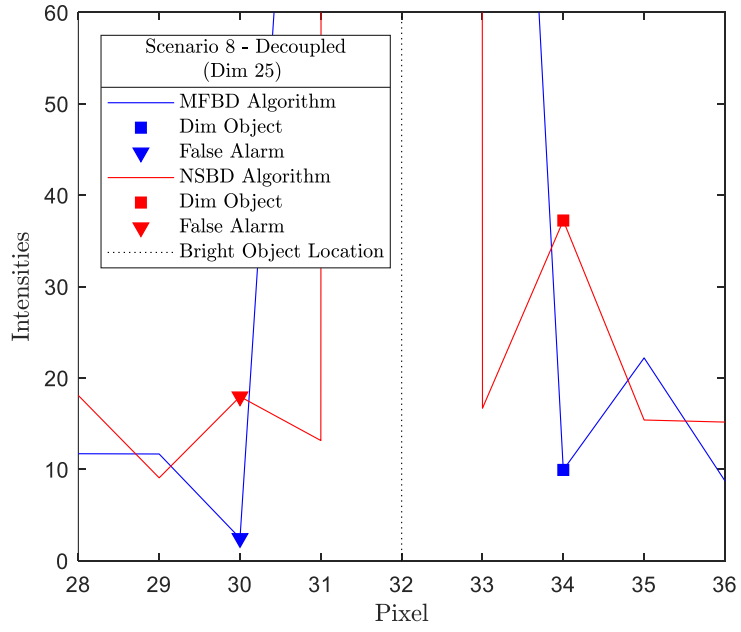


**Figure 4.37. The Average Functions for Scenario 8 for the Decoupled Approach.**

From the results, it can be seen that the NSBD algorithm outperforms the MFBD algorithm for all scenarios for both coupled and decoupled approaches in that the dim object is clearly discernible. In addition, the result from the coupled approach and that from the decoupled approach for each of the scenarios are comparable. The decoupled approach does not require the results from the MFBD algorithm and therefore saves the computational time and resources.

### 4.5. Summary

In summary, the new Neighborhood System Blind Deconvolution algorithm was developed and its performance was compared to that of the Multi-Frame Blind Deconvolution algorithm using two performance metrics, which are (1) dim object detection (2) dim object average. From the results, the Neighborhood System Blind Deconvolution algorithm outperformed the Multi-Frame Blind Deconvolution algorithm in both performance metrics. This chapter also used the confidence interval with the level of significance of $\alpha = 0.10$. For Scenario 1 through Scenario 7, the Neighborhood System Blind Deconvolution algorithm outperforms the Multi-Frame Blind Deconvolution algorithm in a majority of the operating regions and the performance is significantly different. For Scenario 8, even though the Neighborhood System Blind Deconvolution algorithm outperforms the Multi-Frame Blind Deconvolution algorithm, the performance is not significantly different. The Neighborhood System Blind Deconvolution algorithm also outperforms in the average function for all eight scenarios. In the next chapter, the algorithms are applied to the data collected at the Optics Laboratory of the Electrical and Computer Engineering Department at the Air Force Institute of Technology.

# 5. Undersampled Blind Deconvolution

## 5.1. Chapter Overview

In this chapter, the performance of the Multi-Frame Blind Deconvolution (MFBD) algorithm is compared with the Neighborhood System Blind Deconvolution (NSBD) algorithm using the data collected at the Optics Laboratory of the Electrical and Computer Engineering Department at the Air Force Institute of Technology. The setup for the laboratory experiment and the data collection process are explained. The data in this chapter represents an undersampled case. Therefore, the algorithms are modified to compensate for the undersampling effect, which is explained in Section 2.8. A set of computer simulated data is also generated to mimic the laboratory data, using the statistical photocalibration techniques. The effects of the detector gain and the readout noise are also added. The performance results are shown with the two performance metrics described in Section 3.5.

## 5.2. Laboratory Data

For the laboratory data, an object function is projected onto a computer monitor and a lens is placed approximately two meters away from the monitor to form an intermediate image. The lens has a diameter of 2.5 cm and F# of 10. The CCD is placed 30 cm away from the lens. A stream of hot air is blown across the lens to simulate atmospheric turbulence as shown in Figure 5.1. The CCD is a monochromatic scientific camera, Model Number 8050-TE-GE by Thorlab, with the pixel size of 5.5 μm×5.5 μm

[93]. Using Equation (2.81), the Nyquist pixel size for this experiment is calculated to be 3 μm×3 μm, creating an undersampling effects with $L$ = 1.83.

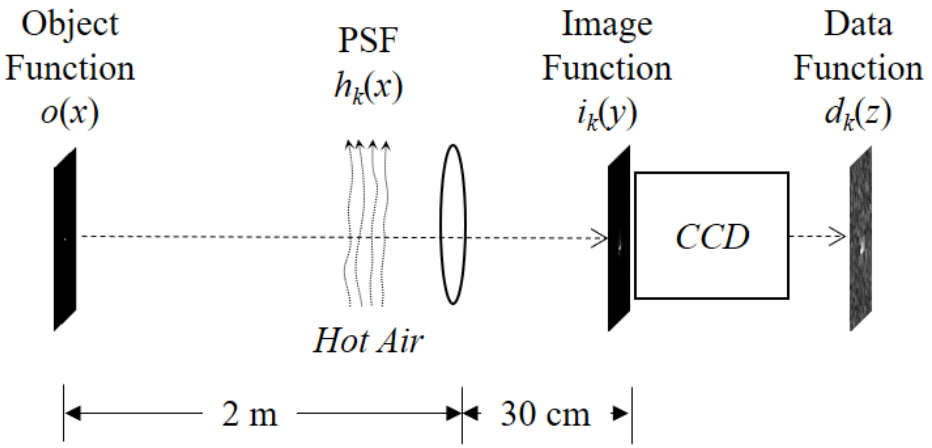


**Figure 5.1. Laboratory Setup for Collecting the Experimental Data.**

The object function projected on the monitor is created using MATLAB. It contains a bright object and a dim object. The intensity of the dim object is set at ten percent of the intensity of the bright object. The exposure time is set to 30 milliseconds. Even though the CCD can capture up to 3296×2472 pixels, only the regions with 64×64 pixels that contain the objects are captured. A total of 2,500 data functions are collected with one second delay between each collection. The one second delay allows enough time for the atmospheric effects to be uncorrelated [74]. The delay is also the limitation of the ThorLab camera to collect between any two frames. Next, a set of five data functions are grouped together to form one dataset, generating a total of 500 datasets.

The dim object and the bright object are estimated to be located six pixels apart. With the above conditions, it is very difficult to determine the center of the dim object,

even if the exposure time is increased to one second, as shown Figure 5.2 (a). To determine the dim object center, its intensity is temporarily increased to match that of the bright one as shown in Figure 5.2 (b).

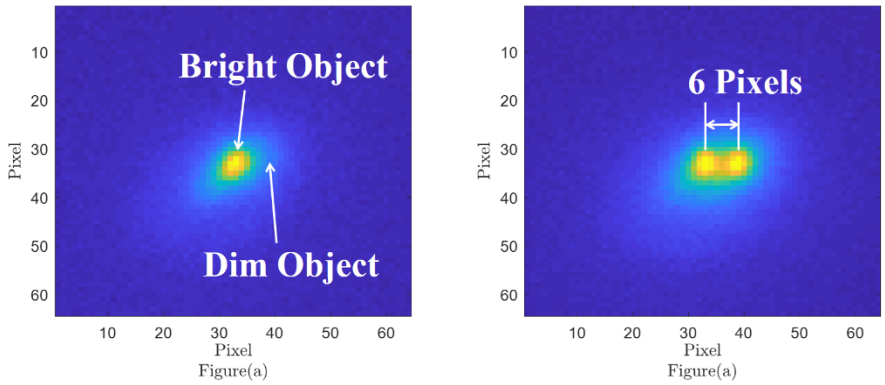


**Figure 5.2. Long Exposure Images for Determining Dim Object Location.**

## 5.3. Computer Generated Data

In addition to the laboratory data, computer generated data are also created to mimic the laboratory data. In Section 5.3.1, the average photon count, the variance of the readout noise, and the gain of the detector are estimated using the statistical photocalibration techniques [94], [95]. In Section 5.3.2, the remaining parameters are determined empirically.

### 5.3.1. Statistical Photocalibration

In this section, the average photon count per pixel, the variance of the readout noise, and the gain of the detector are estimated from the data functions, $d'(z)$ and $d''(z)$ that are

collected at two different exposure times. Their averages are given by the following equations [94], [95].

$$\bar{d}'(z) = G(z)\bar{K}(z) + B(z), \tag{5.1}$$

$$\bar{d}''(z) = N_G G(z)\bar{K}(z) + B(z), \tag{5.2}$$

where $\bar{d}'(z)$ and $\bar{d}''(z)$ are the means of the collected data with two different exposure times, $G(z)$ is the detector gain, $\bar{K}(z)$ is the average photon count, $N_G$ is the factor of the increase or reduction of the integration time between the two data functions, and $B(z)$ is the bias. The detector gain and the average photon count are then given by the following equations [94], [95].

$$G(z) = \frac{\sigma^2_{d''}(z) - \sigma^2_{d'}(z)}{\bar{d}''(z) - \bar{d}'(z)}, \tag{5.3}$$

$$\bar{K}(z) = \frac{\bar{d}''(z) - \bar{d}'(z)}{(N_G - 1)G(z)}, \tag{5.4}$$

$$\sigma^2_n(z) = \sigma^2_{d'}(z) - G^2(z)\bar{K}(z), \tag{5.5}$$

where $\sigma^2_{d''}(z)$ and $\sigma^2_{d'}(z)$ are the variances of the data functions and $\sigma^2_n(z)$ is the readout noise variance. In the previous section, the laboratory data were collected with 30 millisecond exposure time. This data function is designated as $d''(z)$. In the absence of the availability of the data functions without any objects, the corner regions of the data functions are good candidates to use for photocalibration because these regions experience minimal effects from the object intensities. Assuming that the gain of the detector remains the same over the imaging region, which is 64×64 pixel region, the top left 10×10 pixel regions from all 2,500 data functions from the previous section are used in calculating its

variance, $\sigma^2_{d'}(z)$. In addition, a different set of data collected at 15 millisecond exposure time, designated as $d'(z)$, is also used to calculate its variance, $\sigma^2_{d'}(z)$. The two distributions are shown in Figure 5.3.

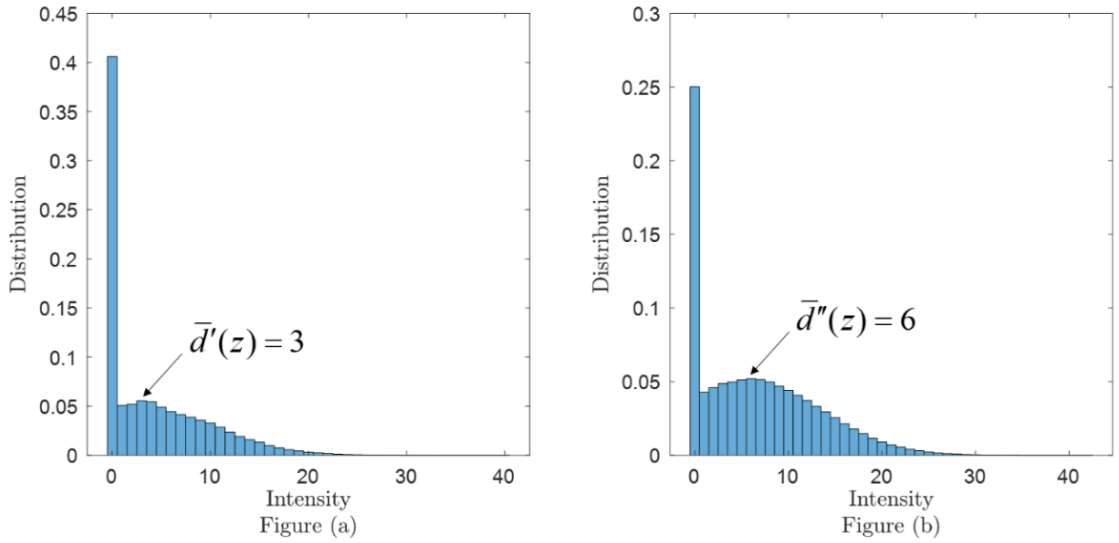


**Figure 5.3. Distribution of Data Collected at Two Different Exposure Times. (a) Data Collected with 15 Millisecond Exposure and (b) with 30 Millisecond Exposure.**

From the figures, the data for both exposures are cutoff on the left hand side of their distributions because the CCD only generates non-negative integers. In this section, the workarounds to estimate the parameters are explained.

First, the means of the two sets of data are determined using the median value because the mean and the median values of the Poisson and Normal distributions are close in values when the distributions are not truncated. The two means are estimated to be 3 and 6 for $d'(z)$ and $d''(z)$ respectively.

Next, the variances of the two sets of data are determined using the right side of the estimated means because the variance is a measurement of the differences between the mean and the observed values. Using this technique, the two variances are estimated to be 52.64 and 57.33 for $d'(z)$ and $d''(z)$ respectively.

With the means and variances, using Equation (5.3) and Equation (5.4), the detector gain, the average photon count, and the variance of the readout noise are estimated to be 1.56, 1.92, and 47.94 respectively. In Equation (5.4), the value of $N_G$ is 2 because the exposure time for $d''(z)$ is twice that for $d'(z)$.

### 5.3.2. Empirical Estimation

The computer generated data are then created to simulate the laboratory data. The goal is to mimic the average intensity of all laboratory data to that of all computer generated data through empirical estimation. From Section 5.2, it is determined that the dim object is located 6 pixels away and that the laboratory data is undersampled by a factor of approximately $L = 2$. Because the laboratory region is 64×64 pixels in size, the computer generated data is created on a 128×128 pixel plane. The bright object is placed at the center of the plane and the dim object is placed 12 pixels apart. The bright object and the dim object are created with a circular region of with 4 pixel radius as shown in Figure 5.4.

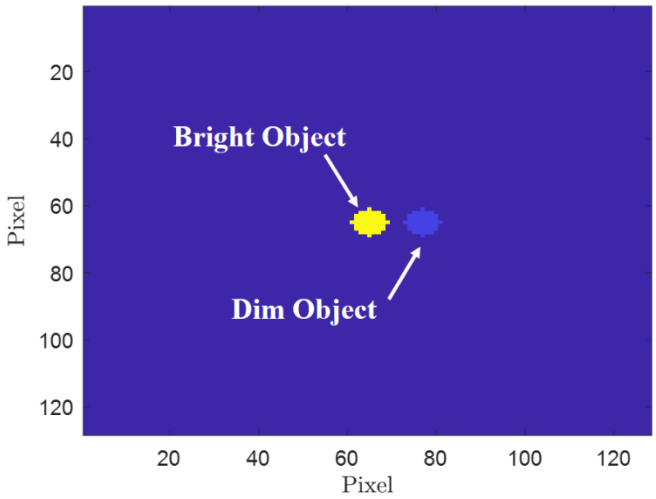


**Figure 5.4. The Object Function for Computer Simulated Data.**

The intensities are set in a way that the ratio between the bright object to the dim object is $1:0.10$ and the ratio between the bright object and the background is $1:0.0267$. From empirical estimation, the strength of the turbulence is estimated to be $D/r_0 = 10$. The Zernike phase screen generation technique described in Section 2.4.2 is used in generating the random atmospheric phase with 120 polynomials. Next, the image functions are formed with the resultant random and uncorrelated atmospheric PSFs. As described in Section 2.8, each image function is averaged and decimated by a factor of $L = 2$. Then, the CCD gain of $G = 1.56$ is multiplied after the signal-dependent Poisson noise is added to the decimated image function. The readout noise with variance of $\sigma_n^2 = 47.94$ is added to the Poisson noise to generate the data function. The gain and the readout noise values are obtained from Section 5.3.1. Since the CCD only generates non-negative integer values, the output of the simulated data is also rounded and any negative value is set to zero. Similar to the laboratory data, a total of 2,500 data functions are generated. Then, a

set of $K = 5$ data functions are grouped together to form a total of 500 datasets. Figure 5.5 compares the instances of the two data functions, each of which is randomly selected from the laboratory data and the computer simulated data respectively. Figure 5.6 compares the average of the 2,500 data functions from the laboratory and that from the computer simulated data. All the data functions are aligned first before averaged. A cross sectional view is shown along the axis where the centers of both the bright object and the dim object are located.

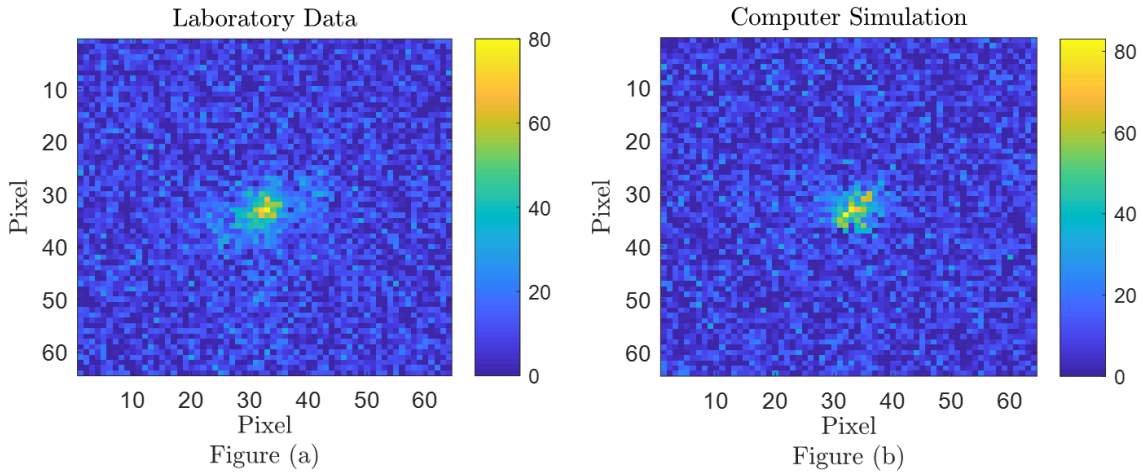


**Figure 5.5. Comparison between a Randomly Selected Data Function from the Laboratory Data and the Computer Simulated Data.**

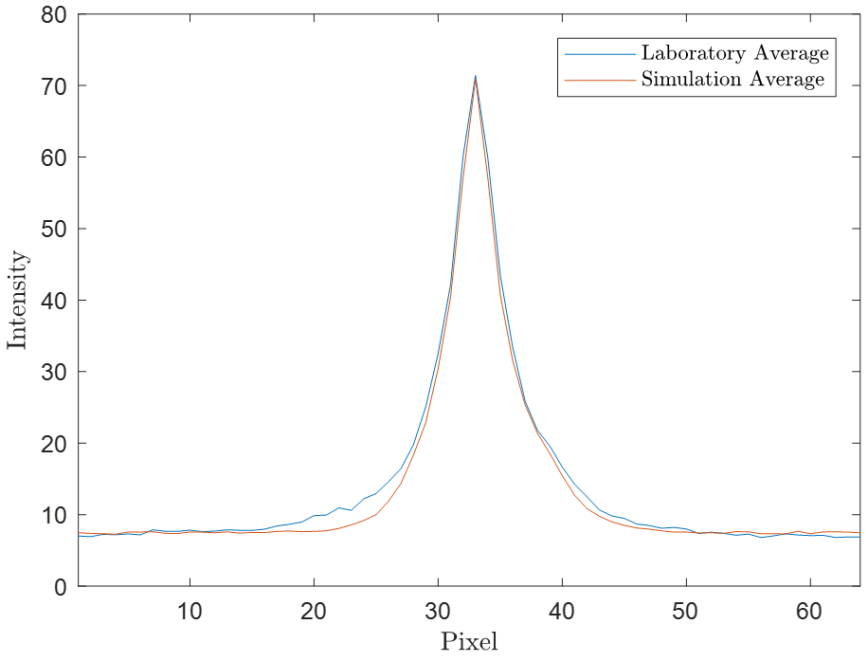


**Figure 5.6. Cross Sectional Comparison between the Average Intensity of All Laboratory Data Functions and that of All Computer Simulated Data Functions.**

## 5.4. Undersampled Blind Deconvolution

In this section, the MFBD and the NSBD are derived for the undersampled data. There are many interpolation techniques to reconstruct the images back on the Nyquist space but they introduce artifacts [96]. Even though these artifacts might not affect the human vision and perception in photography, they could become very problematic for closely spaced object detection, especially the dim ones. Therefore, in this section, the image reconstruction is done without interpolation, while taking into account the undersampling effect in the derivation.

### 5.4.1. Multi-Frame Blind Deconvolution

First, the complete data function is redefined to include the effects of the averaging function and the decimation. As described in Section 2.6.3, the incomplete data function, $\tilde{d}_k(z,x,\upsilon)$, is formed from a summation of the complete data functions, $d_k(z)$, as shown in the equation below. It is similar to Equation (2.56) but it has an additional summation term.

$$d_k(z) = \sum_{\forall \upsilon} \sum_{\forall x} \tilde{d}_k(z,x,\upsilon). \tag{5.6}$$

Taking the expectation of above equation results in the following.

$$\begin{aligned} \mathrm{E}\left[d_k(z)\right] &= \mathrm{E}\left[\sum_{\forall \upsilon} \sum_{\forall x} \tilde{d}_k(z,x,\upsilon)\right] \\ &= \sum_{\forall \upsilon} \sum_{\forall x} \mathrm{E}\left[\tilde{d}_k(z,x,\upsilon)\right]. \end{aligned} \tag{5.7}$$

It is shown in Equation (2.85) that the expectation of the data function is equal to the decimated image function which is restated below.

$$i_k^D(z) = \sum_{\forall \upsilon} \sum_{\forall x} o(x) r(\upsilon) h_k(Lz - x - \upsilon). \tag{5.8}$$

From Equation (5.6) and Equation (5.7), the expectation of the complete data function can be expressed as shown below.

$$\mathrm{E}\left[\tilde{d}_k(z,x,\upsilon)\right] = o(x) r(\upsilon) h_k(Lz - x - \upsilon). \tag{5.9}$$

The complete data function also follows a Poisson distribution because the sum of Poisson random variables is also a Poisson random variable. Therefore, the probability mass function (PMF) can be expressed as shown below.

$$\mathrm{P}\left(\tilde{d}_k(z,x,\upsilon)\right) = e^{-o(x)r(\upsilon)h_k(Lz-x-\upsilon)} \frac{\left(o(x)r(\upsilon)h_k(Lz-x-\upsilon)\right)^{\tilde{d}_k(z,x,\upsilon)}}{\tilde{d}_k(z,x,\upsilon)!}. \tag{5.10}$$

With the PMF defined, the joint log likelihood function can be defined as shown in the equation below. This approach is similar to the one shown in Section 2.6.3.

$$\begin{aligned}
\ell\left(o,h_k\right) &= \ln\left[\prod_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}} \mathrm{P}\left(\tilde{d}_k(z,x,\upsilon)\right)\right] \\
&= \sum_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}} \ln\left[\mathrm{P}\left(\tilde{d}_k(z,x,\upsilon)\right)\right] \\
&= \sum_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}} \ln\left[e^{-o(x)r(\upsilon)h_k(Lz-x-\upsilon)} \frac{\left(o(x)r(\upsilon)h_k(Lz-x-\upsilon)\right)^{\tilde{d}_k(z,x,\upsilon)}}{\tilde{d}_k(z,x,\upsilon)!}\right] \\
&= -\sum_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}} o(x)r(\upsilon)h_k(Lz-x-\upsilon) + \\
&\quad \sum_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}} \tilde{d}_k(z,x,\upsilon)\ln\left[o(x)r(\upsilon)h_k(Lz-x-\upsilon)\right] + \text{A.T.}
\end{aligned} \tag{5.11}$$

In the equation above, the term that does not affect the maximization is designated as the additional term (A.T.) [62]. This term will be ommitted in the subsequent equations. Next, in the Expectation Step of the Expectation-Maximization (EM) algorithm, the conditional expectation given the incomplete data function, *Q*, is calculated as shown below.

$$\begin{aligned}
Q\left(o,h_k\right) &= \mathrm{E}\left[\ell\left(o,h_k\right)\middle| d_k(z)\right] \\
&= -\sum_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}} o(x)r(\upsilon)h_k(Lz-x-\upsilon) + \\
&\quad \sum_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}} \mathrm{E}\left[\tilde{d}_k(z,x,\upsilon)\middle| d_k(z)\right]\ln\left[o(x)r(\upsilon)h_k(Lz-x-\upsilon)\right].
\end{aligned} \tag{5.12}$$

Since the complete data function is the only random variable, the conditional expectation applies only to it. The conditional expectation term can be derived as shown in Equation (5.13) because both the complete data and the incomplete data are Poisson random variables and the former is a part of the latter. In the EM algorithm, the new or updated function is iteratively updated from the estimations of the old estimates. Therefore, Equation (5.13) represents the old terms.

$$\mathrm{E}\left[\tilde{d}_k(z,x,\upsilon)\middle|d_k(z)\right]=\frac{o^{\mathrm{old}}(x)r(\upsilon)h_k^{\mathrm{old}}(Lz-x-\upsilon)}{i_k^{\mathrm{old}}(z)}d_k(z). \tag{5.13}$$

Substituting Equation (5.13) into Equation (5.12) results in the following equation.

$$\begin{aligned}Q(o,h_k)=&-\sum_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}} o(x)r(\upsilon)h_k(Lz-x-\upsilon)+\\ &\sum_{\substack{\forall k,\forall z,\\ \forall x,\forall \upsilon}}\frac{o^{\mathrm{old}}(x)r(\upsilon)h_k^{\mathrm{old}}(Lz-x-\upsilon)}{i_k^{\mathrm{old}}(z)}d_k(z)\ln\left[o(x)r(\upsilon)h_k(Lz-x-\upsilon)\right].\end{aligned} \tag{5.14}$$

In the Maximization Step of the EM algorithm, the conditional expectation term is maximized with respect to the object function and the PSFs to obtain the iterative update solutions. First, the conditional expectation function is differentiated with respect to each pixel of the object function.

$$\begin{aligned}\frac{\partial}{\partial o(x)}Q(o,h_k)=&-\sum_{\forall k,\forall z,\forall \upsilon} r(\upsilon)h_k(Lz-x-\upsilon)+\\ &\sum_{\forall k,\forall z,\forall \upsilon}\frac{o^{\mathrm{old}}(x)r(\upsilon)h_k^{\mathrm{old}}(Lz-x-\upsilon)}{i_k^{\mathrm{old}}(z)}d_k(z)\frac{1}{o(x)}.\end{aligned} \tag{5.15}$$

The first term on the right hand side of the equation sums up to be $K/L^2$. It is because the PSF is constrained to sum up to unity to ensure that the process is conservative. A convolution between a unity and the averaging function also results in unity. Decimation

of the unity by a factor of $L$ in both dimensions reduces the value of its summation by a factor of $L^2$. Performing this over $K$ frames results in $K/L^2$. Therefore, Equation (5.15) can be simplified as shown below.

$$\frac{\partial}{\partial o(x)} Q(o, h_k) = -\frac{K}{L^2} + \frac{o^{\text{old}}(x)}{o^{\text{new}}(x)} \sum_{\forall k, \forall z, \forall \upsilon} \frac{r(\upsilon) h_k^{\text{old}}(Lz - x - \upsilon)}{i_k^{\text{old}}(z)} d_k(z). \tag{5.16}$$

Setting the derivative to zero gives us the following update equation for the object function.

$$o^{\text{new}}(x) = \frac{L^2}{K} o^{\text{new}}(x) \sum_{\forall k} \left( \sum_{\forall z} \frac{d_k(z)}{i_k^{\text{old}}(z)} \left( \sum_{\forall \upsilon} r(\upsilon) h_k^{\text{old}}(Lz - x - \upsilon) \right) \right). \tag{5.17}$$

The PSF update equation can be derived similarly, using change of summation variables. The new PSF is also constrained to sum up to unity by dividing with some constant, $U^{\text{new}}$.

$$h_k^{\text{new}}(x) = \frac{1}{U^{\text{new}}} h_k^{\text{old}}(x) \left( \sum_{\forall z} \frac{d_k(z)}{i_k^{\text{old}}(z)} \left( \sum_{\forall \upsilon} r(\upsilon) o^{\text{old}}(Lz - x - \upsilon) \right) \right). \tag{5.18}$$

Since the EM algorithm is derived to overcome the photon counting noise, the Gerchberg-Saxton phase retrieval algorithm is applied on the PSF updates to overcome the atmospheric turbulence.

### 5.4.2. Neighborhood System Blind Deconvolution

Similar to the technique mentioned in Chapter 4, to derive the NSBD, the data function is separated into the primary object function, $o^p(x)$, the neighborhood system function, $o^s(x)$, and the background function, $o^b(x)$. The assumption is made that the intensity of the three intensities are statistically independent of each other.

$$o(x) = o^p(x) + o^s(x) + o^b(x). \tag{5.19}$$

Because the system is modeled as linear and shift-invariant, the data function can also be separated into three data functions. The superscripts, *p*, *s*, and *b* represent the primary function, the neighborhood system, and the background function respectively. The subscript, *k*, represents the $k^{\text{th}}$ frame.

$$d_k(z) = d_k^p(z) + d_k^s(z) + d_k^b(z). \tag{5.20}$$

Similar to Section 5.4.1, the corresponding complete data functions and their Poisson PMFs are derived. Next, the conditional expectation function, *Q*, is calculated by taking the conditional expectation of the joint log-likelihood function given the observed incomplete data functions, similar to Equation (5.12). Maximizing the conditional expectation function with respect to the primary object function, the neighborhood system function, the background function, and the PSF results in the following update equations.

$$o^{p^{\text{new}}}(x) = \frac{L^2}{K} o^{p^{\text{old}}}(x) \sum_{\forall k} \left( \sum_{\forall z} \frac{d_k(z)}{i_k^{\text{old}}(z)} \left( \sum_{\forall \upsilon} r(\upsilon) h_k^{\text{old}}(Lz - x - \upsilon) \right) \right), \tag{5.21}$$

$$o^{s^{\text{new}}}(x) = \frac{L^2}{K} o^{s^{\text{old}}}(x) \sum_{\forall k} \left( \sum_{\forall z} \frac{d_k(z)}{i_k^{\text{old}}(z)} \left( \sum_{\forall \upsilon} r(\upsilon) h_k^{\text{old}}(Lz - x - \upsilon) \right) \right), \tag{5.22}$$

$$o^{b^{\text{new}}}(x) = \frac{L^2}{K} o^{b^{\text{old}}}(x) \sum_{\forall k} \left( \sum_{\forall z} \frac{d_k(z)}{i_k^{\text{old}}(z)} \left( \sum_{\forall \upsilon} r(\upsilon) h_k^{\text{old}}(Lz - x - \upsilon) \right) \right), \tag{5.23}$$

$$h_k^{\text{new}}(x) = \frac{1}{U^{\text{new}}} h_k^{\text{old}}(x) \left( \sum_{\forall z} \frac{d_k(z)}{i_k^{\text{old}}(z)} \left( \sum_{\forall \upsilon} r(\upsilon) o^{\text{old}}(Lz - x - \upsilon) \right) \right). \tag{5.24}$$

Similar to the previous section, the Gerchberg-Saxton phase retrieval algorithm is also applied on the PSF updates from the EM algorithm. The PSF update equations remain the same as the one from the previous section.

## 5.5. Performance Comparison

In this section, the 500 datasets from the laboratory collected data and another 500 datasets from the computer generated data are processed with the MFBD and the NSBD algorithms. Each dataset contains $K = 5$ data functions. Since the undersampled factor is $L = 2$, the estimated object functions are reconstructed on a 128×128 pixel plane even though the data functions are of 64×64 pixels in size. A total of 200 EM iterations are completed for both algorithms. A total of 100 Gerchberg-Saxton phase retrieval iterations are completed for each EM iteration.

### 5.5.1. Results without Prior Knowledge

For the first set of results, the concept of blind deconvolution is applied in a sense that no assumption about the primary object function is made. The primary object function is assigned as a Dirac delta and the neighborhood system function is initialized with a circular region with a radius of 30 pixels, which is large enough to over the region where the dim object exists. The primary object function and the neighborhood system function are mutually exclusive as shown in Figure 5.7.

The intensity of the primary object function is initialized with the average of the highest pixels from the five data functions. That of the neighborhood system function is initialized with twice the background estimate, which can be easily computed from the corner regions of the data functions where there are very minimal influences from the

primary object and the dim object. These exact initial conditions are not required for the algorithm. As long as the intensity of the primary function is higher than that of the neighborhood system function, which in turn is higher than that of the background function, the NSBD will provide consistent results. For the PSF, all five are initialized as the diffraction-limited PSF, where the phases are initialized as planes of zeros.

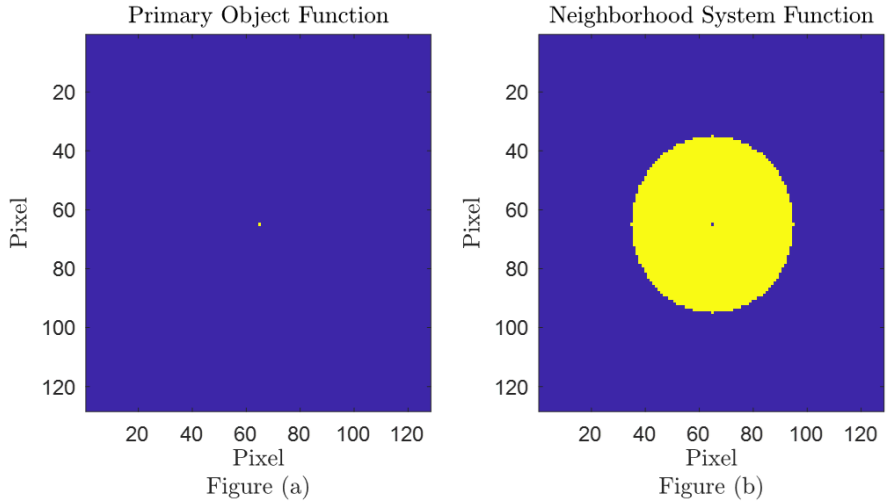


**Figure 5.7. (a) Primary Object Function and (b) Neighborhood System Function used in the NSBD Algorithm without any Prior Knowledge.**

To use the ROC function as a performance metric, the intensities of the dim object and the false alarm from the 500 estimated object functions from the output of each algorithm are first estimated. To calculate the dim object intensity, a 4-pixel radius circular region is used. It is centered 12 pixels away from the center of the primary object function where the dim object is placed. The total value of the estimated object function in the circular region is used as the intensity of the dim object. For the false alarm intensity, a 4-

pixel radius circular area which is located 12 pixels away in the opposite direction is selected as shown below.

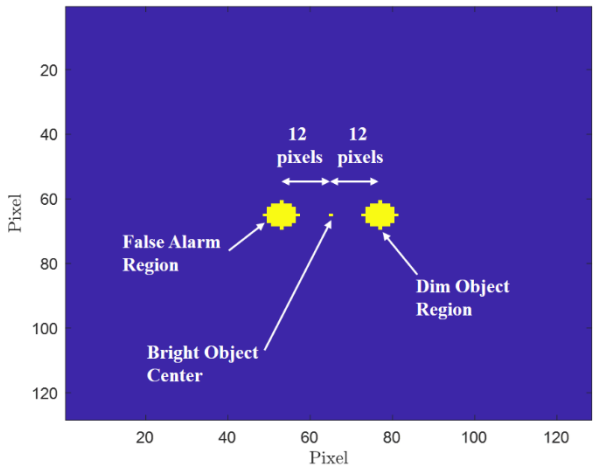


**Figure 5.8. The Regions for Calculating False Alarm and Dim Object Intensities.**

The intensities are fitted using the Gaussian distributions and the ROC functions are compared in Figure 5.9 and Figure 5.10 for the laboratory data and the computer generated data respectively. The reason for selecting the Gaussian distribution is that, with the exception of the distribution for the false alarm intensities from the computer generated data processed with the NSBD algorithm, the three other distributions fail to reject the null hypothesis that the data comes from a distribution in the normal family [97], [98]. The comparison between the unfitted data and the fitted data are shown in Appendix C. From the results, the NSBD outperforms the MFBD for both the laboratory data and the computer generated data, even in the presence of the readout noise. However, with an 80 percent confidence interval $(\alpha = 0.20)$, the lower confidence bound of the NSBD overlaps with the upper confidence bound of the MFBD.

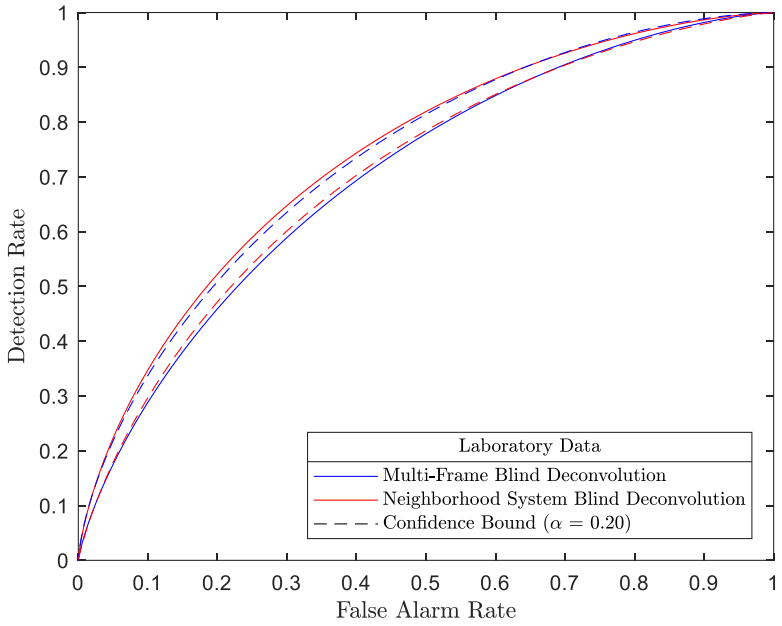


**Figure 5.9. The ROC Comparison for the Laboratory Data.**

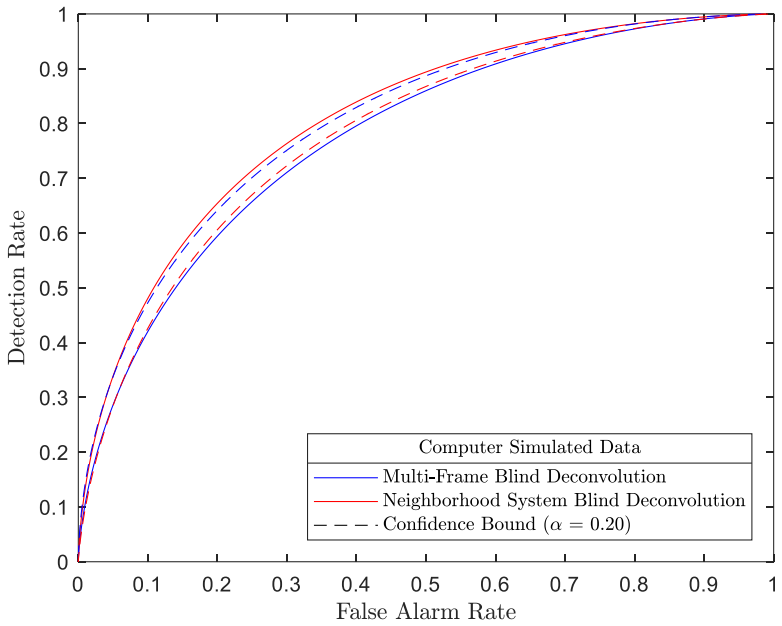


**Figure 5.10. The ROC Comparison for the Computer Generated Data.**

### 5.5.2. Results with Additional Knowledge

The advantage of the NSBD is that the further assumptions can be made about the primary object function. In the next set of results, the assumption is made that the approximate shape of the primary object function is known. The primary object function is then initialized with a circular region with 4-pixel radius. This knowledge is based on the computer simulated data, even though the exact knowledge for the laboratory data is unknown. The neighborhood system function is kept the same at 30-pixel radius as shown in Figure 5.11.

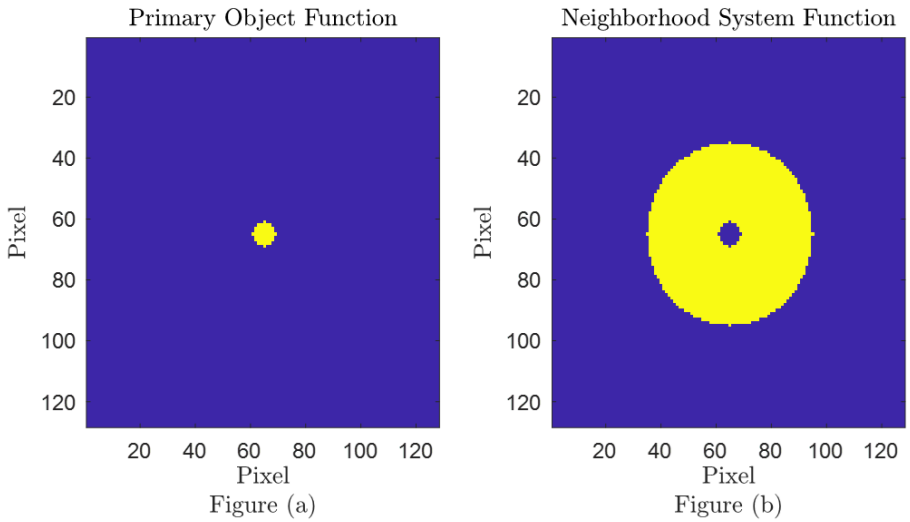


**Figure 5.11. (a) Primary Object Function and (b) Neighborhood System Function used in the NSBD Algorithm with Prior Shape Knowledge.**

For the primary object function initialization, the whole circular region is assigned with the average of the highest pixels from the five data functions from each dataset. For the neighborhood system function, the background function, the PSFs, and the phases are initialized in the exact same manner as described previously. In addition, the technique for

calculating the dim object intensity and the false alarm intensity are also kept the same for generating the ROC functions. The number of EM iterations and phase retrieval iterations are also kept the same. The intensities are also fitted with the Gaussian distributions. With the exception of the distribution for the false alarm intensities from the computer generated data processed with the NSBD algorithm, the three other distributions also fail to reject the null hypothesis that the data comes from a distribution in the normal family. The results are shown in Figure 5.12 and Figure 5.13.

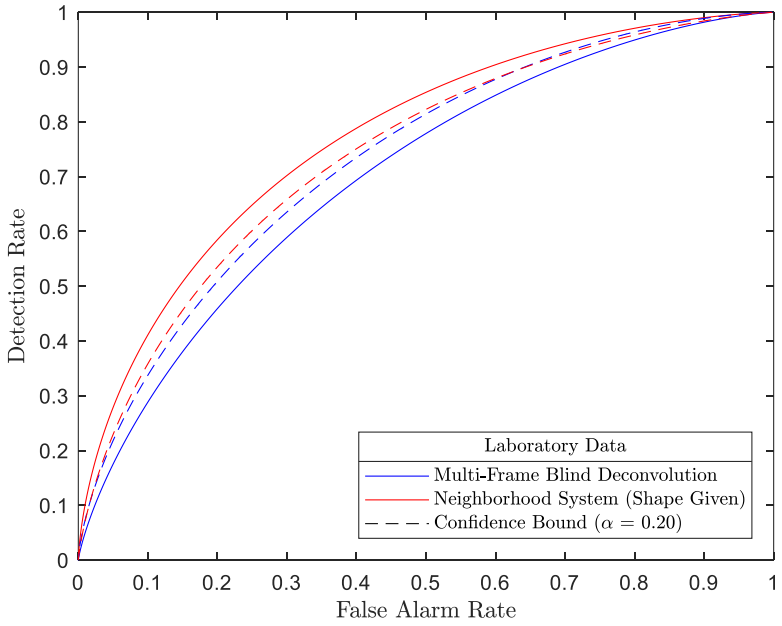


**Figure 5.12. The ROC Comparison for the Laboratory Data with Additional Knowledge about the Shape of the Primary Object Function for the Neighborhood System Algorithm.**

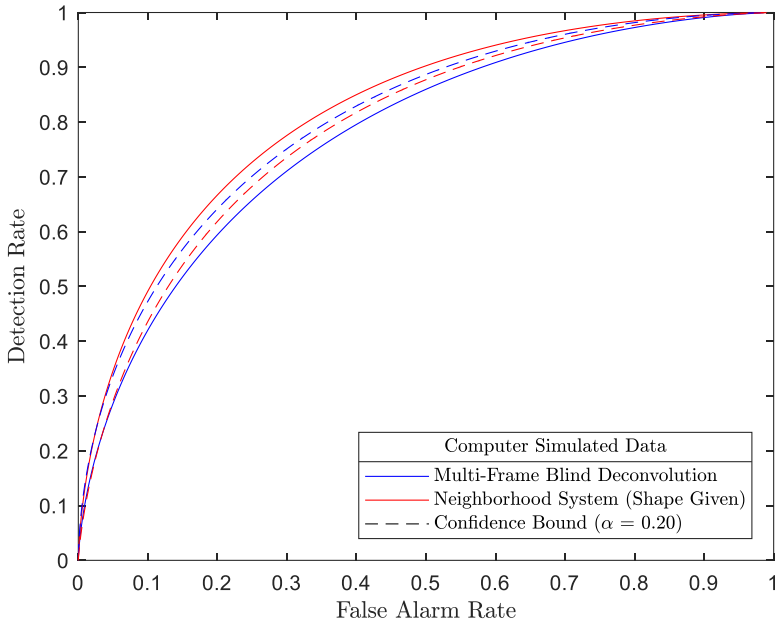


**Figure 5.13. The ROC Comparison for the Computer Simulated Data with Additional Knowledge about the Shape of the Primary Object Function for the Neighborhood System Algorithm.**

From the results, when an estimated primary object function shape is given, the NSBD algorithm can provide an additional performance increase over the blind technique. To be consistent with the previous chapters, the averages of the estimated object functions processed with the MFBD algorithm and the NSBD algorithm for the laboratory data and the computer generated data are shown in Figure 5.14 and Figure 5.15. From the results, it can be seen that the average of the intensities of the dim object are more pronounced for the NSBD algorithm with the primary object function shape given.

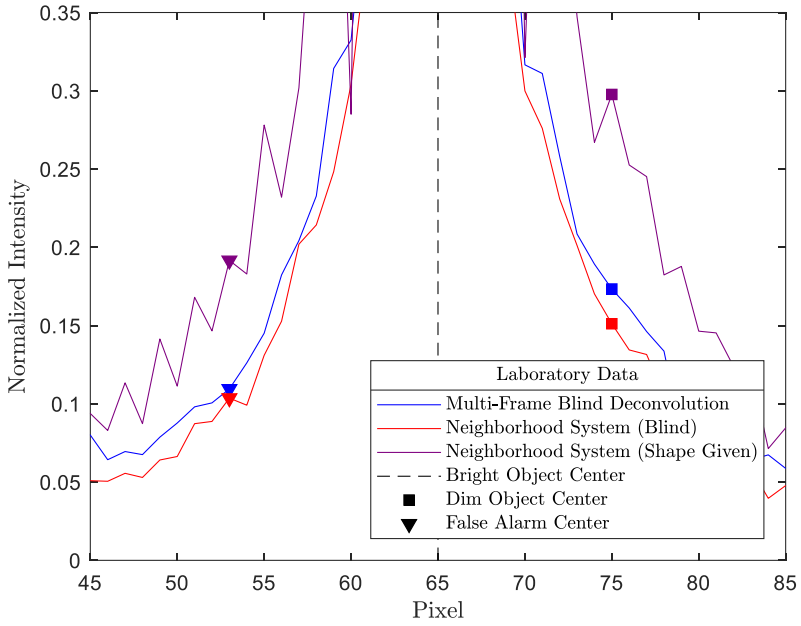


**Figure 5.14. The Average Functions of the Intensities of the Estimated Object Functions Processed with the MFBD and the NSBD for the Laboratory Data.**

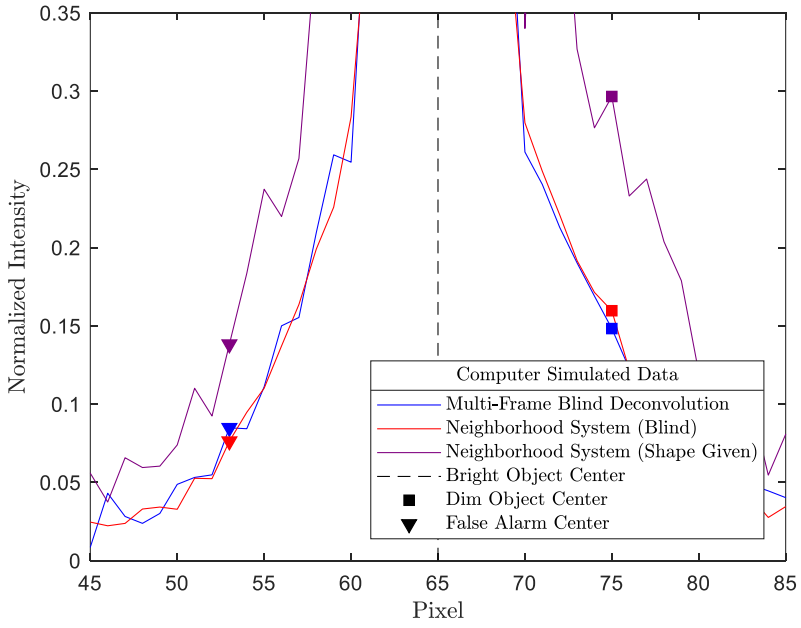


**Figure 5.15. The Average Functions of the Intensities of the Estimated Object Functions Processed with the MFBD and the NSBD for the Computer Simulated Data.**

## 5.6. Summary

In summary, this chapter showed the performance comparison between the Multi-Frame Blind Deconvolution algorithm and the Neighborhood System Blind Deconvolution algorithm for the laboratory data collected in the Optics Laboratory at the Air Force Institute of Technology. The data was undersampled by a factor of two. The algorithms were modified to take into account the undersampling effects in the derivation. A set of computer generated data was created to simulate the laboratory data using the statistical photocalibration techniques. The effects of the detector gain and the readout noise were also introduced. The data from the two experiments were processed with both the Multi-Frame Blind Deconvolution algorithm and the Neighborhood System Blind Deconvolution algorithm. The Gerchberg-Saxton phase retrieval algorithm was also applied in conjunction with both algorithms. The estimated object functions were compared using the Receiver Operating Characteristic function and the average function. From the results, the Neighborhood System Blind Deconvolution algorithm outperformed the Multi-Frame Blind Deconvolution algorithm. Furthermore, additional information about the primary object function can be given prior to processing with the Neighborhood System Blind Deconvolution. With the additional information, the performance of the Neighborhood System Blind Deconvolution was further increased over the situation where no prior information was given.

# 6. Dimension Reduction Blind Deconvolution

## 6.1. Chapter Overview

In this chapter, the Dimension Reduction Blind Deconvolution (DRBD) is derived. This approach is built on the One-Dimensional Multi-Frame Blind Deconvolution algorithm (1DBD) [99]. The 1DBD algorithm assumes that the object function is spatially separable, meaning that it can be described by an outer product between two one-dimensional (1D) vectors. The DRBD algorithm is derived to make two improvements. The first improvement is the ability to detect the dim objects that are in close proximity to a bright one. The second improvement is the ability to perform image processing when the object function is no longer spatially separable and cannot be described by an outer product between two 1D vectors. The results are compared with the Neighborhood System Blind Deconvolution (NSBD) as well as the Multi-Frame Blind Deconvolution (MFBD).

## 6.2. One-Dimensional Multi-Frame Blind Deconvolution

The algorithm makes the assumption that the objects are spatially separable, where the object function can be described as the outer product between the two 1D object functions, as shown in the equation below.

$$\mathbf{o} = \mathbf{o}_1 \otimes \mathbf{o}_2 , \tag{6.1}$$

where $\mathbf{o}_1$ and $\mathbf{o}_2$ are the $N\times1$ and $1\times M$ vectors respectively and $\otimes$ is the matrix multiplication between the two 1D object functions. Each element can also be expressed as the following.

$$o(x_1, x_2) = o_1(x_1) o_2(x_2) , \tag{6.2}$$

$$\mathbf{o}_1 \triangleq o_1(x_1), \quad \text{where } x_1 \in \{1,2,...,N\}, \tag{6.3}$$

$$\mathbf{o}_2 \triangleq o_2(x_2), \quad \text{where } x_2 \in \{1,2,...,M\}, \tag{6.4}$$

$$\mathbf{o} \triangleq o(x_1,x_2), \quad \text{where } x_1 \in \{1,2,...,N\} \text{ and } x_2 \in \{1,2,...,M\}, \tag{6.5}$$

where $x_1$ and $x_2$ are the discretized region over which the *Y*-axis and *X*-axis are defined and $o_1$ and $o_2$ with $x_1$ and $x_2$ indices represent the elements of the 1D vectors respectively. In other words, the rank of the object function is one [100]. The object function is assumed to meet the Nyquist sampling criteria.

To simplify the derivation, the assumption is made that the object function is a square function, where $N = M$. Throughout this chapter, each variable is used for the 1D coordinate for each axis, such as $x_1$, $x_2$, $y_1$, and $y_2$, where the subscript annotates the direction. The subscript "1" is used for *Y*-axis coordinates whereas the subscript "2" is used for the *X*-axis coordinates. This is a departure from the previous chapters where one variable is used to express a two-dimensional coordinate.

### 6.2.1. Algorithm Development

Using the same imaging model shown in Figure 2.10, the object function passes through a set of uncorrelated atmospheric point spread functions (PSF) to form the image functions. When the image functions arrive at the charge-coupled devices (CCD), they are converted to the data functions. From the data functions, the total intensity of the object function can be derived by averaging the total intensities of the data functions.

$$\frac{1}{K}\sum_{\forall k,\forall y_1,\forall y_2} d_k(y_1,y_2) = \sum_{\forall y_1,\forall y_2}\left(\frac{1}{K}\sum_{k=1}^{K} d_k(y_1,y_2)\right), \tag{6.6}$$

where $K$ is the total number of PSFs used in multi-frame image processing. To make the equations compact, the nested summations are annotated with multiple variables in the subscript. Recognizing that the term inside the parentheses in Equation (6.6) is the sample mean, the above equation is approximated as the statistical mean as shown below.

$$\frac{1}{K}\sum_{\forall k,\forall y_1,\forall y_2} d_k(y_1,y_2) \approx \sum_{\forall y_1,\forall y_2} \mathrm{E}\left[d_k(y_1,y_2)\right]. \tag{6.7}$$

From Equation (2.55), the Poisson mean of the data function is equal to the image function, which is formed from the convolution between the object function and the atmospheric PSF. Therefore, the above equation can be written as the following.

$$\begin{aligned}\frac{1}{K}\sum_{\forall k,\forall y_1,\forall y_2} d_k(y_1,y_2) &\approx \sum_{\forall y_1,\forall y_2} i_k(y_1,y_2) \\ &= \sum_{\forall y_1,\forall y_2}\left(\sum_{\forall x_1,\forall x_2} o(x_1,x_2)h_k(y_1-x_1,y_2-x_2)\right).\end{aligned} \tag{6.8}$$

Next, the nested summation is rearranged. Then, the concept of the conservative nature of the PSF or the constraint to sum up to unity is applied. The above equation can be simplified as the following.

$$\begin{aligned}\frac{1}{K}\sum_{\forall k,\forall y_1,\forall y_2} d_k(y_1,y_2) &\approx \sum_{\forall x_1,\forall x_2}\left(o(x_1,x_2)\sum_{\forall y_1,\forall y_2} h_k(y_1-x_1,y_2-x_2)\right) \\ &= \sum_{\forall x_1,\forall x_2}\left(o(x_1,x_2)\cdot 1\right) \\ &= \sum_{\forall x_1,\forall x_2} o(x_1,x_2) \\ &\triangleq \eta.\end{aligned} \tag{6.9}$$

where $\eta$ is defined as the total intensities of the objects in the object function. In addition, a constraint is made in such a way that the total intensity of the function $o_2$ is equal one.

$$\sum_{\forall x_2} o_2(x_2) = 1. \tag{6.10}$$

As a result, the total intensity of the $o_1$ object function is equal to $\eta$, as shown below.

$$\begin{aligned}
\sum_{\forall x_1} o_1(x_1) &= \sum_{\forall x_1} o_1(x_1) \cdot 1 \\
&= \sum_{\forall x_1} o_1(x_1) \left( \sum_{\forall x_2} o_2(x_2) \right) \\
&= \sum_{\forall x_1, \forall x_2} o_1(x_1) o_2(x_2) \\
&= \sum_{\forall x_1, \forall x_2} o(x_1, x_2) \\
&= \eta.
\end{aligned} \tag{6.11}$$

As explained in Section 2.6.2, the concept of complete and incomplete functions is applied on the image functions and the data functions as shown below.

$$\begin{aligned}
i_k(y_1, y_2) &= \sum_{\forall x_1, \forall x_2} o(x_1, x_2) h_k(y_1 - x_1, y_2 - x_2) \\
&= \sum_{\forall x_1, \forall x_2} \tilde{i}_k(x_1, x_2, y_1, y_2),
\end{aligned} \tag{6.12}$$

where $\tilde{i}_k$ is the complete image function which is defined as shown below.

$$\tilde{i}_k(x_1, x_2, y_1, y_2) \triangleq o(x_1, x_2) h_k(y_1 - x_1, y_2 - x_2). \tag{6.13}$$

As explained in Section 2.3, the sum of Poisson random variables results in a Poisson random variable, the incomplete data function can also be expressed as a sum of complete data functions, as shown in the equations below.

$$\begin{aligned}
\mathrm{E}\left[ d_k(y_1, y_2) \right] &= i_k(y_1, y_2) \\
&= \sum_{\forall x_1, \forall x_2} \tilde{i}_k(x_1, x_2, y_1, y_2) \\
&= \sum_{\forall x_1, \forall x_2} \mathrm{E}\left[ \tilde{d}_k(x_1, x_2, y_1, y_2) \right],
\end{aligned} \tag{6.14}$$

where $\tilde{d}_k$ is the complete data function which can be expressed as shown below.

$$\mathrm{E}\left[\tilde{d}_k(x_1, x_2, y_1, y_2)\right] = \tilde{i}_k(x_1, x_2, y_1, y_2). \tag{6.15}$$

The probability mass function of the complete data function can then be expressed as the following.

$$\mathrm{P}\left(\tilde{d}_k(x_1, x_2, y_1, y_2)\right) = \frac{\mathrm{e}^{-\tilde{i}_k(x_1, x_2, y_1, y_2)}\, \tilde{i}_k(x_1, x_2, y_1, y_2)^{\tilde{d}_k(x_1, x_2, y_1, y_2)}}{\tilde{d}_k(x_1, x_2, y_1, y_2)!}. \tag{6.16}$$

Using the natural logarithm (ln), the log likelihood function is derived, similar to the derivation in Section 2.6.2.

$$\begin{aligned}
\ell(o_1, o_2, h_k) &= \ln\left(\prod_{\substack{\forall x_1, \forall x_2, \\ \forall y_1, \forall y_2, \\ \forall k}} \mathrm{P}\left(\tilde{d}_k(x_1, x_2, y_1, y_2)\right)\right) \\
&= \sum_{\substack{\forall x_1, \forall x_2, \\ \forall y_1, \forall y_2, \\ \forall k}} \ln \mathrm{P}\left(\tilde{d}_k(x_1, x_2, y_1, y_2)\right) \\
&= \sum_{\substack{\forall x_1, \forall x_2, \\ \forall y_1, \forall y_2, \\ \forall k}} \ln \frac{\mathrm{e}^{-\tilde{i}_k(x_1, x_2, y_1, y_2)}\, \tilde{i}_k(x_1, x_2, y_1, y_2)^{\tilde{d}_k(x_1, x_2, y_1, y_2)}}{\tilde{d}_k(x_1, x_2, y_1, y_2)!} \\
&= \sum_{\substack{\forall x_1, \forall x_2, \\ \forall y_1, \forall y_2, \\ \forall k}} \left(-\tilde{i}_k(x_1, x_2, y_1, y_2) + \tilde{d}_k(x_1, x_2, y_1, y_2) \ln \tilde{i}_k(x_1, x_2, y_1, y_2)\right) + \text{A.T.}
\end{aligned} \tag{6.17}$$

where A.T. is the additional term that does not affect the maximization [62]. Next, the conditional expectation function, *Q*, is derived as shown in the equation below, where the complete data function is only the random variable. The additional term is omitted in the subsequent equations.

$$
\begin{aligned}
Q(o_1, o_2, h_k) &= \mathrm{E}\left[ Q(o_1, o_2, h_k) \middle| d_k(y_1, y_2) \right] \\
&= -\sum_{\substack{\forall x_1, \forall x_2, \\ \forall y_1, \forall y_2, \\ \forall k}} \tilde{i}_k(x_1, x_2, y_1, y_2) \\
&\quad + \sum_{\substack{\forall x_1, \forall x_2, \\ \forall y_1, \forall y_2, \\ \forall k}} \mathrm{E}\left[ \tilde{d}_k(x_1, x_2, y_1, y_2) \middle| d_k(y_1, y_2) \right] \ln \tilde{i}_k(x_1, x_2, y_1, y_2).
\end{aligned}
\tag{6.18}
$$

As shown in Equation (2.7) and Equation (2.63), the expectation term inside the summation from the equation above can be derived as the following.

$$
\mathrm{E}\left[ \tilde{d}_k(x_1, x_2, y_1, y_2) \middle| d_k(y_1, y_2) \right] = \frac{\tilde{i}_k(x_1, x_2, y_1, y_2)}{i_k(y_1, y_2)} d_k(y_1, y_2) \,. \tag{6.19}
$$

Substituting Equation (6.19) back into Equation (6.18) results in the following.

$$
\begin{aligned}
Q(o_1, o_2, h_k) &= -\sum_{\substack{\forall x_1, \forall x_2, \\ \forall y_1, \forall y_2, \\ \forall k}} \tilde{i}_k(x_1, x_2, y_1, y_2) \\
&\quad + \sum_{\substack{\forall x_1, \forall x_2, \\ \forall y_1, \forall y_2, \\ \forall k}} \frac{\tilde{i}_k^{old}(x_1, x_2, y_1, y_2)}{i_k^{old}(y_1, y_2)} d_k(y_1, y_2) \ln \tilde{i}_k(x_1, x_2, y_1, y_2).
\end{aligned}
\tag{6.20}
$$

Equation (6.20) is the output of the Expectation Step (E-Step) of the Expectation-Maximization (EM) algorithm. In the Maximization Step (M-Step), the conditional expectation function is maximized with respect to the 1D object functions, $o_1$ and $o_2$, and the PSF, $h_k$. The M-Step for $o_1$ is shown below.

$$
\begin{aligned}
\frac{\partial Q(o_1, o_2, h_k)}{\partial o_1(x_1)} &= -\sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} o_2^{\mathrm{new}}(x_2) h_k^{\mathrm{new}}(y_1 - x_1, y_2 - x_2) \\
&\quad + \frac{1}{o_1^{\mathrm{new}}(x_1)} \sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} \frac{o_1^{\mathrm{old}}(x_1) o_2^{\mathrm{old}}(x_2) h_k^{\mathrm{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\mathrm{old}}(y_1, y_2)} d_k(y_1, y_2).
\end{aligned}
\tag{6.21}
$$

Setting the derivation to zero results in the following update equation for $o_1$.

$$o_1^{\text{new}}(x_1) = \frac{o_1^{\text{old}}(x_1)}{\sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} o_2^{\text{new}}(x_2) h_k^{\text{new}}(y_1 - x_1, y_2 - x_2)} \sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} \frac{o_2^{\text{old}}(x_2) h_k^{\text{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\text{old}}(y_1, y_2)} d_k(y_1, y_2). \tag{6.22}$$

The first denominator on the right-hand side of Equation (6.22) seems problematic because the updated equation for $o_1$ depends on the updated $o_2$ and $h_k$. However, the nested summations of the denominator can be rearranged as shown below.

$$\sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} o_2^{\text{new}}(x_2) h_k^{\text{new}}(y_1 - x_1, y_2 - x_2) = \sum_{\forall k} \left( \sum_{\forall x_2} o_2^{\text{new}}(x_2) \left( \sum_{\forall y_1, \forall y_2} h_k^{\text{new}}(y_1 - x_1, y_2 - x_2) \right) \right). \tag{6.23}$$

Since the PSF sums up to unity, the above equation is simplified as below.

$$\sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} o_2^{\text{new}}(x_2) h_k^{\text{new}}(y_1 - x_1, y_2 - x_2) = \sum_{\forall k} \left( \sum_{\forall x_2} o_2^{\text{new}}(x_2) \cdot 1 \right) = \sum_{\forall k} \left( \sum_{\forall x_2} o_2^{\text{new}}(x_2) \right). \tag{6.24}$$

Using the constraint made in Equation (6.10) that $o_2$ sums up to one, the above equation can be further simplified as the following.

$$\sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} o_2^{\text{new}}(x_2) h_k^{\text{new}}(y_1 - x_1, y_2 - x_2) = \sum_{k=1}^{K} \left( \sum_{\forall x_2} o_2^{\text{new}}(x_2) \right) = \sum_{\forall k} 1 = K. \tag{6.25}$$

The update equation for $o_1$ can then be written as shown in the equation below.

$$o_1^{\text{new}}(x_1) = \frac{o_1^{\text{old}}(x_1)}{K} \sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} \frac{o_2^{\text{old}}(x_2) h_k^{\text{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\text{old}}(y_1, y_2)} d_k(y_1, y_2). \tag{6.26}$$

Using the same technique, the update equation for $o_2$ is obtained as shown in the equation below.

$$o_2^{\text{new}}(x_2) = \frac{o_2^{\text{old}}(x_2)}{\sum\limits_{\substack{\forall k,\forall x_1,\\ \forall y_1,\forall y_2}} o_1^{\text{new}}(x_1)h_k^{\text{new}}(y_1 - x_1, y_2 - x_2)} \sum_{\substack{\forall k,\forall x_2,\\ \forall y_1,\forall y_2}} \frac{o_1^{\text{old}}(x_1)h_k^{\text{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\text{old}}(y_1, y_2)} d_k(y_1, y_2).$$

(6.27)

The first denominator can also be simplified using the conservative nature of the PSF and the constraint made in Equation (6.11) as shown in the equation below.

$$\begin{aligned}
\sum_{\substack{\forall k,\forall x_1,\\ \forall y_1,\forall y_2}} o_1^{\text{new}}(x_1)h_k^{\text{new}}(y_1 - x_1, y_2 - x_2) &= \sum_{k=1}^{K}\left(\sum_{\forall x_1} o_1^{\text{new}}(x_1)\left(\sum_{\forall y_1,\forall y_2} h_k^{\text{new}}(y_1 - x_1, y_2 - x_2)\right)\right) \\
&= \sum_{k=1}^{K}\left(\sum_{\forall x_1} o_1^{\text{new}}(x_1)\cdot 1\right) \\
&= \sum_{k=1}^{K}\left(\sum_{\forall x_1} o_1^{\text{new}}(x_1)\right) \\
&= \sum_{k=1}^{K}\eta \\
&= \eta K.
\end{aligned} \tag{6.28}$$

The update equation for $o_1$ can then be written as shown in the equation below.

$$o_2^{\text{new}}(x_2) = \frac{o_2^{\text{old}}(x_2)}{\eta K} \sum_{\substack{\forall k,\forall x_1,\\ \forall y_1,\forall y_2}} \frac{o_1^{\text{old}}(x_1)h_k^{\text{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\text{old}}(y_1, y_2)} d_k(y_1, y_2). \tag{6.29}$$

Using the same derivation techniques, the PSF update equation is obtained as shown below.

$$h_k^{\text{new}}(x_1, x_2) = \frac{1}{U^{\text{new}}} h_k^{\text{old}}(x_1, x_2) \sum_{\forall y_1,\forall y_2} \frac{o^{\text{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\text{old}}(y_1, y_2)} d_k(y_1, y_2), \tag{6.30}$$

where $U^{\text{new}}$ is a constant to ensure that the new PSF also sums up to unity. Similar to the algorithms developed Section 2.7, the Gerchberg-Saxton phase retrieval algorithm is applied on the updated PSF in Equation (6.30) to recover the two-dimensional phase from

the intensity measurement only. The technique of spatial separation is not applied on the Gerchberg-Saxton algorithm.

### 6.2.2. Algorithm Implementation

Unlike the algorithm developed in Chapter 4, the implementation is not as straightforward. This section explains a step-by-step implementation. To do so, three temporary functions, $g_k$, $f_k$, and $u$, are introduced. Even though these functions are superfluous, the purpose is to make the implementation easy and simple to follow. A pseudo MATLAB code is also provided using these functions.

To begin, the summation of the update equation for $o_1$ from Equation (6.26) is rearranged. First, the old object function, $o^{\text{old}}$, is formed from $o_1^{\text{old}}$ and $o_2^{\text{old}}$ as shown in the equation below.

$$\begin{aligned} o_1^{\text{new}}(x_1) &= \frac{o_1^{\text{old}}(x_1)}{K} \sum_{\forall k} \sum_{\forall y_1} \sum_{\forall y_2} \sum_{\forall x_2} \frac{o_2^{\text{old}}(x_2) h_k^{\text{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\text{old}}(y_1, y_2)} d_k(y_1, y_2) \\ &= \frac{1}{K} \sum_{\forall k} \sum_{\forall x_2} \left( \left( o_1^{\text{old}}(x_1) o_2^{\text{old}}(x_2) \right) \sum_{\forall y_1} \sum_{\forall y_2} h_k^{\text{old}}(y_1 - x_1, y_2 - x_2) \frac{d_k(y_1, y_2)}{i_k^{\text{old}}(y_1, y_2)} \right) \\ &= \frac{1}{K} \sum_{\forall k} \sum_{\forall x_2} \left( o^{\text{old}}(x_1, x_2) \sum_{\forall y_1} \sum_{\forall y_2} h_k^{\text{old}}(y_1 - x_1, y_2 - x_2) \frac{d_k(y_1, y_2)}{i_k^{\text{old}}(y_1, y_2)} \right). \end{aligned} \tag{6.31}$$

Next, the first temporary function, $g_k$, is defined and Equation (6.31) can be written as shown in Equation (6.33).

$$g_k(y_1, y_2) \triangleq \frac{d_k(y_1, y_2)}{i_k^{old}(y_1, y_2)}, \tag{6.32}$$

$$o_1^{\text{new}}(x_1) = \frac{1}{K} \sum_{\forall k} \sum_{\forall x_2} \left( o^{\text{old}}(x_1, x_2) \left( \sum_{\forall y_1} \sum_{\forall y_2} h_k^{\text{old}}(y_1 - x_1, y_2 - x_2) g_k(y_1, y_2) \right) \right). \tag{6.33}$$

In the equation above, the term inside innermost parenthesis is a 2D convolution, which is defined by the second temporary function, $f_k$.

$$f_k(x_1, x_2) \triangleq \sum_{\forall y_1} \sum_{\forall y_2} h_k^{\text{old}}(y_1 - x_1, y_2 - x_2) g_k(y_1, y_2) \,. \tag{6.34}$$

The update equation for $o_1$ from Equation (6.31) can be written as the following.

$$o_1^{\text{new}}(x_1) = \frac{1}{K} \sum_{\forall k} \sum_{\forall x_2} o^{\text{old}}(x_1, x_2) f_k(x_1, x_2) \,. \tag{6.35}$$

Using the distributive property, Equation (6.35) can be written as the following.

$$o_1^{\text{new}}(x_1) = \frac{1}{K} \sum_{\forall x_2} o^{\text{old}}(x_1, x_2) \sum_{\forall k} f_k(x_1, x_2) \,. \tag{6.36}$$

Here, the last temporary function, $u$, is defined as the following.

$$u(x_1, x_2) \triangleq \sum_{\forall k} f_k(x_1, x_2) \,. \tag{6.37}$$

Therefore, the update equation for $o_1$ can be simplified as the following.

$$o_1^{\text{new}}(x_1) = \frac{1}{K} \sum_{\forall x_2} o^{\text{old}}(x_1, x_2) u(x_1, x_2) \,. \tag{6.38}$$

Similarly, the update equation for $o_2$ can also be simplified as the following.

$$o_2^{\text{new}}(x_2) = \frac{1}{\eta K} \sum_{\forall x_1} o^{\text{old}}(x_1, x_2) u(x_1, x_2) \,. \tag{6.39}$$

Next, the constraints from Equation (6.10) and Equation (6.11) are applied on $o_2$ and $o_1$ respectively. These two equations ensure that the elements of the vector $o_2$ sum up to 1 and those of the vector $o_1$ sum up to $\eta$. Next, the update equation for the PSF can be implemented by change of summation variables. It also requires creating the old 2D object function, $o^{old}$, from the outer product of $o_1^{\text{old}}$ and $o_2^{\text{old}}$, before convolving with $g_k$.

$$h_k^{\text{new}}(x_1,x_2)=\frac{1}{\eta}h_k^{\text{old}}(x_1,x_2)\sum_{\forall y_1,\forall y_2}o^{\text{old}}(y_1-x_1,y_2-x_2)\frac{d_k(y_1,y_2)}{i_k^{\text{old}}(y_1,y_2)}$$
$$=\frac{1}{U^{\text{new}}}h_k^{\text{old}}(x_1,x_2)\sum_{\forall y_1,\forall y_2}o^{\text{old}}(y_1-x_1,y_2-x_2)g_k(y_1,y_2). \quad (6.40)$$

The pseudo MATLAB implementation example for the update equations is shown in the figure below.

```
% Pseudo MATLAB Implementation Example (One Expectation-Maximization Iteration)

o_old = o1_new * o2_new;   % o1 is Nx1 vector and o2 is 1xN vector
u     = zeros(N,N);        % N is the size of the square object function
h_old = h_new;             % PSF Intensity
p_old = p_new;             % PSF Phase

for k = 1:K
  i_old(:,:,k) = real(ifft2(fft2(o1_old*o2_old).*fft2(h_old(:,:,k))));
  g(:,:,k) = d(:,:,k)./i_old(:,:,k);                                    % See Eq (6.32)
  f(:,:,k) = real(ifft2(conj(fft2(h_old(:,:,k))).*fft2(g(:,:,k))));    % See Eq (6.34)
  u        = u + f(:,:,k);                                              % See Eq (6.37)
end

o1_new = sum(o_old.*u,2)/K;                 % See Eq (6.38)
o2_new = sum(o_old.*u,1)/(eta*K);           % See Eq (6.39)

o1_new = alpha * o1_new/sum(o1_new(:));     % See Eq (6.11)
o2_new = o2_new/sum(o2_new(:));             % See Eq (6.10)

for k = 1:K
  h_new(:,:,k) = h_new(:,:,k).*real(ifft2(fft2(o_old).*fft2(g(:,:,k))));% See Eq (6.40)
  h_new(:,:,k) = h_new(:,:,k)/sum(sum(h_new(:,:,k)));        % PSF is conservative
  p_new(:,:,k) = GerchbergSaxton(h_new(:,:,k),p_old(:,:,k)); % See Section 2.7
end
```

**Figure 6.1. Pseudo MATLAB Implementation for Spatial Separation Approach.**

Next, Scenario 1 from Table 3.4 is used for validating the algorithm. In this scenario, the bright object has a photoelectron count of 10,000 and the dim object has 1,000. The background noise with an average photoelectron count of 10 is added to all pixels. For the 1DBD algorithm, $o_1$ and $o_2$ are initially assigned to as constant vector, whose elements add up to $\eta$ and 1 respectively. The PSFs are initialized as the diffraction-limited PSFs, where the phases are planes of zeros. The performance of the 1DBD

algorithm is compared with that of the MFBD algorithm, using the two performance metrics described in Section 3.5. The MFBD algorithm is based on Schulz's approach explained in Section 2.6.4. For the first performance metric using the Receiver Operating Characteristic (ROC) function, it can be seen that the current implementation of the 1DBD algorithm does not perform as well as the MFBD algorithm for the dim object detection, when the false alarm rate is very low as shown in Figure 6.2. For the second metric, the average of the reconstructed object functions using the MFBD also outperforms that using the 1DBD as shown in Figure 6.3.

Since the performance of the 1DBD algorithm does not exceed that of the MFBD algorithm, additional scenarios are not processed. In the next section, the Dimension Reduction Blind Deconvolution algorithm (DRBD) is introduced, where improvements are made for dim object detection. In addition, the algorithm also allows when the object function can no longer be represented by the outer product of two 1D vectors.

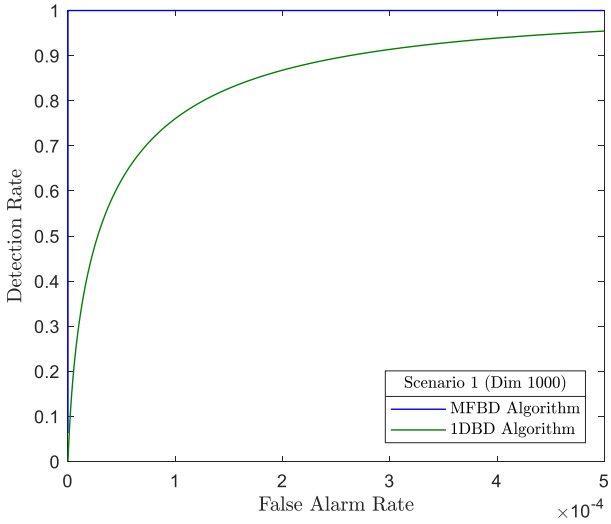


**Figure 6.2. The ROC Comparison for Scenario 1 between the MFBD Algorithm and the 1DBD Algorithm for Very Low False Alarm Rate.**

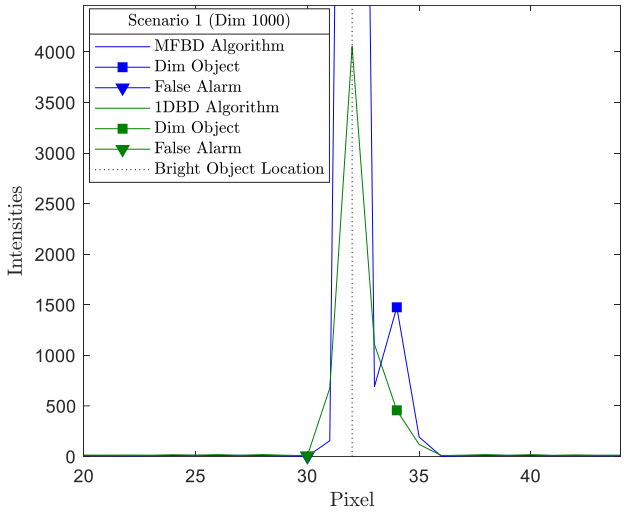


**Figure 6.3. The Average Function Comparison for Scenario 1 between the MFBD Algorithm and the 1DBD Algorithm.**

## 6.3. Dimension Reduction Blind Deconvolution

This section introduces the DRBD algorithm with two improvements made to the 1DBD. In Section 6.3.1, the emphasis is on improving the detection of the dim object. In Section 6.3.2, the emphasis is on applying the algorithm when the object function is no longer spatially separable and cannot be represented by the outer product between two 1D vectors, meaning the rank of the object function is higher than one.

### 6.3.1. Spatially Separable Object Function

To understand the reason 1DBD algorithm does not outperform the MFBD algorithm, the averages of reconstructed $o_1$ and $o_2$ from Section 6.2 are examined. From Figure 6.4 (a), it can be seen that the reconstructed $o_1$ background level is over twice of the true value, while the peak value is at 7,212 instead of the true value of 10,000. One

explanation is that the gap between the true value of 10,000 and the calculated peak value of 7,212 is spread over the remaining pixels, causing the background level to rise.

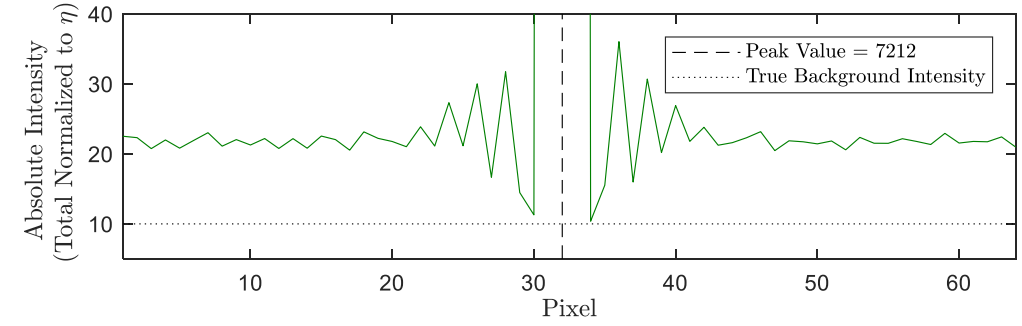


**Fig. (a) Averaged 1D Object Function ($o_1$)**

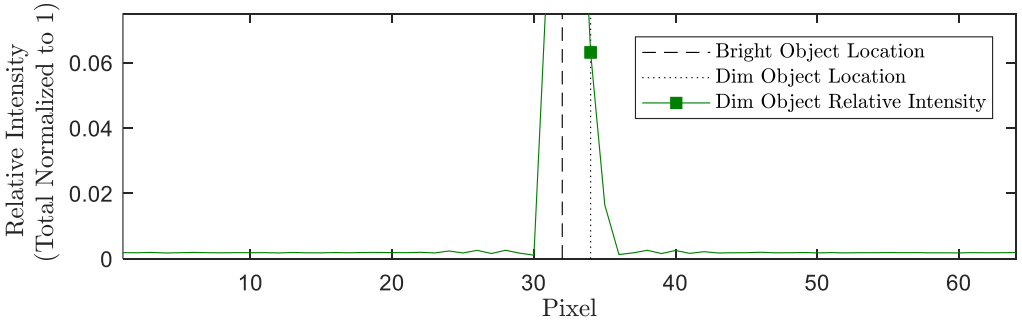


**Fig. (b) Averaged 1D Object Function ($o_2$)**

**Figure 6.4. The Averages of the 1D Vectors Processed with 1DBD Algorithm.**

One technique to address is by separating the background level, *B*, from the 1D vectors shown in the equation below.

$$o(x_1, x_2) = o_1(x_1)o_2(x_2) + B. \tag{6.41}$$

The background level can be estimated from the regions in the data functions that are far away from the bright object and the dim object and that contain no known objects, as shown in Figure 6.5.

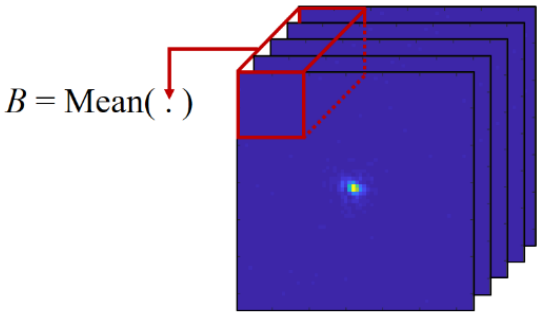


**Figure 6.5. Background Level Estimation.**

Using similar derivation, the update functions can be derived as the followings.

$$o_1^{\text{new}}(x_1) = \frac{1}{K} \sum_{\substack{\forall k, \forall x_2, \\ \forall y_1, \forall y_2}} \frac{\left(o_1^{\text{old}}(x_1) o_2^{\text{old}}(x_2) + B\right) h_k^{\text{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\text{old}}(y_1, y_2)} d_k(y_1, y_2), \tag{6.42}$$

$$o_2^{\text{new}}(x_2) = \frac{1}{\eta K} \sum_{\substack{\forall k, \forall x_1, \\ \forall y_1, \forall y_2}} \frac{\left(o_1^{\text{old}}(x_1) o_2^{\text{old}}(x_2) + B\right) h_k^{\text{old}}(y_1 - x_1, y_2 - x_2)}{i_k^{\text{old}}(y_1, y_2)} d_k(y_1, y_2). \tag{6.43}$$

The update equation for the PSF remains the same as shown in Equation (6.41). Similar to the implementation shown in Section 6.2.2, the two update equations can be simplified as shown in the equation below.

$$o_1^{\text{new}}(x_1) = \frac{1}{K} \sum_{\forall x_2} \left(o_1^{\text{old}}(x_1) o_2^{\text{old}}(x_2) + B\right) u(x_1, x_2), \tag{6.44}$$

$$o_2^{\text{new}}(x_2) = \frac{1}{\eta K} \sum_{\forall x_1} \left(o_1^{\text{old}}(x_1) o_2^{\text{old}}(x_2) + B\right) u(x_1, x_2). \tag{6.45}$$

From Equation (6.32), Equation (6.34), and Equation (6.37), $u(x_1, x_2)$ is defined as shown below.

$$u(x_1, x_2) = \sum_{\forall k} \left( \sum_{\forall y_1} \sum_{\forall y_2} h_k^{\text{old}}(y_1 - x_1, y_2 - x_2) \frac{d_k(y_1, y_2)}{i_k^{\text{old}}(y_1, y_2)} \right). \tag{6.46}$$

### 6.3.2. Non-Spatially Separable Object Function

A non-spatially separable object function is defined as one that cannot be represented by the outer product of two 1D vectors as shown in the figure below, where the rank of the object function is higher than one.

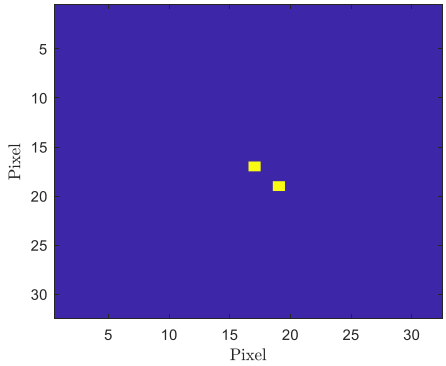


**Figure 6.6. Non-Spatially Separable Object Function**

If the object function from Figure 6.6 were to be represented by an outer product of two metrics, it would need to be an outer product between an $N \times 2$ matrix, $\mathbf{o}_1^{N\times 2}$, and a $2 \times N$ matrix, $\mathbf{o}_2^{2\times N}$, as shown in Equation (6.47). Similar to the DRBD algorithm in Section 6.3.1, the background level, *B*, is added as a separate variable.

$$\mathbf{o} = \left( \mathbf{o}_1^{N\times 2} \otimes \mathbf{o}_2^{2\times N} \right) + B \tag{6.47}$$

Similar to Equation (6.10) and Equation (6.11), $\mathbf{o}_1^{N\times 2}$ is constrained to sum up to the total intensity, $\eta$, and $\mathbf{o}_2^{2\times N}$ is constrained to sum up to unity as shown below.

$$\sum_{\forall x_1}\sum_{\upsilon=1}^{2} o_1(x_1,\upsilon)=\eta, \tag{6.48}$$

$$\sum_{\forall x_2}\sum_{\upsilon=1}^{2} o_2(\upsilon,x_2)=1. \tag{6.49}$$

The derivation is similar those shown in Sections 6.2.1 and 6.3.1. Instead of summing along $x_1$ and $x_2$ as shown in Equation (6.44) and Equation (6.45), a singular value decomposition (SVD) is performed on the term inside the summation [101]–[103].

$$\mathbf{USV}^{\mathbf{T}}=\left(\left(\mathbf{o}_1^{N\times 2^{\text{old}}}\otimes\mathbf{o}_2^{2\times N^{\text{old}}}\right)+B\right)\odot\boldsymbol{\mathcal{u}}, \tag{6.50}$$

where $\mathbf{U}$ is an $N\times M$ matrix, $\mathbf{S}$ is an $M\times M$ diagonal matrix, and $\mathbf{V}^{\mathbf{T}}$ is an $M\times N$ matrix since the assumption is made that the object function is a square $N\times N$ matrix. In addition, $\odot$ is the piecewise matrix multiplication and $\boldsymbol{\mathcal{u}}$ is the $N\times N$ matrix whose elements are $u(x_1,x_2)$ from Equation (6.46) as shown below.

$$\boldsymbol{\mathcal{u}}\triangleq\begin{bmatrix} u(1,1) & \cdots & u(1,N)\\ \vdots & \ddots & \vdots\\ u(N,1) & \cdots & u(N,N)\end{bmatrix}. \tag{6.51}$$

Even though the number of resultant singular values may exceed two $(M>2)$, only the largest two singular values are used to match the dimensions of $\mathbf{o}_1^{N\times 2}$ and $\mathbf{o}_2^{2\times N}$. Therefore, the update equations become the following.

$$\mathbf{o}_1^{N\times 2^{\text{new}}}=\frac{1}{\eta_1}\mathbf{U}^{N\times 2}, \tag{6.52}$$

$$\mathbf{o}_2^{2\times N^{\text{new}}}=\frac{1}{\eta_2}\mathbf{S}^{2\times 2}\left(\mathbf{V}^{N\times 2}\right)^{\mathbf{T}}, \tag{6.53}$$

where $\eta_1$ and $\eta_1$ are some constants to ensure that $\mathbf{0}_1^{N\times 2^{new}}$ and $\mathbf{0}_2^{2\times N^{new}}$ sum up to $\eta$ and 1 respectively as shown in Equation (6.48) and Equation (6.49). $\mathbf{S}^{2\times 2}$ is a diagonal matrix made up with the two largest singular values from $\mathbf{S}$. $\mathbf{U}^{N\times 2}$ and $\left(\mathbf{V}^{N\times 2}\right)^{\mathbf{T}}$ are made up of two columns and two rows that are associated with the singular values in $\mathbf{S}^{2\times 2}$. The PSF update equation remains the same as the one shown in Equation (6.40). A pseudo MATLAB implementation is shown in the figure below.

```
% Pseudo MATLAB Implementation Example for SVD )

[U, S, V] = svd( (o1_old * o2_old) .* U );     % See Eq (6.50)

o1_new = U(:,1:2) / eta1;                      % See Eq (6.52)
o2_new = S(1:2,1:2) * (V(:,1:2)') / eta2;      % See Eq (6.53)
```

**Figure 6.7. Pseudo MATLAB Implementation for SVD Approach.**

## 6.4. Performance Comparison

In this section, the results are shown for spatially separable object functions and non-spatially separable object functions. The data used for spatially separable object function is the same as that used in Chapter 3 and Chapter 4. However, instead of running the algorithm through all eight scenarios from Table 3.4, this section shows the results for only two scenarios, which are Scenario 1 and Scenario 8. The intensity of the bright object is set to 10,000 photoelectrons for both scenarios, but that for the dim object is set to 1,000 photoelectrons for Scenario 1 and 25 for Scenario 8. The reason for selecting Scenario 1 is to make sure that the new algorithm does not suffer performance degradation for a trivial

case. The reason for selecting Scenario 8 is to compare the performance for the most challenging case.

For all the image processing completed in this section, a total of 500 EM iterations are completed. For each EM iteration, a total of 100 Gerchberg-Saxton phase retrieval are completed. Instead of using a Gaussian mask approach, the intensities at the false alarm and the dim object locations are recorded. The intensities of the false alarm pixel and that of the dim object intensities are compared using the ROC function. The average information of all estimated object functions are also shown.

For all initializations, $o_1$ and $o_2$ are initialized with as constant vectors or constant matrices while satisfying their constraints of summing up to $\eta$ and 1 respectively. The PSFs are initialized as the diffraction-limited PSFs where the phases are initialized with planes of zeros.

### 6.4.1. Spatially Separable Object Function

This section presents the performance result of Scenario 1 and Scenario 8 using the DRBD algorithm. Because the object function is spatially separable in these scenarios, two approaches of the DRBD algorithm are used. The first one is the summation approach as shown in Section 6.3.1. The second one is the SVD approach as shown in Section 6.3.2. However, $o_1$ and $o_2$ are decomposed as $N \times 1$ and $1 \times N$ vectors instead of $N \times 2$ and $2 \times N$ vectors.

For Scenario 1, the ROC performances of the DRBD algorithm using the summation approach and the SVD approach are shown are shown in Figure 6.8 and Figure 6.9. From the results, it can be seen that even for the very low false alarm rate $(1 \times 10^{-10})$,

the DRBD gives perfect detection regardless of whether the summation approach or the SVD approach is used. The average functions of the DRBD algorithm using two approaches are shown in Figure 6.10 and Figure 6.11. From the results, the averaged reconstructed intensities of the bright object and that of the dim object closely match their true values.

For Scenario 8, the ROC performances of the DRBD algorithm using the summation approach and the SVD approach are shown are shown in Figure 6.12 and Figure 6.13 respectively. Because these intensities of the false alarm and the dim object do not follow Gamma, Gaussian, Weibull, or any known distributions, the ROC functions are not fitted to a known distribution. Appendix D shows how fitting these intensities to Gamma or Weibull could lead to inaccurate detection rates.

As a reference, the ROC performance of the DRBD algorithm is compared with that of the NSBD algorithm and the MFBD algorithm from Figure 4.20. From the results, the DRBD algorithm outperforms the NSBD algorithm and the MFBD algorithm for both the summation approach and the SVD approach. The average function of the DRBD algorithm using the two approaches are shown in Figure 6.14 and Figure 6.15. From the results, even though the averaged dim object intensity is higher than that of the false alarm pixel, they are very close to the background level. Therefore, the dim object is discernable from the false alarm using the DRBD algorithm, even though the estimated intensity of the dim object is significantly less than that of the true intensity.

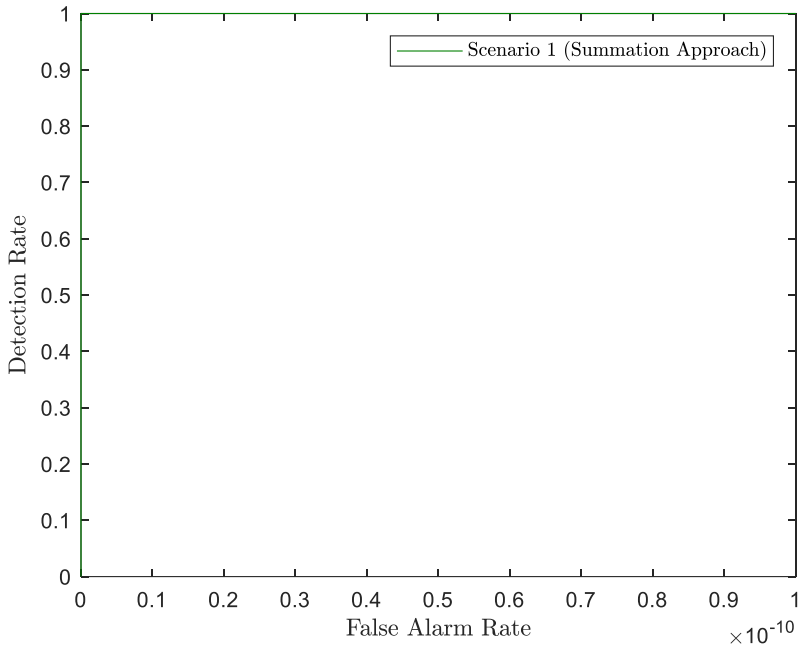


**Figure 6.8. The ROC Performance for Scenario 1 Using the DRBD Algorithm with the Summation Approach for Very Low False Alarm Rate.**

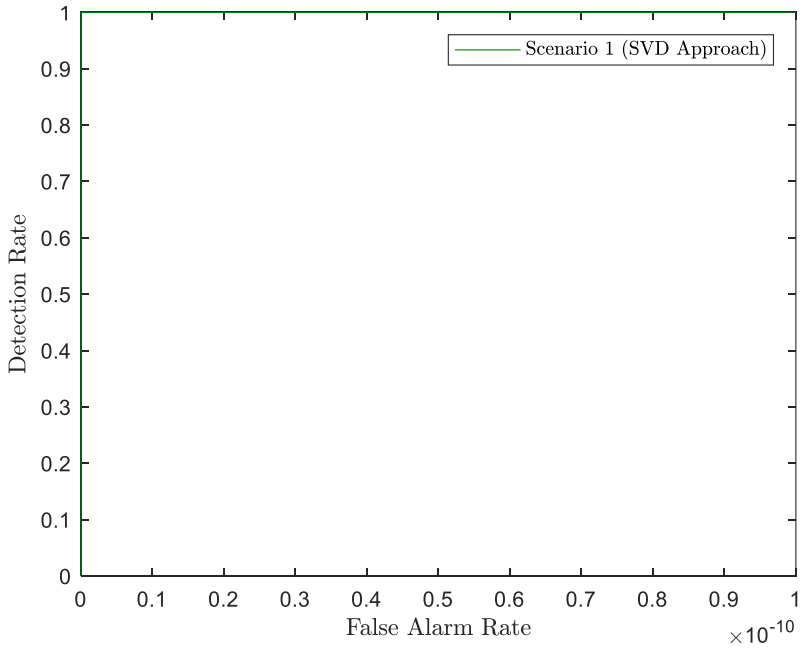


**Figure 6.9. The ROC Performance for Scenario 1 Using the DRBD Algorithm with the SVD Approach for Very Low False Alarm Rate.**

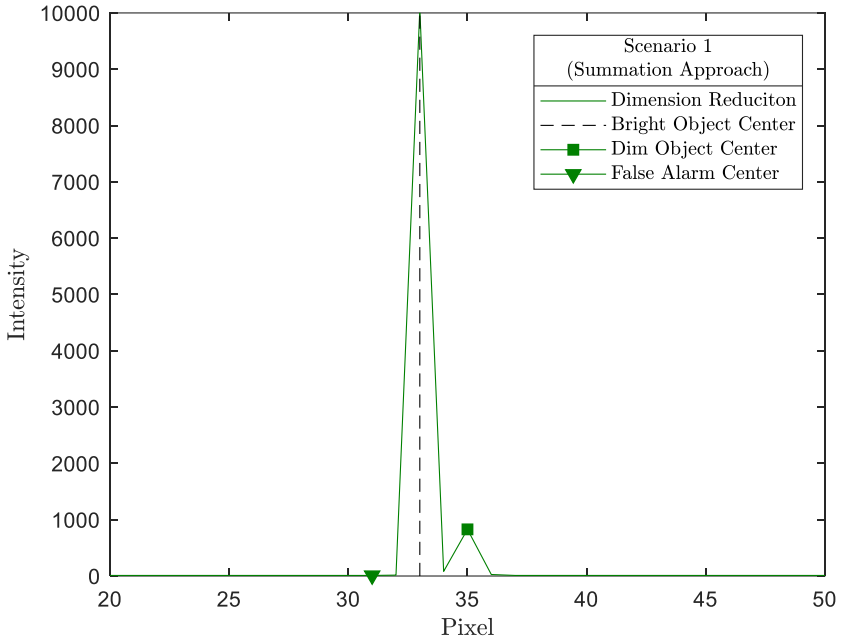


**Figure 6.10. The Average Function for Scenario 1 Using the DRBD Algorithm with the Summation Approach.**

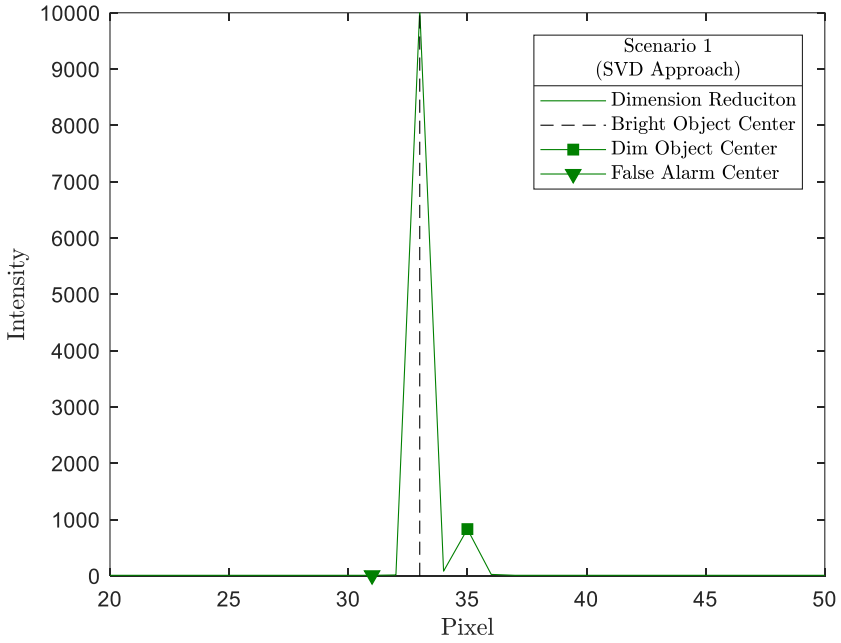


**Figure 6.11. The Average Function for Scenario 1 Using the DRBD Algorithm with the SVD Approach.**

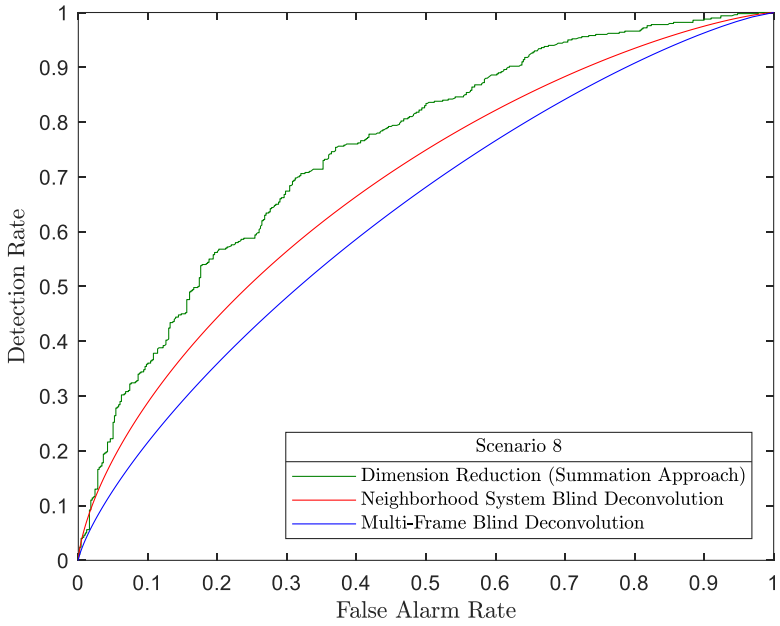


**Figure 6.12. The ROC Comparison for Scenario 8 Using the DRBD Algorithm with the Summation Approach.**

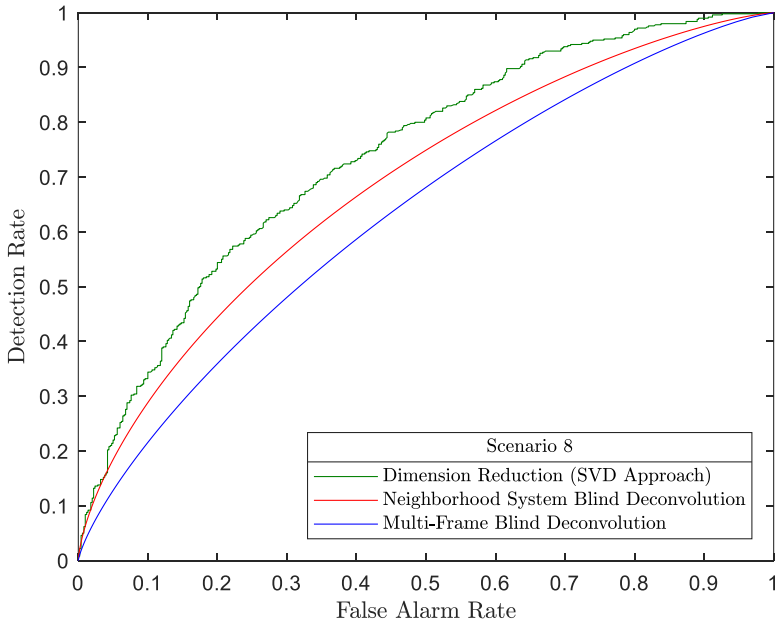


**Figure 6.13. The ROC Comparison for Scenario 8 Using the DRBD Algorithm with the SVD Approach.**

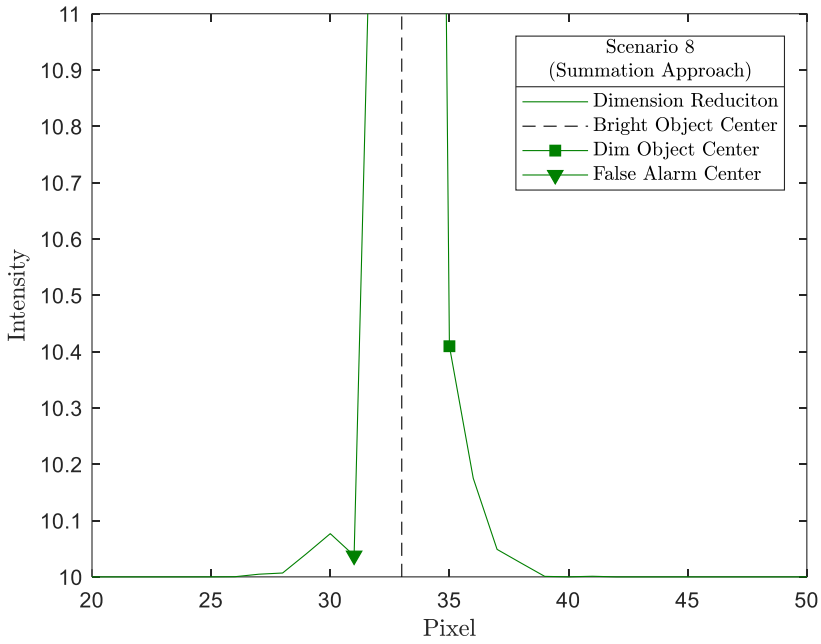


**Figure 6.14. The Average Function for Scenario 8 Using the DRBD Algorithm with the Summation Approach.**

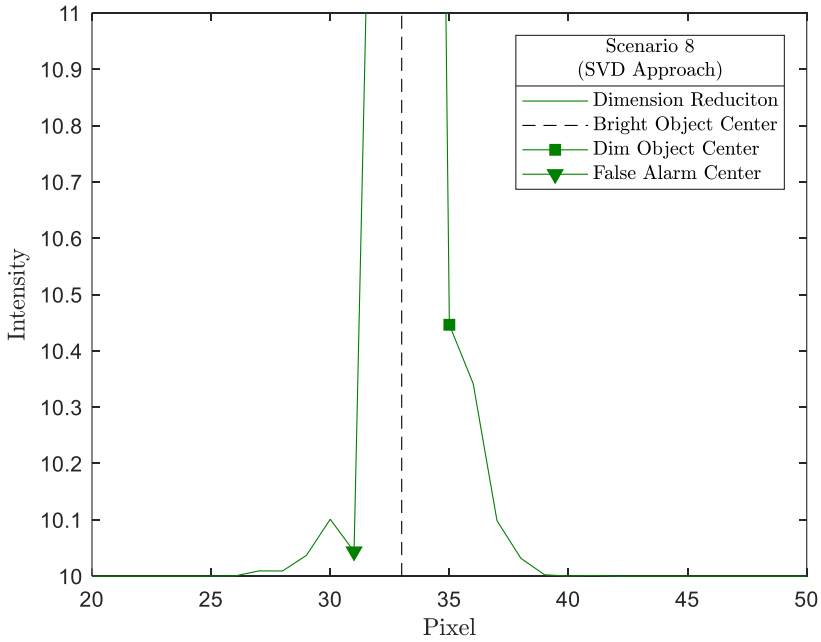


**Figure 6.15. The Average Function for Scenario 8 Using the DRBD Algorithm with the SVD Approach.**

### 6.4.2. Non-Spatially Separable Object Function

For non-spatially separable object function, two new scenarios are generated where the bright object is place in the center and the dim object is placed two pixels below and two pixels to the right of the bright object at a $\sqrt{2}$ pixel separation as shown in the figure below. Even though the bright object is placed at (32, 32) for the scenarios in Chapter 3 and Chapter 4, all performance comparisons in this research are based on the relative positions between the bright object and the dim object instead of their absolute positions.

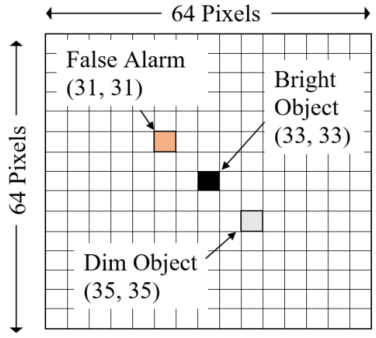


**Figure 6.16. Non-Spatially Separable Object Function with Bright Object, Dim Object, and False Alarm Pixel Locations.**

To distinguish from the two new scenarios described in Table 3.4, the new ones are annotated with "N". For Scenario 1N and Scenario 8N, the intensities of the bright object, the dim object, and the background level are kept the same as those in Scenario 1 and Scenario 8. The only difference is the placement of the dim object relative to the bright object. The data functions are also generated using the same atmospheric turbulence strength of $D/r_0 = 2$, using the Zernike technique with 100 polynomials as described in Section 3.3.7. A total of 2,500 data functions are generated. A set of 5 data functions are grouped to form one dataset, generating a total of 500 datasets per scenario.

Similar to Section 6.4.1, the ROC function and the average function are used for comparing the performance of the DRBD algorithm. Because the object in this section are not spatially separable, only the SVD approach could be used where $o_1$ and $o_2$ are decomposed $N \times 2$ and $2 \times N$ vectors. For the average function, the intensities of the pixels on the diagonal line is recorded where the bright object, the dim object, and the false alarm are located as shown in the figure below.

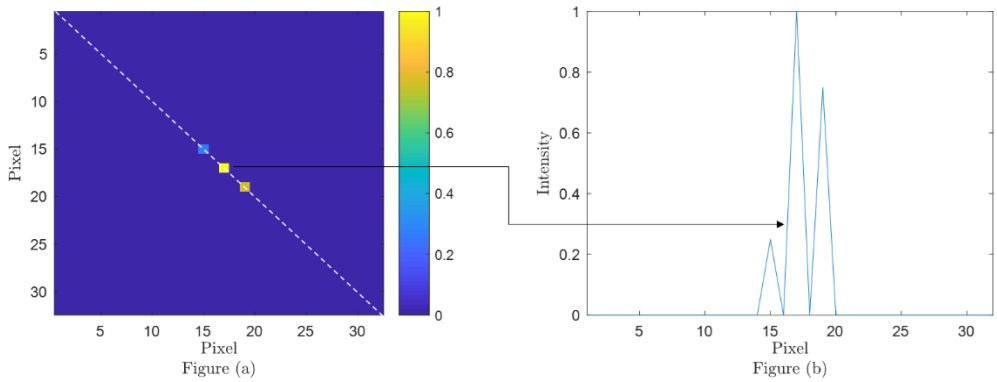


**Figure 6.17. Average Function Used for Non-Spatially Separable Object Function.**

The results for Scenario 1N and Scenario 8N are shown in Figure 6.18 through Figure 6.21 using the ROC function and the average function. For Scenario 8N, the ROC functions are fitted with Weibull distribution because the dim object intensities and the false alarm intensities follow such distribution as shown in Figure D-9 in Appendix D [87], [104]. The ROC functions are shown along with that of the NSBD algorithm and the MFBD algorithm from Figure 4.20 for reference. This is not a comparison because Figure 4.20 shows the results from Scenario 8, which is different from Scenario 8N.

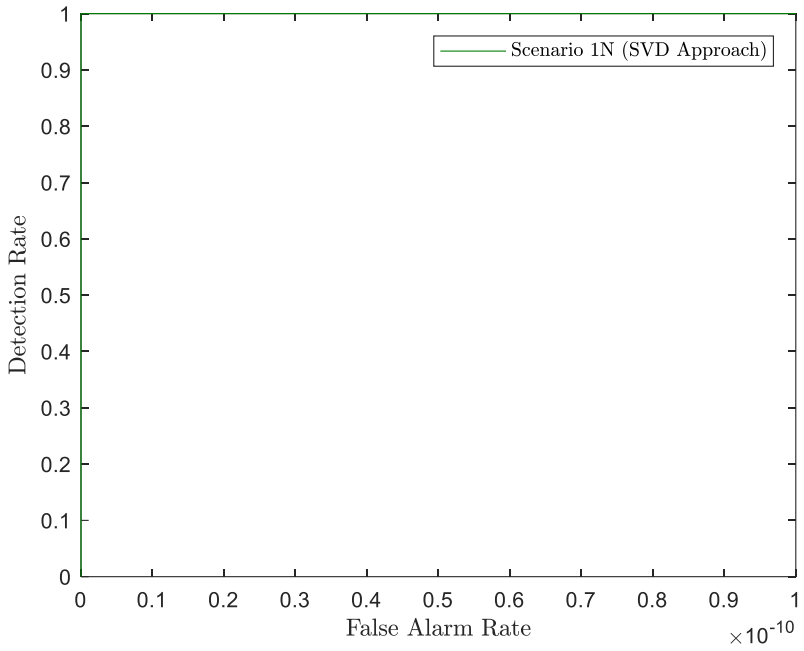


**Figure 6.18. The ROC Performance for Scenario 1N Using the DRBD Algorithm with SVD Approach for Very Low False Alarm Rate.**

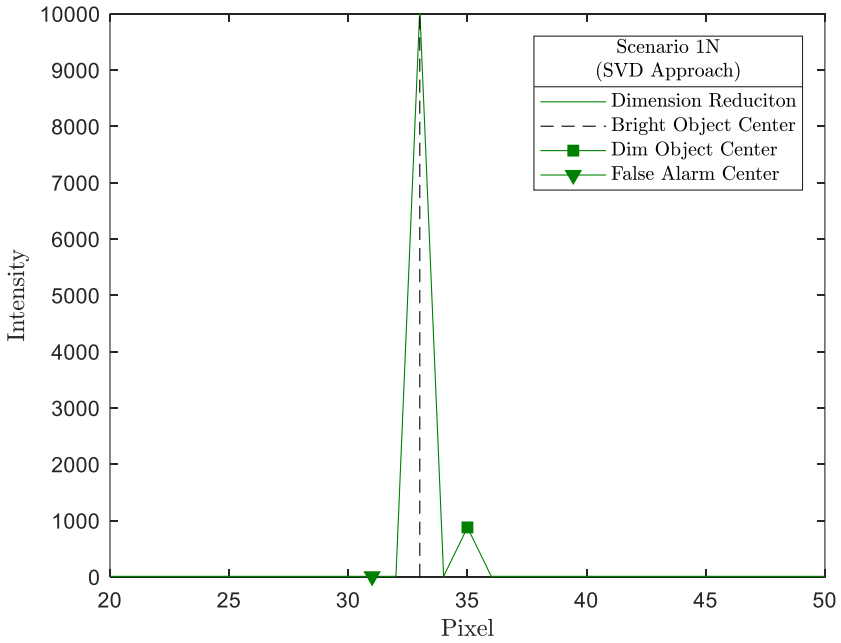


**Figure 6.19. The Average Function for Scenario 1N Using the DRBD Algorithm with SVD Approach.**

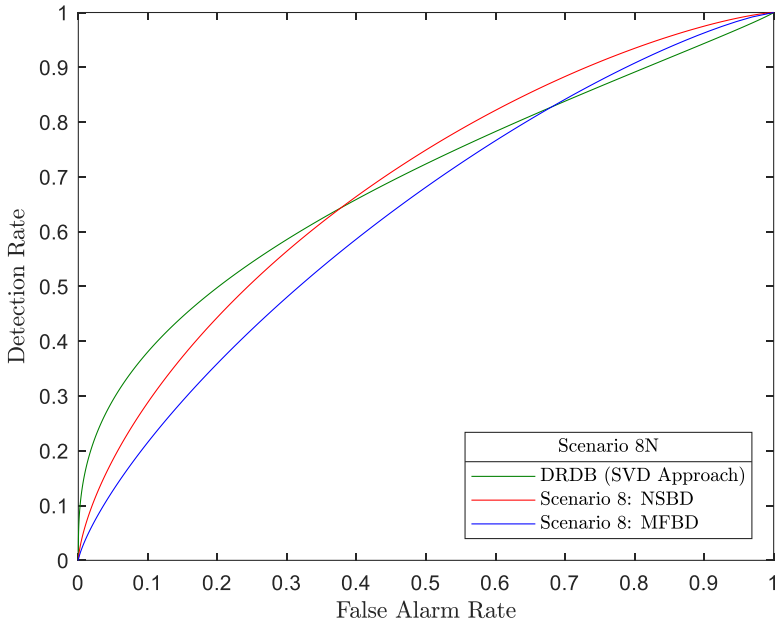


**Figure 6.20. The ROC Comparison for Scenario 8N Using the DRBD Algorithm with the SVD Approach.**

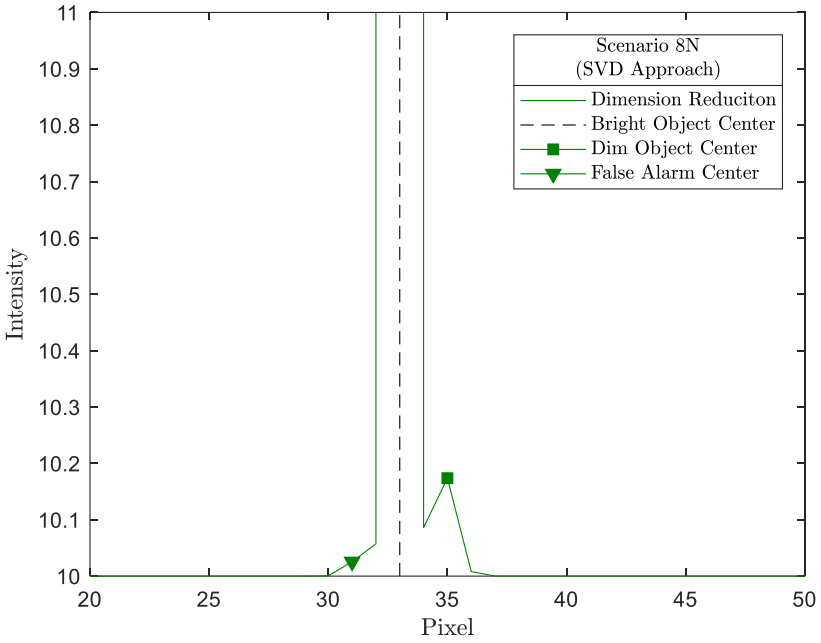


**Figure 6.21. The Average Function for Scenario 8N Using the DRBD Algorithm with SVD Approach.**

For Scenario 1N, the ROC function and the average function from the DRBD algorithm are shown in Figure 6.18 and Figure 6.19. The DRBD algorithm gives a perfect detection even for very low false alarm rate $(1\times10^{-10})$. In addition, the average reconstructed intensities of the bright object and the dim object match their true values.

For Scenario 8N, the ROC functions and the average functions from the DRBD algorithm are shown in Figure 6.20 and Figure 6.21. From the results, the DRBD algorithm provides a better detection technique over the NSDB algorithm and the MFBD algorithm for low false alarm rates. For the average function, the dim object is discernable from the false alarm with the DRBD algorithm, even though the estimated intensity of the dim object is significantly less than that of the true intensity.

## 6.5. Summary

In summary, this chapter provided a new Dimension Reduction Blind Deconvolution algorithm that was based on the One-Dimensional Multi-Frame Blind Deconvolution algorithm. The latter requires the object function to be spatially separable, meaning that it needs to be represented by the outer product of two one-dimensional vectors. In the Dimension Reduction Blind Deconvolution algorithm, two improvements were made. The first one was the improved ability to detect dim objects in close proximity to a bright one. The second one was the ability to perform image processing when the object function was no longer spatially separable. Two existing scenarios were used for testing the first improvement where the object function was spatially separable. Two new scenarios were generated for testing the second improvement where the object function is

no longer spatially separable. The Receiver Operating Characteristic function and the average function were used to compare the performance. The DRBD algorithm provides an improved technique to detect the dim object over the Multi-Frame Blind Deconvolution algorithm and the One-Dimensional Multi-Frame Blind Deconvolution Algorithm.

# 7. Conclusion

## 7.1. Chapter Overview

This chapter summarizes the purpose that this research is set out to address, which is to improve the closely spaced dim object detection. It provides a summary of the algorithms developed to support the three tasks in fulfilling the purpose. The potential future research is also explained. In addition, the academic contributions are also listed.

## 7.2. Research Purpose

The overall purpose of this research to improve the detection of dim stellar objects that are in close proximity to a bright one through statistical imaging without the need to invest in additional hardware infrastructure. To fulfill the purpose, this research accomplished three tasks.

1. Develop algorithm(s) to improve the ability of detecting dim stellar objects using short exposure images.
2. Validate the algorithm(s) through computer generated data.
3. Validate the algorithm(s) on the telescope systems within the Space Surveillance Network (SSN) or the academic and scientific communities.

Due to unavailability of the SSN telescope data, the data collected in the Optics Laboratory at the Air Force Institute of Technology is used instead.

## 7.3. Work Completed

This section explains the work completed for each task in support of improving dim object detection.

***Task 1: Develop algorithm(s) to improve the ability of detecting dim stellar objects using short exposure images.***

In Chapter 4, the Neighborhood System Blind Deconvolution (NSBD) Algorithm is developed. This algorithm separates the data into three functions, which are the primary bright object function, the neighborhood system function around the primary object, and the background function. In this chapter, it is assumed that the data functions are sampled at Nyquist rate such that there are no undersampling effects. In Chapter 5, the NSBD algorithm is modified so that it can perform the statistical image processing when the data functions are undersampled.

In Chapter 6, the Dimension Reduction Blind Deconvolution (DRBD) algorithm is developed. This algorithm is based on the One-Dimensional Multi-Frame Blind Deconvolution [99], which requires that the object function is spatially separable in a sense that it can be represented by the outer product of two one-dimensional (1D) vectors. The DRBD algorithm makes two improvements. The first one is in improving the ability to detect dim objects that are in close proximity to the bright one. The second one allows the DRBD algorithm to perform statistical image processing on the object functions that can no longer be represented by the outer product of two 1D vectors.

***Task 2: Validate the algorithm(s) through computer generated data.***

In Chapter 4, the performance of the NSBD algorithm is validated with computer generated data. Eight scenarios are generated. For all scenarios, the intensity of the bright object is fixed at 10,000 photoelectrons. The intensities of the dim object are varied from 10% to 0.25% of the intensity of the bright object as described in Table 3.4. A constant background level of 10 photoelectrons is added to all pixels including the bright object and the dim object. A Zernike phase screen generation technique is used in creating random atmospheric phase screens, using 100 polynomials. The strength of the atmospheric turbulence, $D / r_0$, is set to two for all phase screens. For each scenario, a total of 500 datasets are generated with each dataset containing five data functions for multi-frame statistical image processing. The phase screens are generated so that no two screens are statistically correlated. The performance of the NSBD algorithm is compared with the Multi-Frame Blind Deconvolution (MFBD) implementation by Schulz [62], since the MFBD algorithm is used by many scholars in the electro-optics community for performance comparison. Two performance metrics descried in Section 3.5 are used for performance comparison. The first one is the Receiver Operating Characteristics (ROC), which compares how well the distribution of the dim object intensities is separated from that of the false alarm pixel, where no object intensities exist. The second one is the average intensity of the reconstructed dim object. From the results, the NSBD algorithm provides an improved technique over the MFBD algorithm for the dim object detection.

In Chapter 6, a set of data functions are generated to simulate the laboratory collected data. This data presents the undersampling case. The readout noise is also

introduced to simulate the outputs of the charge-coupled devices. In addition, the atmospheric strength is also higher at $D/r_0 = 10$. The NSBD algorithm is modified for the undersampled data. The results are compared to the MFBD algorithm that is also modified for the undersampled data. The results are presented in two cases. In the first case, the NSBD algorithm does not assume any knowledge about the bright object, presenting a true blind deconvolution processing. In the second case, the shape for the primary object is given to the NSBD algorithm. From the results, the NSBD algorithm provides an improved technique over the MFBD algorithm. In addition, the NSBD performance is further improved when the shape of the primary object is given for the second case.

In Chapter 6, to validate the performance of the DRBD algorithm, Scenario 1 and Scenario 8 from Table 3.4 are used for the cases when the object function is spatially separable. For these two scenarios, the DRBD algorithm outperforms the MFBD algorithm. Two new scenarios, Scenario 1N and Scenario 8N, are created where the object function is no longer spatially separable. The DRBD algorithm also outperforms the MFBD algorithm for these scenarios.

***Task 3: Validate the algorithm(s) on the telescope systems within the SSN or the academic and scientific communities.***

In Chapter 5, the performance of the NSBD algorithm is compared using the data collected at the Optics Laboratory at AFIT. This data is undersampled by a factor of approximately two. Therefore, the data is processed with the modified NSBD algorithm

for undersampled data. The performance is compared with that of the MFBD algorithm modified for the undersampled data. Similar to the computer simulated data mentioned in Task 2, two cases are analyzed. In the first case, the NSBD algorithm does not assume any knowledge about the bright object. In the second case, the shape information of the bright object is given. The NSBD algorithm outperforms the MFBD algorithm for both cases. In addition, when the shape information is given, the NSBD algorithm provides additional performance gain over the case where no prior information is given. The performance of the laboratory data is consistent that that of the computer generated data that mimics the former. The list of the completed work for all three tasks is summarized in the table below.

**Table 7.1. Summary of Work Completed.**

| **Task 1: Algorithms Developed** | | | **Task 2: Validation with Computer Generated Data** | **Task 3: Validation with Laboratory Data** |
|---|---|---|---|---|
| **Algorithm Name** | **Nyquist Sampled** | **Undersampled** | | |
| NSBD | Chapter 4 | Chapter 5 | Chapter 4<br>Chapter 5 | Chapter 5 |
| DRBD | Chapter 6 | - | Chapter 6 | - |

## 7.4. Future Work

There are several opportunities for continued research based on the work presented in this dissertation.

***Sub-Pixel Blind Deconvolution***

In this research, the bright object and the dim object are located in different pixels even for the undersampled case. For the sub-pixel undersampled case, the undersampling factor is high enough that a pixel from data function contains both the bright object and the dim object. This dissertation provides the undersampled blind deconvolution technique to reconstruct in the Nyquist space. The NSBD algorithm provides an image processing technique where the bright object is processed separately from its neighborhood system. Building on these foundations, the sub-pixel blind deconvolution algorithm needs to be further investigated.

***Undersampled Dimension Reduction Blind Deconvolution***

Another area for further investigation is the undersampled DRBD algorithm, especially when the readout noise is introduced. This dissertation provides the undersampled blind deconvolution derivation as well as the dimension reduction approach. A combination of these two provides the foundation to be further investigated for the undersampled DRBD algorithm and to process the undersampled laboratory data.

## 7.5. Publications

To ensure the originality and significant contribution of the research in this dissertation, each portion of this research is submitted and published as either a conference or journal paper for evaluation and critic from experts within the field.

**Conference Presentation**

- Based on Chapter 4:

  R. M. Aung and S. C. Cain, "Multi-Frame Blind Deconvolution of Closely Spaced Dim Stellar Objects," in *Advanced Maui Optical and Space Surveillance Technologies Conference,* 2019.

**Journal**

- Based on Chapter 5:

  R. M. Aung and S. C. Cain, "Improving closely spaced dim object detection through multi-frame blind deconvolution of near stellar neighbourhoods," *Journal of Modern Optics*. Submitted May 2020. Accepted August 2020.

## Appendix A: Estimation of Gamma Distribution

The estimation methods for the gamma distribution parameters are adopted from Hahn and Shapiro [105]. Let $\{i_n, n=1,...,N\}$ be the measured intensities that need to be fitted to the gamma probability distribution function. The function takes on the following form, where *a* and *b* are the gamma distribution parameters and *i* the continuous random variable.

$$f(i;a,b)=\frac{b^a}{\Gamma(a)}i^{a-1}e^{-bi}, \tag{A.1}$$

$$\Gamma(a)=\int_0^\infty i^{a-1}e^{-a}di\,. \tag{A.2}$$

The estimation of the gamma distribution parameters, $\hat{a}$ and $\hat{b}$, are given by the following equations.

$$\hat{b}=\frac{(N-1)\bar{i}}{\sum_n^N\left(i_n-\bar{i}\right)^2}, \tag{A.3}$$

$$\hat{a}=\hat{b}\bar{i}\,, \tag{A.4}$$

$$\bar{i}=\frac{1}{N}\sum_{n=1}^{N}i_n\,. \tag{A.5}$$

When using MATLAB, it is important to note that it defines *b* slightly different in the following manner.

$$b_{MATLAB}=\frac{1}{b}. \tag{A.6}$$

# Appendix B: Gamma Distribution Fits from Chapter 4

This appendix shows the gamma distribution fits for the intensities of the dim object and the false alarm pixel for both the coupled and decoupled approaches using the Neighborhood System Blind Deconvolution algorithm developed in Section 4.4.1.

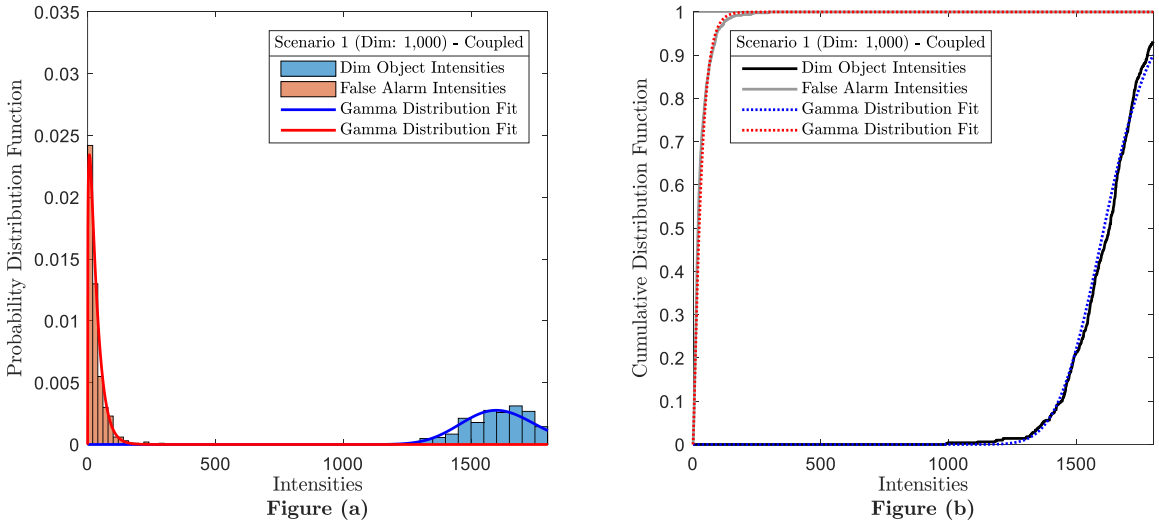


**Figure B-1. The Gamma Fits for the Coupled Approach for Scenario 1.**

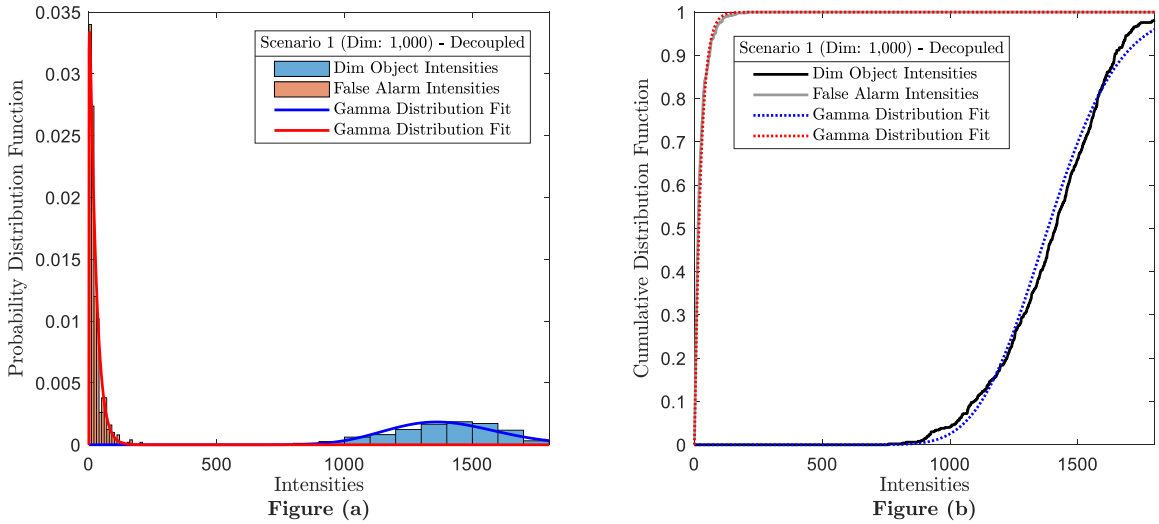


**Figure B-2. The Gamma Fits for the Decoupled Approach for Scenario 1.**

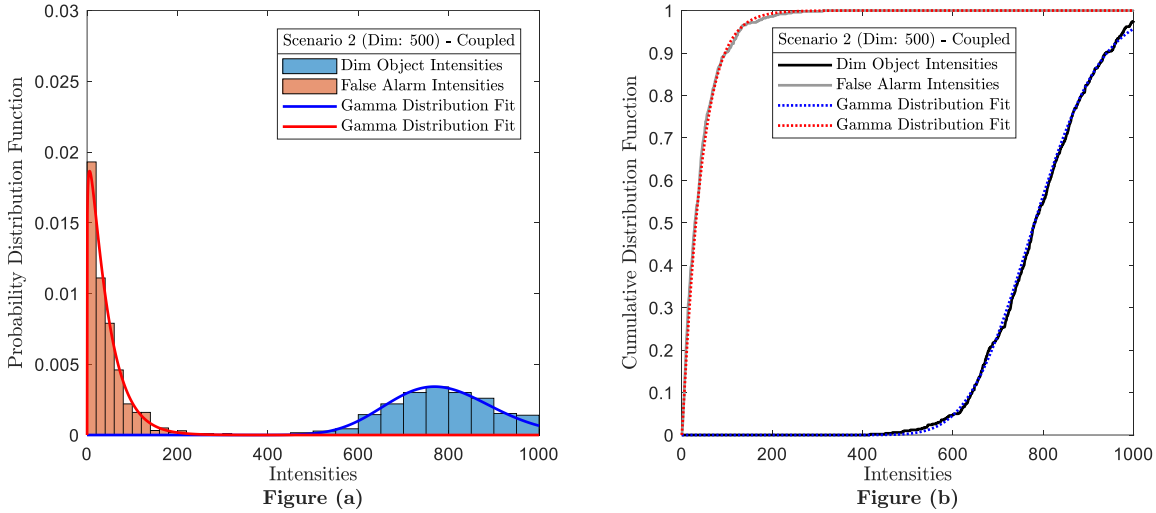


**Figure B-3. The Gamma Fits for the Coupled Approach for Scenario 2.**

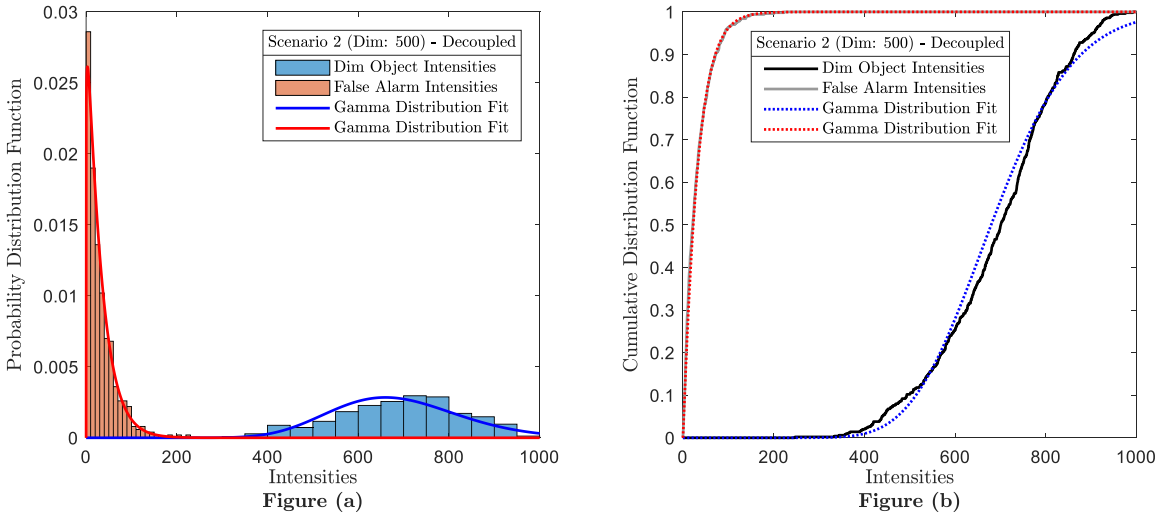


**Figure B-4. The Gamma Fits for the Decoupled Approach for Scenario 2.**

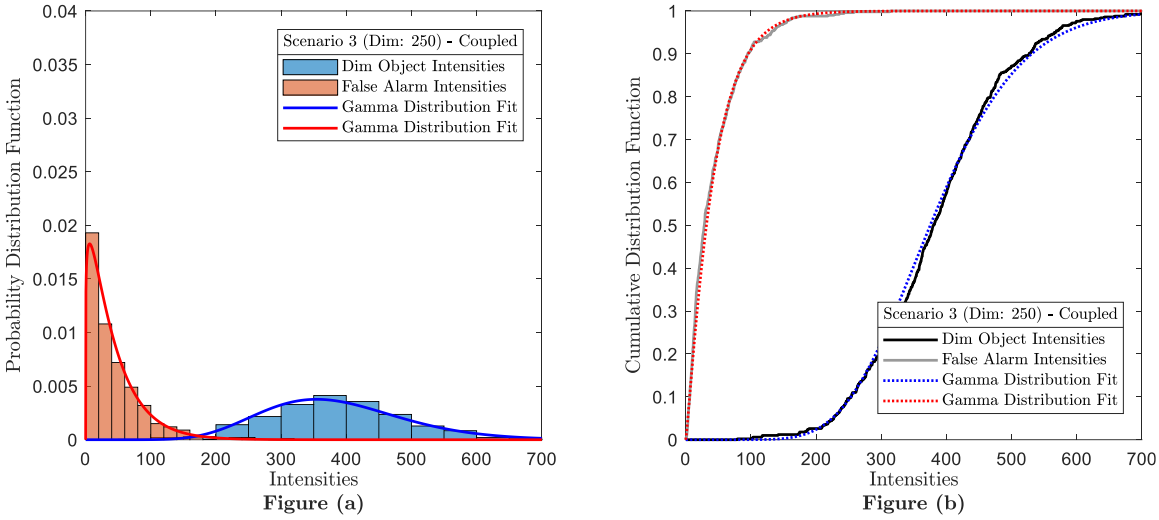


**Figure B-5. The Gamma Fits for the Coupled Approach for Scenario 3.**

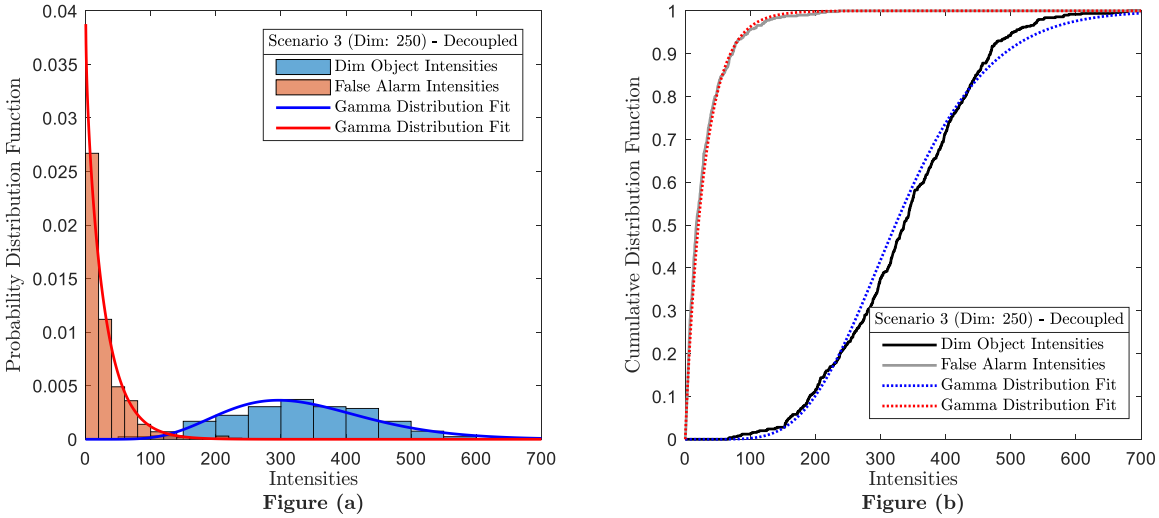


**Figure B-6. The Gamma Fits for the Decoupled Approach for Scenario 3.**

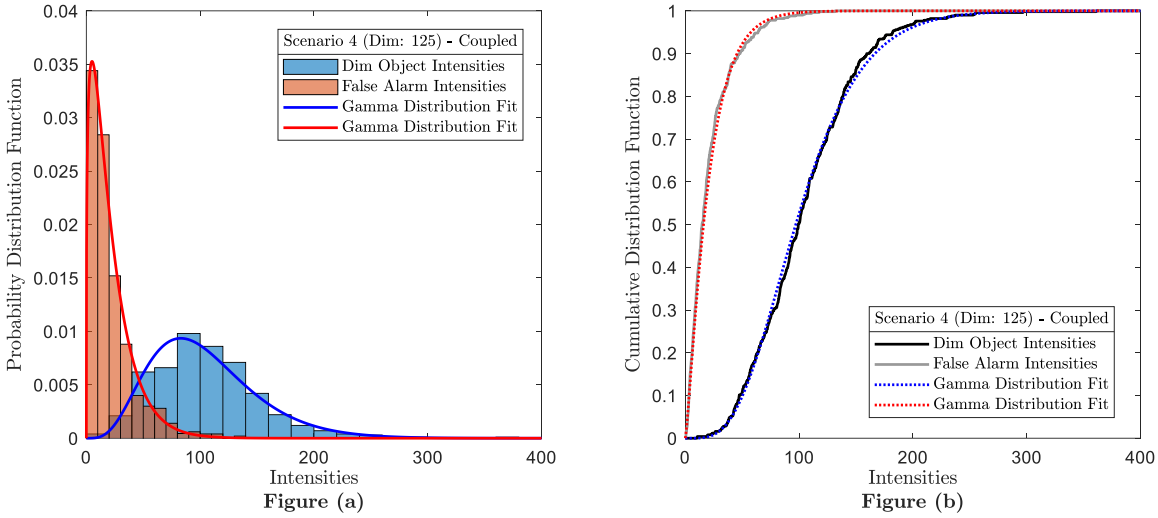


**Figure B-7. The Gamma Fits for the Coupled Approach for Scenario 4.**

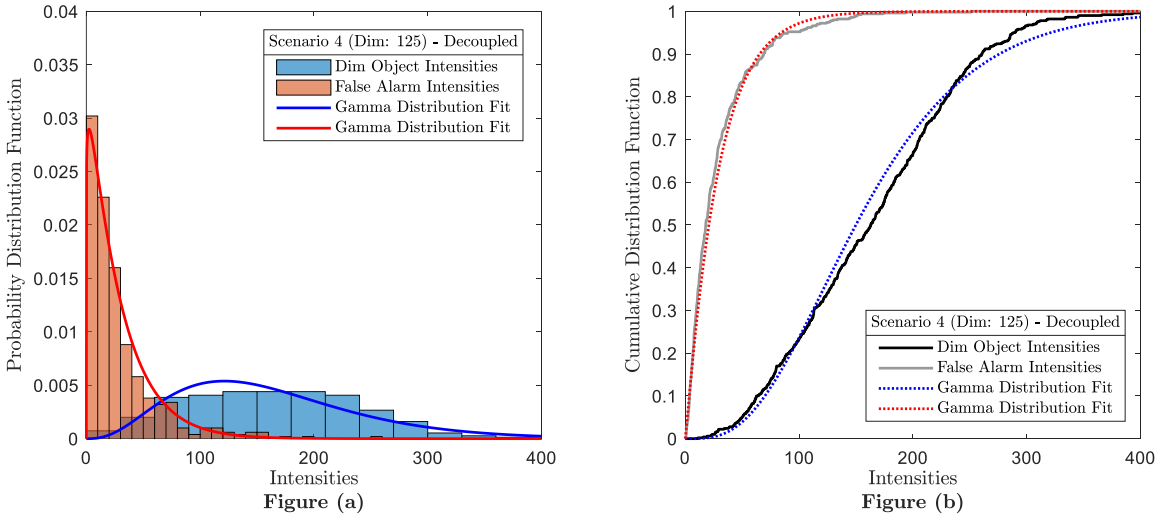


**Figure B-8. The Gamma Fits for the Decoupled Approach for Scenario 4.**

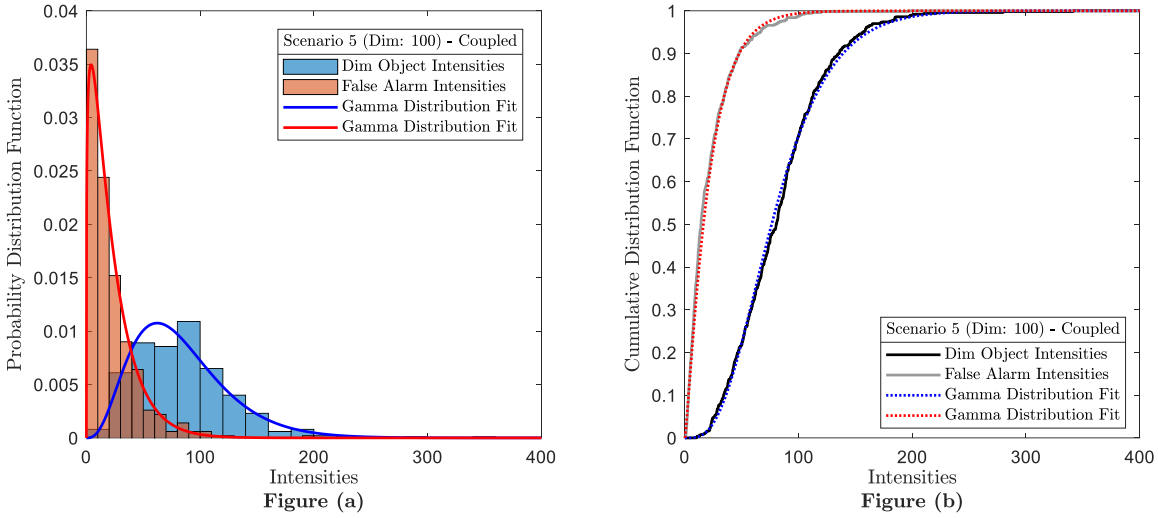


**Figure B-9. The Gamma Fits for the Coupled Approach for Scenario 5.**

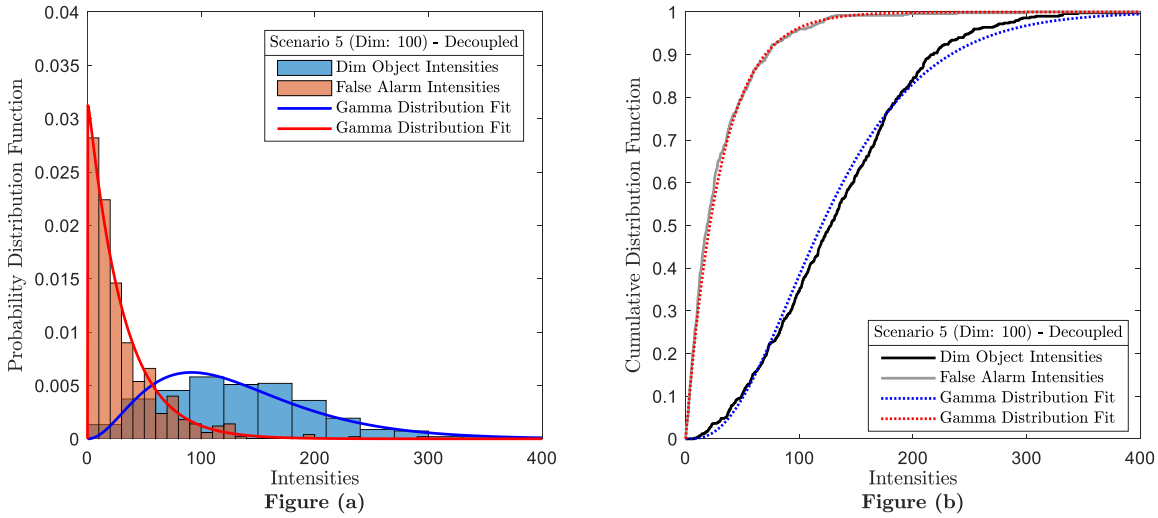


**Figure B-10. The Gamma Fits for the Decoupled Approach for Scenario 5.**

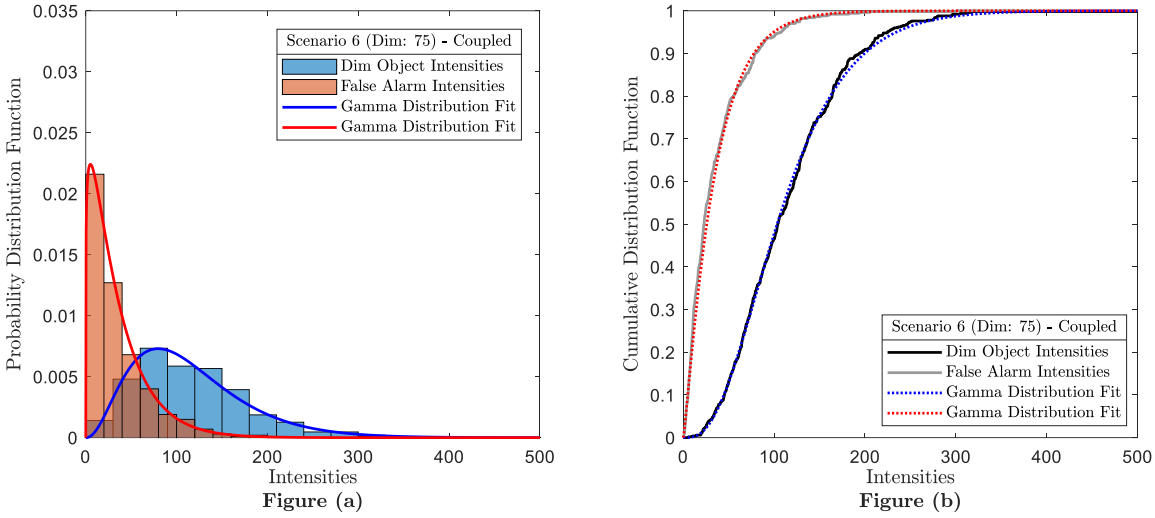


**Figure B-11. The Gamma Fits for the Coupled Approach for Scenario 6.**

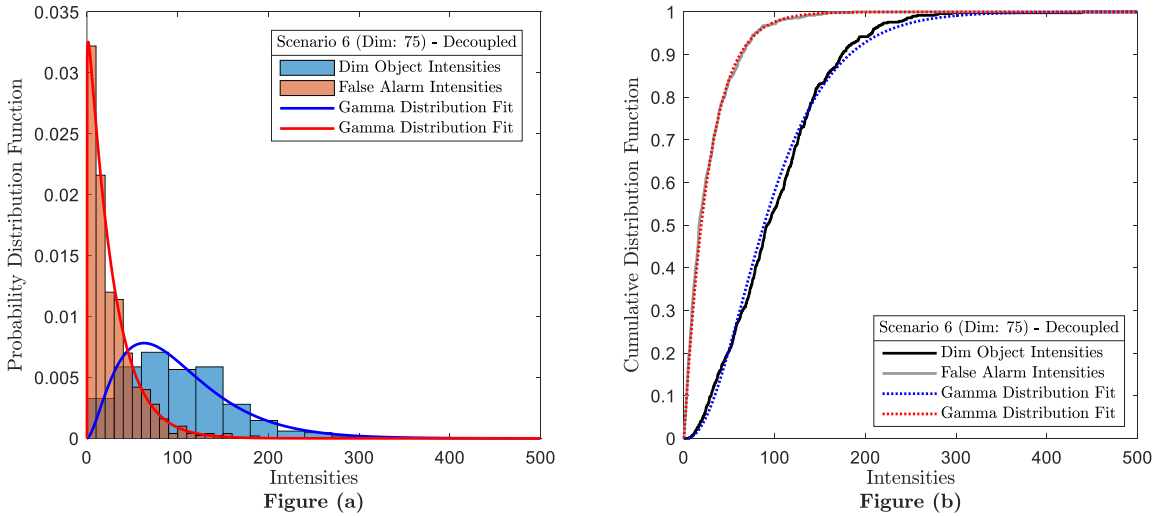


**Figure B-12. The Gamma Fits for the Decoupled Approach for Scenario 6.**

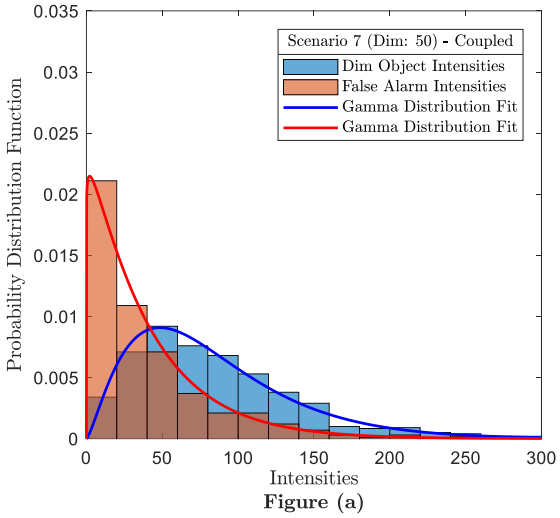

Figure (a)

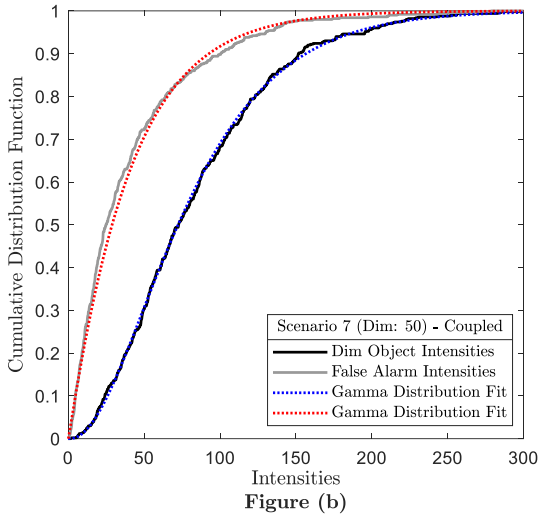

Figure (b)

**Figure B-13. The Gamma Fits for the Coupled Approach for Scenario 7.**

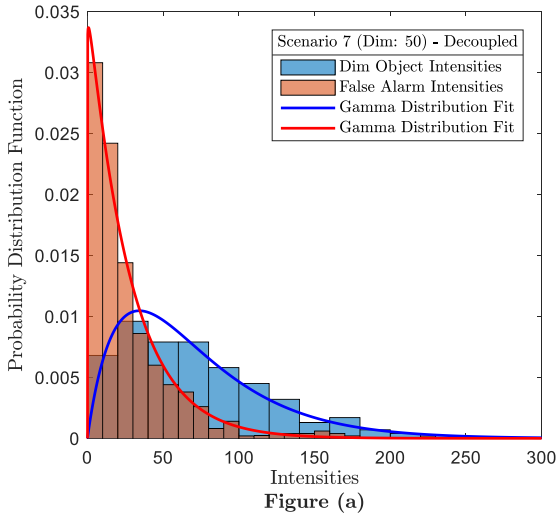

Figure (a)

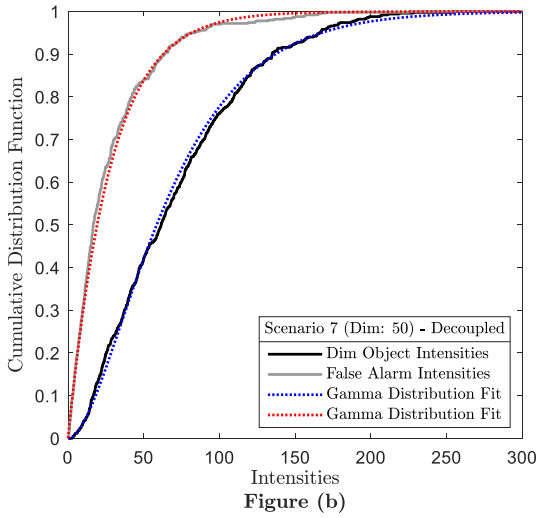

Figure (b)

**Figure B-14. The Gamma Fits for the Decoupled Approach for Scenario 7.**

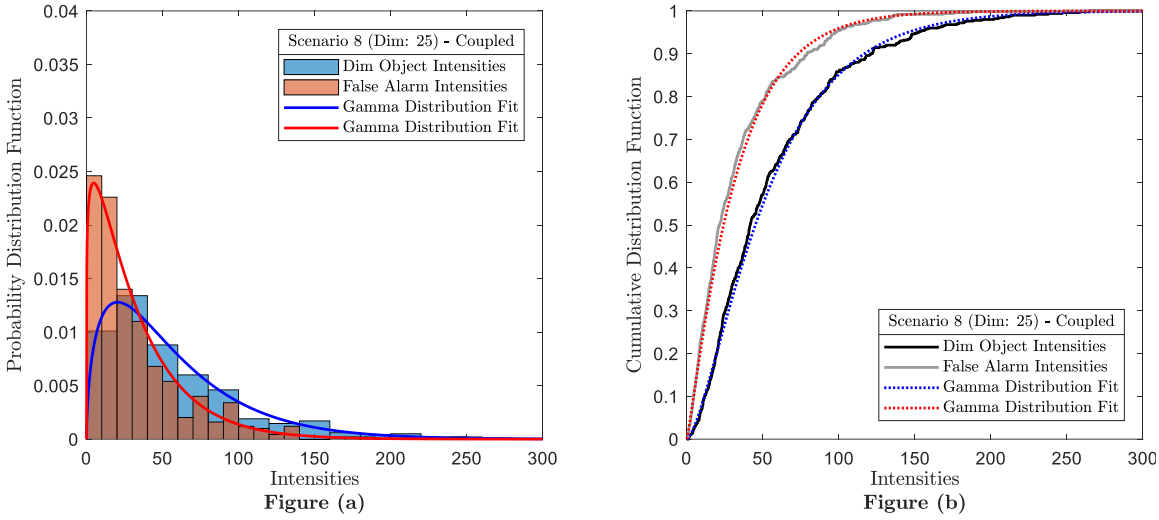


**Figure B-15. The Gamma Fits for the Coupled Approach for Scenario 8.**

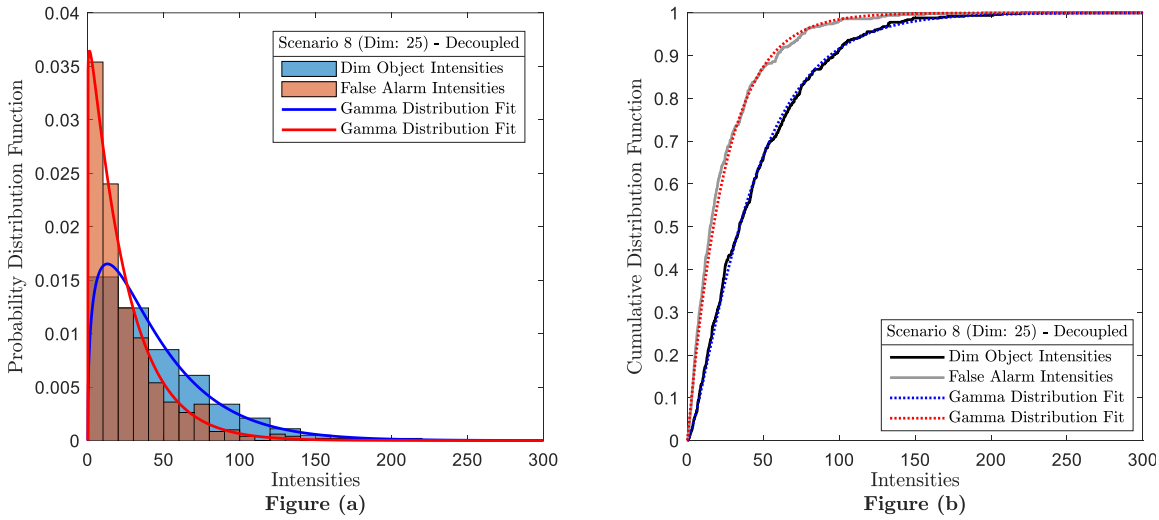


**Figure B-16. The Gamma Fits for the Decoupled Approach for Scenario 8.**

## Appendix C: Gaussian Distribution Fits from Chapter 5

This section shows the Gaussian distribution fits for the dim object intensities and the false alarm intensities obtained from the Multi-Frame Blind Deconvolution (MFBD) and the Neighborhood System Blind Deconvolution (NSBD) for the laboratory data and the computer simulated data from Chapter 5. For the NSBD algorithm, two approaches are used. The first one is the blind deconvolution approach as described in Section 5.5.1 and the other one is processed with the prior knowledge of the shape of the primary object function as described in Section 5.5.2. The distributions and their Gaussian distribution fits are shown in the following figures.

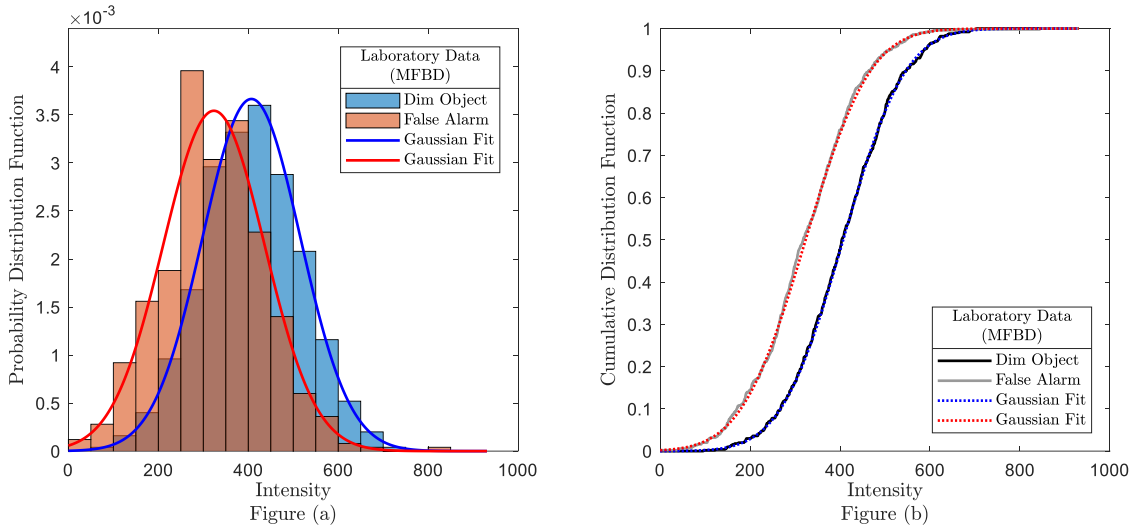


**Figure C-1. The Gaussian Distribution Fits for the Dim Object Intensities and the False Alarm Intensities of the Laboratory Data Processed with the MFBD.**

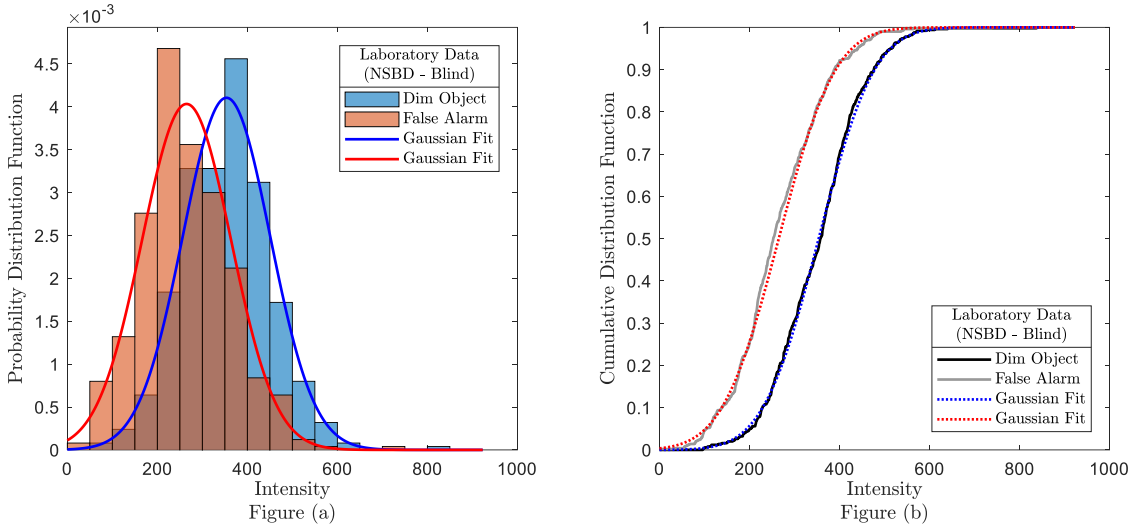


**Figure C-2. The Gaussian Distribution Fits for the Dim Object Intensities and the False Alarm Intensities of the Laboratory Data Processed with the NSBD.**

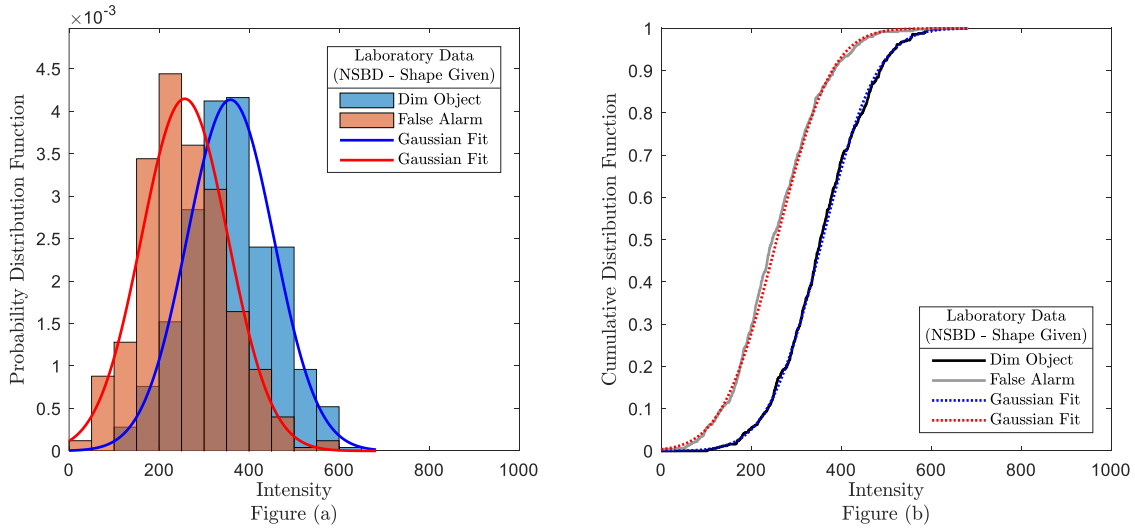


**Figure C-3. The Gaussian Distribution Fits for the Dim Object Intensities and the False Alarm Intensities of the Laboratory Data Processed with the NSBD with the Shape of the Primary Object Function Given.**

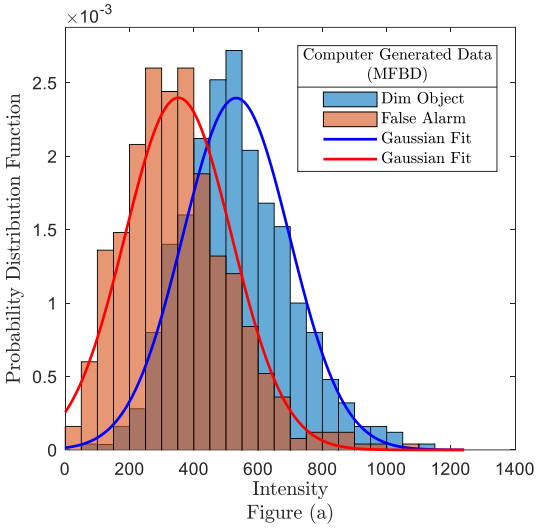

Figure (a)

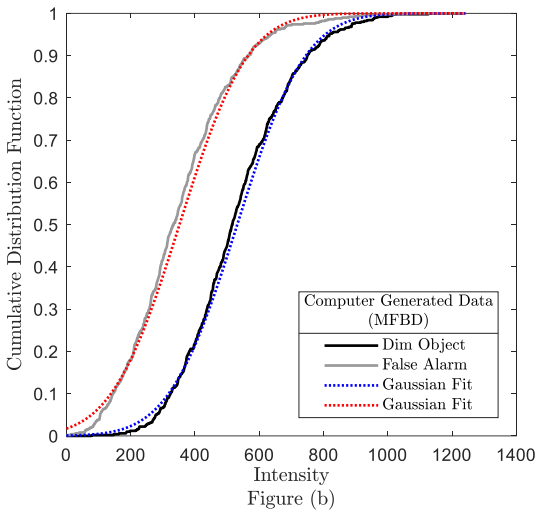

Figure (b)

**Figure C-4. The Gaussian Distribution Fits for the Dim Object Intensities and the False Alarm Intensities of the Computer Generated Data Processed with the MFBD.**

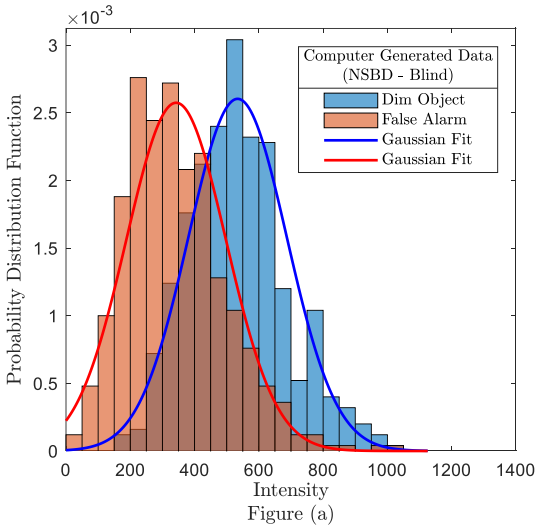

Figure (a)

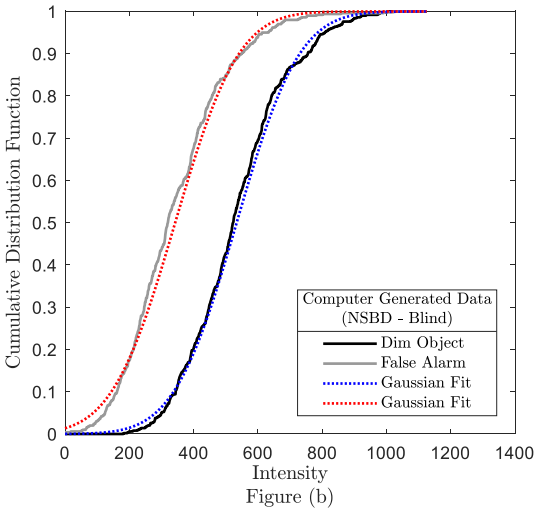

Figure (b)

**Figure C-5. The Gaussian Distribution Fits for the Dim Object Intensities and the False Alarm Intensities of the Computer Generated Data Processed with the NSBD.**

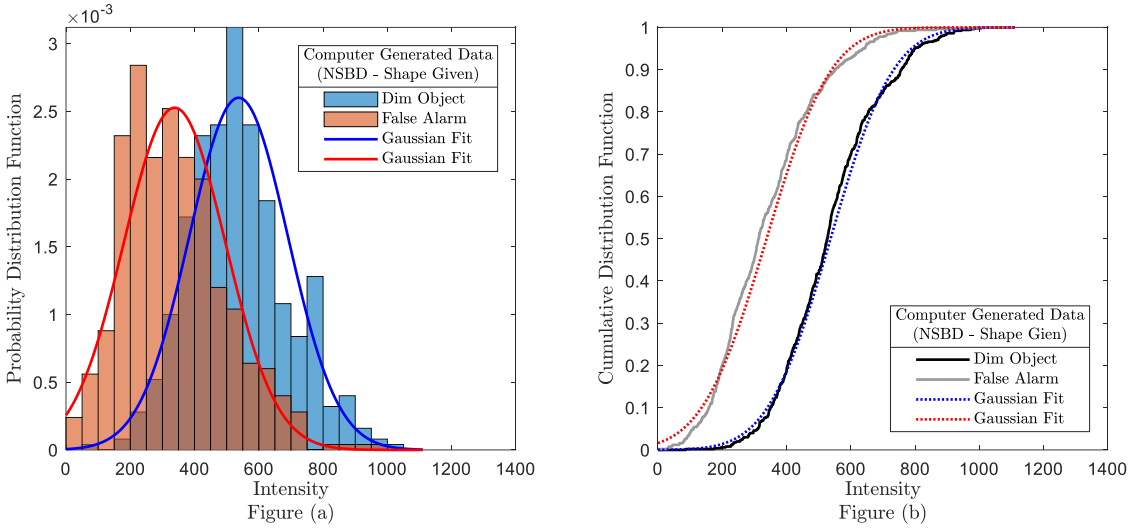


**Figure C-6. The Gaussian Distribution Fits for the Dim Object Intensities and the False Alarm Intensities of the Computer Generated Data Processed with the NSBD with the Shape of the Primary Object Function Given.**

# Appendix D: Distribution Fits from Chapter 6

This appendix shows the distributions of the dim object intensities and the false alarm intensities obtained from the Dimension Reduction Blind Deconvolution (DRBD) from Chapter 6. Figure D-1 and Figure D-2 show the distributions for Scenario 1 using the summation approach and the singular value decomposition (SVD) approach. Figure D-3 shows the distributions for Scenario 1N using the SVD approach. For all scenarios, the intensities of the false alarm are well separated from those of the dim object that either Gamma or Weibull will provide a perfect detection rate.

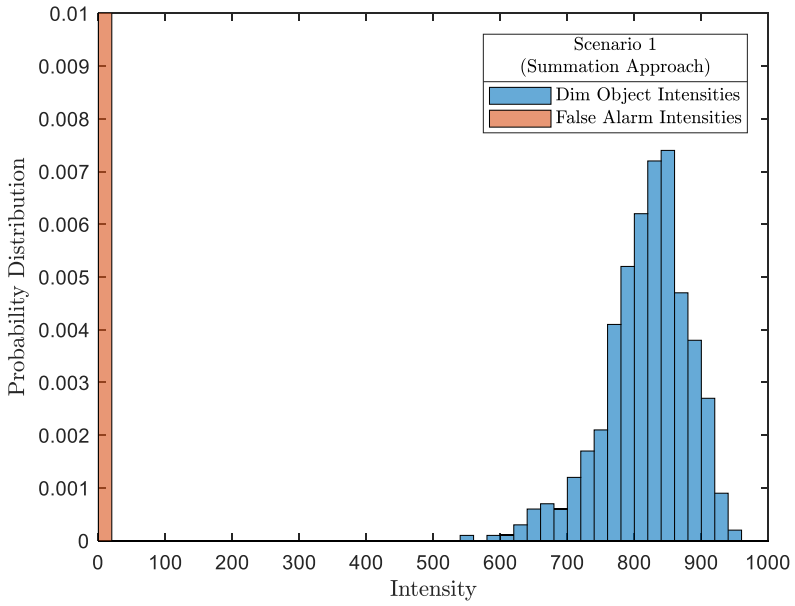


**Figure D-1. The Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 1 Using DRBD with Summation Approach.**

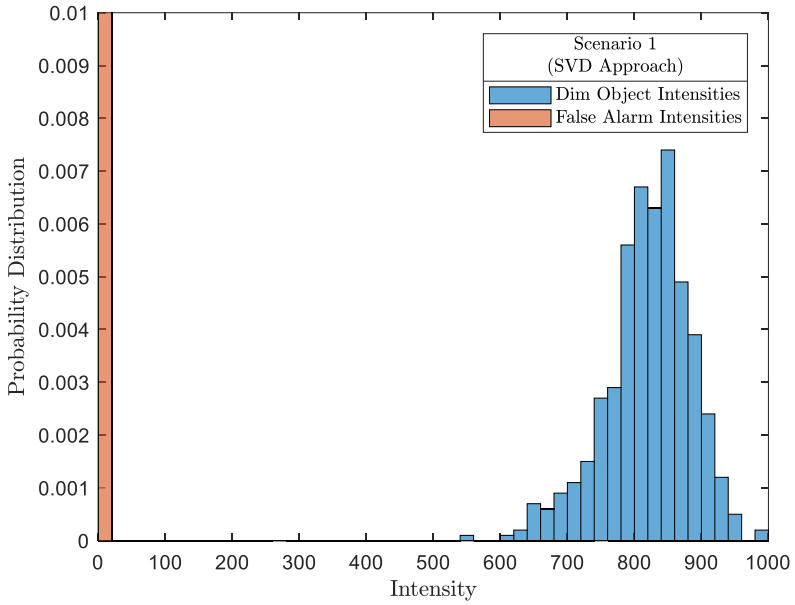


**Figure D-2. The Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 1 Using DRBD with SVD Approach.**

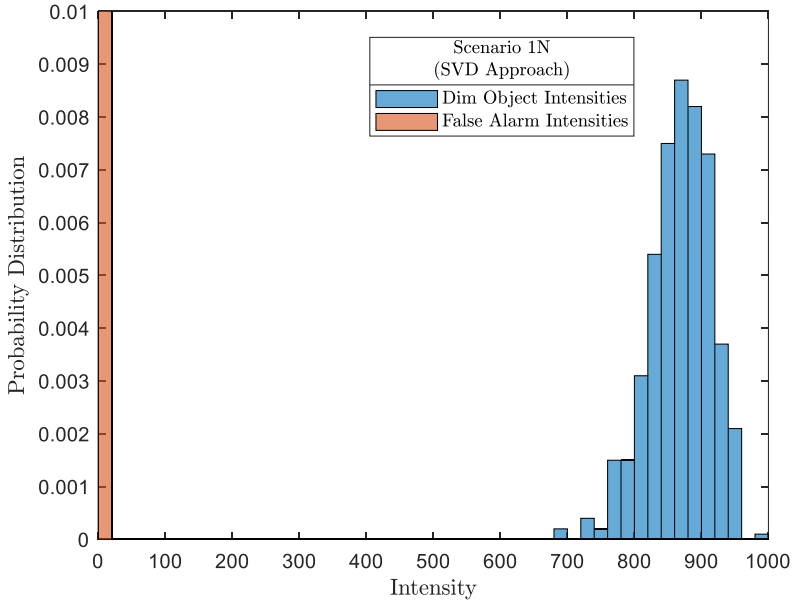


**Figure D-3. The Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 1N Using DRBD with SVD Approach.**

Figure D-4 and Figure D-5 show the distributions for Scenario 8 using the summation approach. In Figure D-4, the distributions are fitted with the Gamma distributions. From the figures, it can be seen that fitting with the Gamma distributions will not provide accurate detection rates. Therefore, the results shown in Figure 6.12 are not fitted with the Gamma distributions.

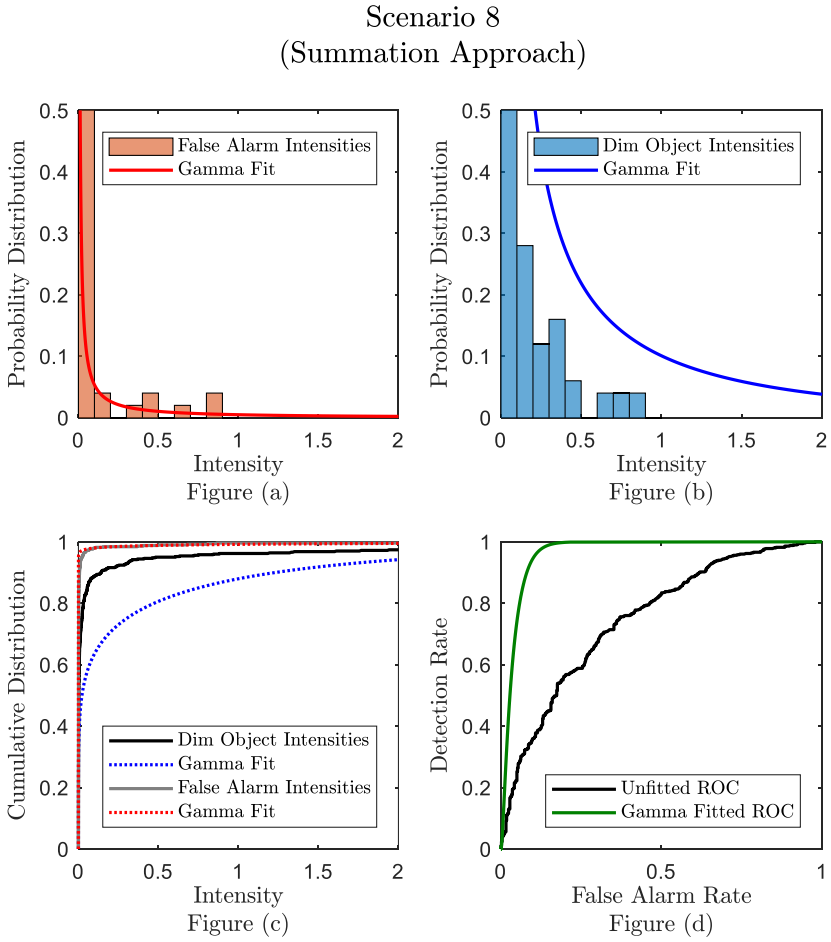


**Figure D-4. The Gamma Fitted Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 8 Using DRBD with the Summation Approach.**

In Figure D-5, the distributions are fitted with the Weibull distributions. From the figures, it can be seen that fitting with the Weibull distributions will not provide accurate detection rates. Therefore, the results shown in Figure 6.12 are not fitted with the Weibull distributions.

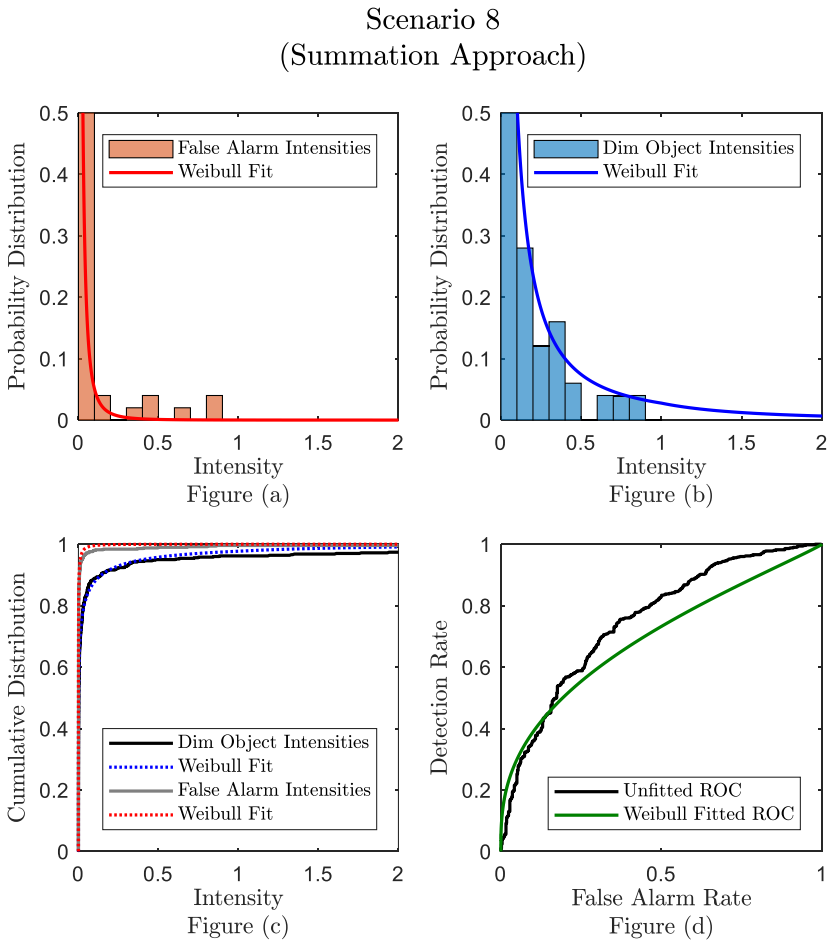


**Figure D-5. The Weibull Fitted Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 8 Using DRBD with the Summation Approach.**

Figure D-6 and Figure D-7 show the distributions for Scenario 8 using the SVD approach. In Figure D-6, the distributions are fitted with the Gamma distributions. From the figures, it can be seen that fitting with the Gamma distributions will not provide accurate detection rates. Therefore, the results shown in Figure 6.13 are not fitted with the Gamma distributions.

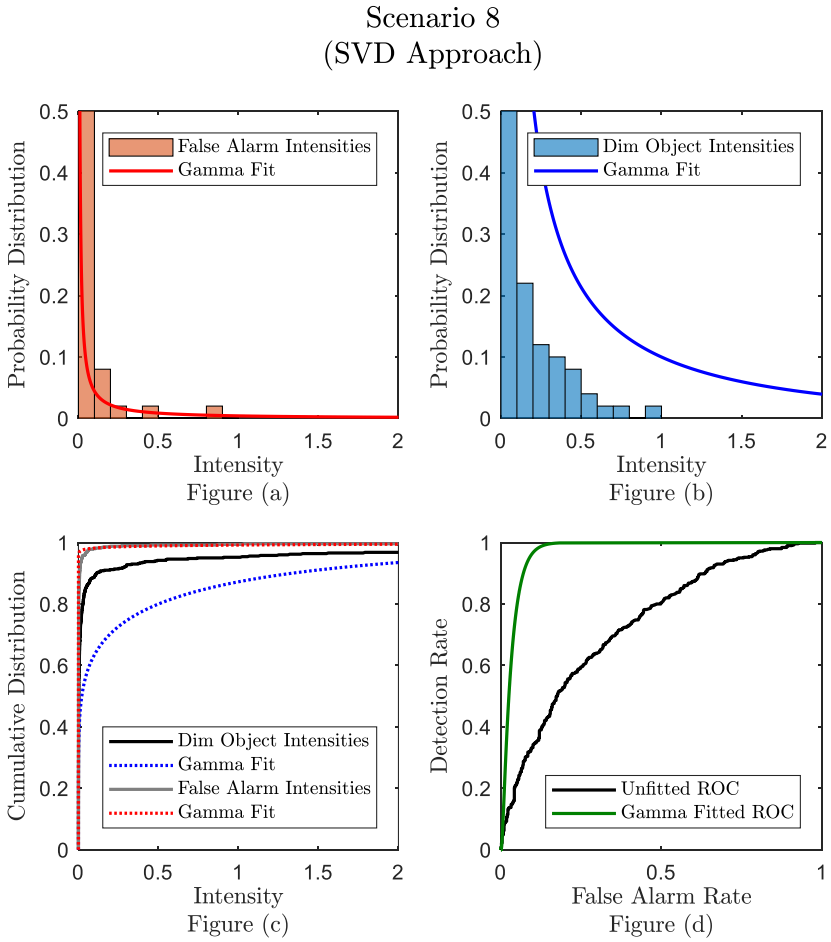


**Figure D-6. The Gamma Fitted Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 8 Using DRBD with the SVD Approach.**

In Figure D-7, the distributions are fitted with the Weibull distributions. From the figures, it can be seen that fitting with the Weibull distributions will not provide accurate detection rates. Therefore, the results shown in Figure 6.13 are not fitted with the Weibull distributions.

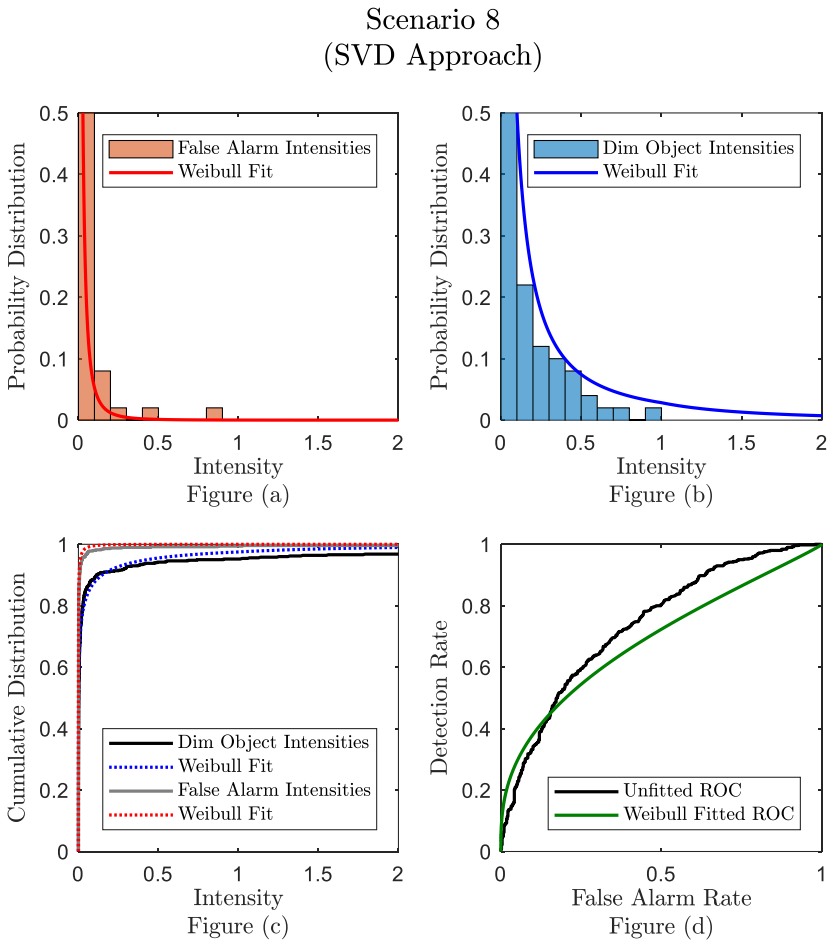


**Figure D-7. The Weibull Fitted Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 8 Using DRBD with the SVD Approach.**

Figure D-8 and Figure D-9 show the distributions for Scenario 8N using the SVD approach. In Figure D-8, the distributions are fitted with the Gamma distributions. From the figures, it can be seen that fitting with the Gamma distributions will not provide accurate detection rates. Therefore, the results shown in Figure 6.20 are not fitted with the Gamma distributions.

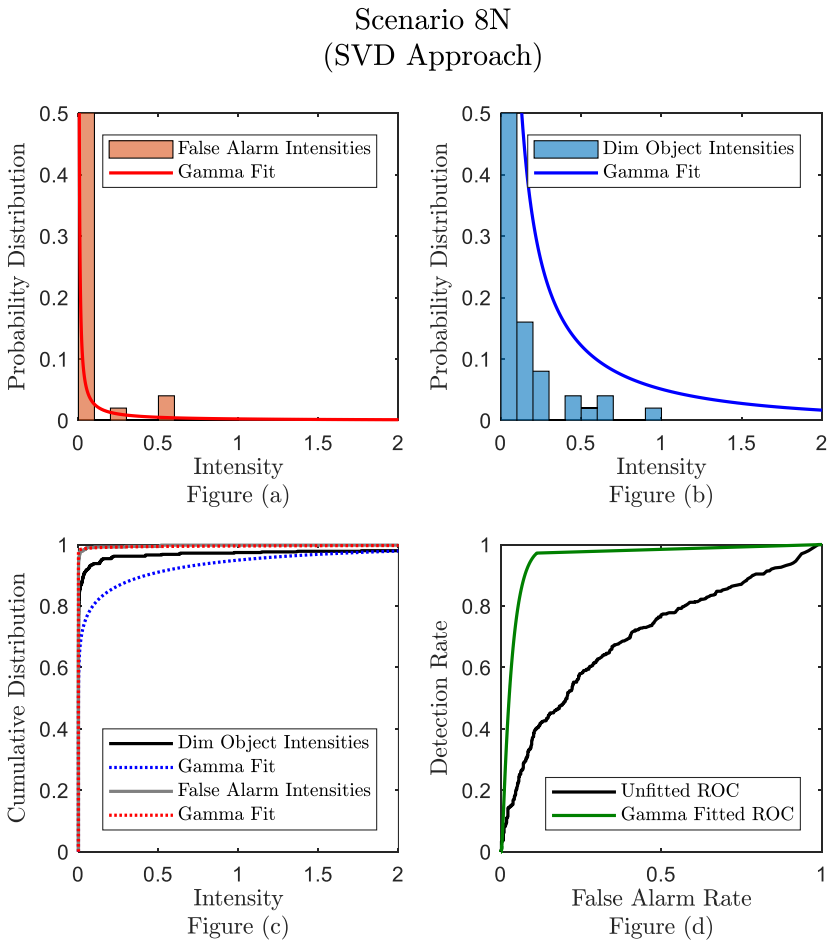


**Figure D-8. The Gamma Fitted Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 8N Using DRBD with the SVD Approach.**

In Figure D-9, the distributions are fitted with the Weibull distributions. From the figures, it can be seen that fitting with the Weibull distributions track the data well. Therefore, the results shown in Figure 6.20 are fitted with the Weibull distributions.

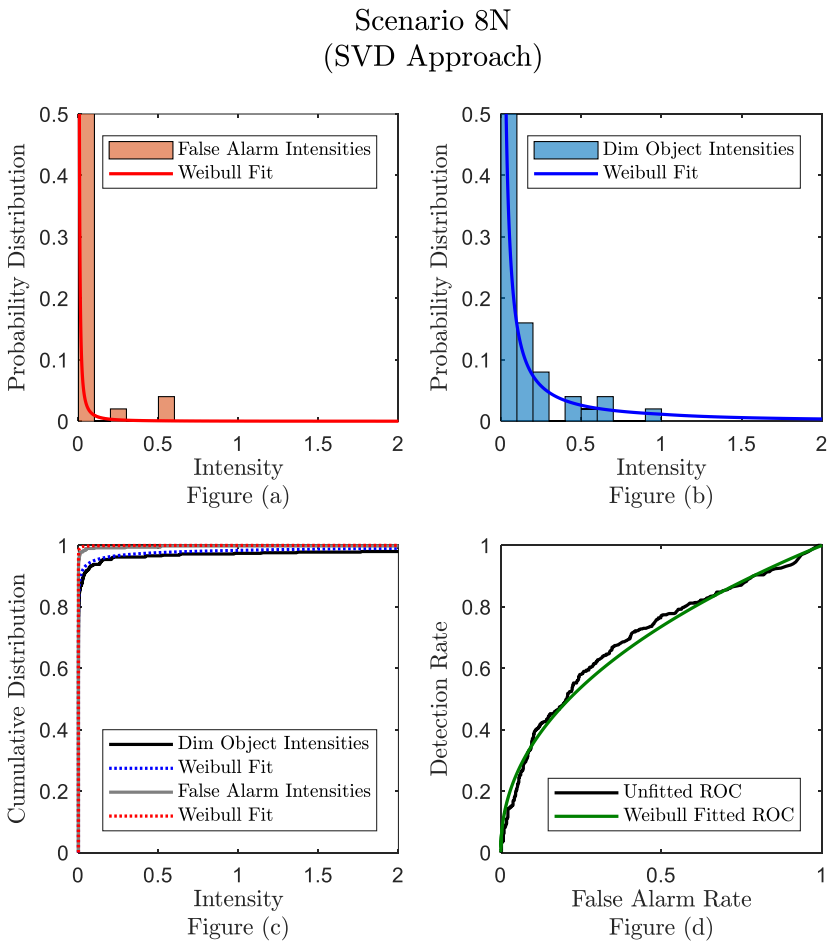


**Figure D-9. The Weibull Fitted Distributions for Dim Object Intensities and False Alarm Intensities for Scenario 8N Using DRBD with the SVD Approach.**

# Bibliography


[1] National Aeronautics and Space Administration, “Sputnik and The Dawn of the Space Age,” 2007. [Online]. Available: https://history.nasa.gov/sputnik/. [Accessed: 30-May-2019].

[2] W. J. Broad and D. E. Sanger, “China Tests Anti-Satellite Weapon, Unnerving U.S.,” *The New York Times*, 2007. [Online]. Available: https://www.nytimes.com/2007/01/18/world/asia/18cnd-china.html. [Accessed: 30-May-2019].

[3] Y. Pushkin and M. Gray, “Russian, U.S. Satellites Collide in Space,” *The Cable News Network*, 2009. [Online]. Available: http://www.cnn.com/2009/TECH/02/12/us.russia.satellite.crash/index.html. [Accessed: 30-May-2019].

[4] SpaceWorks Enterprise, “2019 Nano/Microsatellite Market Forecast, 9th Edition,” 2019. [Online]. Available: https://www.spaceworks.aero/wp-content/uploads/Nano-Microsatellite-Market-Forecast-9th-Edition-2019.pdf. [Accessed: 30-May-2019].

[5] Orbital Debris Program Office, “Orbital Debris Quarterly News,” *Astromaterials Research & Exploration Science*, 2019. [Online]. Available: https://orbitaldebris.jsc.nasa.gov/quarterly-news/pdfs/odqnv23i1.pdf. [Accessed: 30-May-2019].

[6] European Space Agency, “Clean Space,” 2013. [Online]. Available: http://www.esa.int/Our_Activities/Space_Safety/Clean_Space/FAQ_s. [Accessed: 01-Jun-2019].

[7] K. Tate, “Russian Satellite Crash with Chinese ASAT Debris Explained,” *SPACE.com*, 2013. [Online]. Available: https://www.space.com/41418-about-us.html. [Accessed: 30-May-2019].

[8] The Office of the Director of National Intelligence, “National Security Space Strategy,” 2011.

[9] U.S. Strategic Command, “Combined Space Operations Center/614th Air Operations Center,” *USSTRATCOM Media Relations*, 2019. [Online]. Available: https://www.stratcom.mil/Portals/8/Documents/CSpOC_Factsheet_2018.pdf. [Accessed: 25-Oct-2019].

[10] U.S. Air Force, “Combined Space Operations Center established at Vandenberg AFB,” *Joint Force Space Component Command Public Affairs*, 2018. [Online]. Available: https://www.afspc.af.mil/News/Article-Display/Article/1579285/combined-space-operations-center-established-at-vandenberg-afb/. [Accessed: 25-Oct-2019].

[11] Department of Defense, “Department of Defense Establishes U.S. Space Force,” *US Dept of Defense Newsroom*, 2019. [Online]. Available:

https://www.defense.gov/Newsroom/Releases/Release/Article/2045981/department-of-defense-establishes-us-space-force/. [Accessed: 01-Mar-2020].

[12] S. M. Jefferies and J. C. Christou, "Restoration of Astronomical Images by Iterative Blind Deconvolution," *Astrophys. J.*, vol. 415, p. 862, Oct. 1993.

[13] D. A. Hope and S. M. Jefferies, "Compact multiframe blind deconvolution," *Opt. Lett.*, vol. 36, no. 6, p. 867, 2011.

[14] R. Holmes and V. S. R. Gudimetla, "Image reconstructions with active illumination in strong-turbulence scenarios with single-frame blind deconvolution approaches," *Appl. Opt.*, vol. 58, no. 28, pp. 7823–7835, 2019.

[15] G. E. Archer, J. P. Bos, and M. C. Roggemann, "Comparison of bispectrum, multiframe blind deconvolution and hybrid bispectrum-multiframe blind deconvolution image reconstruction techniques for anisoplanatic, long horizontal-path imaging," *Opt. Eng.*, vol. 53, no. 4, 2014.

[16] G. Molesini, "Galileo's telescope and the birth of instrumental optics: a review note," *Appl. Opt.*, vol. 49, no. 16, pp. D1–D5, 2010.

[17] Astronomy Department at Cornell University, "The History of Astronomy," 2015. [Online]. Available: http://curious.astro.cornell.edu/people-and-astronomy/the-history-of-astronomy. [Accessed: 22-Apr-2019].

[18] G. Djorgovski, "AY1 - The Evolving Universe," 2019. [Online]. Available: http://www.astro.caltech.edu/~george/ay1/lec_pdf/Ay1_Lec03.pdf. [Accessed: 24-Apr-2019].

[19] M. Lesser, "A Summary of Charge-Coupled Devices for Astronomy," *Publ. Astron. Soc. Pacific*, vol. 127, no. 957, pp. 1097–1104, 2015.

[20] G. H. Stokes, J. B. Evans, H. E. M. Viggh, F. C. Shelly, and E. C. Pearce, "Lincoln Near-Earth Asteroid Program (LINEAR)," *Icarus*, vol. 148, no. 1, pp. 21–28, 2000.

[21] J. R. Janesick, T. Elliott, S. Collins, M. M. Blouke, and J. Freeman, "Scientific Charge-Coupled Devices," *Opt. Eng.*, vol. 26, no. 8, pp. 692–714, Aug. 1987.

[22] R. D. Richmond and S. C. Cain, *Direct-Detection LADAR Systems*. SPIE Publications, 2010.

[23] S. C. Pohlig, "An algorithm for detection of moving optical targets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 25, no. 1, pp. 56–63, 1989.

[24] T. J. Hardy, "Optical Theory Improvements to Space Domain Awareness," PhD Dissertation, Air Force Institute of Technology, 2016.

[25] D. J. Becker, "Techniques for Improved Space Object Detection Performance from Ground-Based Telescope Systems Using Long and Short Exposure Images," PhD Dissertation, Air Force Institute of Technology, 2018.

[26] D. L. Snyder, R. L. White, and A. M. Hammoud, "Image recovery from data

acquired with a charge-coupled-device camera," *J. Opt. Soc. Am. A*, vol. 10, no. 5, p. 1014, 1993.

[27] D. Bertsimas and J. Tsitsiklis, *Introduction to Probability*. Belmont: Athena Scientific, 2002.

[28] J. Li, "Conditional Probability." [Online]. Available: http://www.personal.psu.edu/jol2/course/stat416/notes/chap3.pdf. [Accessed: 30-Apr-2019].

[29] M. C. Roggemann and B. M. Welsh, *Imaging Through Turbulence*. Boca Raton: CRC Press, 1996.

[30] R. G. Lane, "Simulation of a Kolmogorov Phase Screen," *Waves in Random Media*, vol. 2, no. 3, pp. 209–224, 1992.

[31] B. L. McGlamery, "Computer Simulation Studies Of Compensation Of Turbulence Degraded Images," in *Proc.SPIE*, 1976, vol. 0074.

[32] L. C. Andrews and R. L. Phillips, "Optical Turbulence in the Atmosphere," in *Laser Beam Propagation through Random Media*, 2nd ed., SPIE Publications, 2005, pp. 57–82.

[33] V. I. Tatarski, *Wave Propagation in a Turbulent Medium*. McGraw-Hill Book Company, 1961.

[34] J. D. Schmidt, *Numerical Simulation of Optical Wave Propagation with Examples in MATLAB*. Bellingham: SPIE, 2010.

[35] N. A. Roddier, "Atmospheric Wavefront Simulation Using Zernike Polynomials," *Opt. Eng.*, vol. 29, no. 10, pp. 1174–1180, Oct. 1990.

[36] I. B. Putnam and S. C. Cain, "Modeling a Temporally Evolving Atmosphere with Zernike Polynomials," in *Advanced Maui Optical and Space Surveillance Technologies Conference*, 2012.

[37] R. J. Noll, "Zernike Polynomials and Atmospheric Turbulence," *J. Opt. Soc. Am.*, vol. 66, no. 3, pp. 207–211, 1976.

[38] D. L. Fried, "Statistics of a Geometric Representation of Wave-front Distortion," *J. Opt. Soc. Am.*, vol. 55, no. 11, pp. 1427–1435, 1965.

[39] J. W. Goodman, *Introduction to Fourier Optics*, 3rd ed. Greenwood Village: Roberts & Company Publishers, 2005.

[40] S. Haykin, *Blind Deconvolution*. Englewood Cliffs: Prentice-Hall, 1994.

[41] M. H. Hayes, *Statistical Digital Signal Processing and Modeling*. Hoboken: John Wiley & Sons, 1996.

[42] J. Ferdous and S. Ali, "A Comparison of Wiener and Kalman Filters for the Artifact Suppression from EEG Signal," *Int. J. Sci. Res.*, vol. 6, no. 4, pp. 2029–2035, 2015.

[43] C. Wunsch, "12.864 Inference from Data and Models," *Massachusetts Institute of Technology: MIT OpenCourseWare*, 2005. [Online]. Available: https://ocw.mit.edu/courses/earth-atmospheric-and-planetary-sciences/12-864-inference-from-data-and-models-spring-2005/#. [Accessed: 28-Oct-2019].

[44] S. C. Cain and T. Watts, "Nonparaxial Fourier Propagation Tool for Aberration Analysis and Point Spread Function Calculation," *Opt. Eng.*, vol. 55, no. 8, 2016.

[45] R. K. Ward and B. E. A. Saleh, "Deblurring random blur," *IEEE Trans. Acoust.*, vol. 35, no. 10, pp. 1494–1498, 1987.

[46] B. C. McCallum, "Blind Deconvolution by Simulated Annealing," *Opt. Commun.*, vol. 75, no. 2, pp. 101–105, 1990.

[47] R. G. Lane, "Blind Deconvolution of Speckle Images," *J. Opt. Soc. Am. A*, vol. 9, no. 9, pp. 1508–1514, 1992.

[48] Y. V. Zhulina, "Multiframe blind deconvolution of heavily blurred astronomical images," *Appl. Opt.*, vol. 45, no. 28, pp. 7342–7352, 2006.

[49] F. Šroubek, G. Cristóbal, and J. Flusser, "A unified approach to superresolution and multichannel blind deconvolution," *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2322–2332, 2007.

[50] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," *2009 IEEE Conf. Comput. Vis. Pattern Recognit.*, no. 2, pp. 1964–1971, 2010.

[51] T. Le, R. Chartrand, and T. J. Asaki, "A variational approach to reconstructing images corrupted by poisson noise," *J. Math. Imaging Vis.*, vol. 27, no. 3, pp. 257–263, 2007.

[52] E. Jonsson, T. Chan, and S.-C. Huang, "Total Variation Regularization in Positron Emission Tomography," *Tech. Rep.*, 1998.

[53] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear Total Variation Based Noise Removal Algorithms," *Phys. D*, vol. 60, pp. 259–268, 1992.

[54] T. F. Chan, S. Osher, and J. Shen, "The digital TV filter and nonlinear denoising," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 231–241, 2001.

[55] P. A. Jansson, *Deconvolution of Images and Spectra*, 2nd ed. Mineola, NY: Dover Publications, 2012.

[56] W. H. Richardson, "Bayesian-Based Iterative Method of Image Restoration," *J. Opt. Soc. Am.*, vol. 62, no. 1, pp. 55–59, 1972.

[57] L. B. Lucy, "An Iterative Technique for the Rectification of Observed Distributions," *Astron. J.*, vol. 79, no. 6, pp. 745–754, 1974.

[58] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. R. Stat. Soc. Ser. B*, vol. 39, no. 1, pp. 1–38, 1977.

[59] C. F. Wu, "On the Convergence Properties of the EM Algorithm," *Ann. Stat.*, vol. 11, no. 1, pp. 95–103, 2013.

[60] R. B. Millar, *Maximum Likelihood Estimation and Inference: with examples in R, SAS and ADMB.* Chichester, West Sussex: Wiley, 2011.

[61] L. A. Shepp and Y. Vardi, "Maximum Likelihood Reconstruction Emission Tomography," *IEEE Trans. Med. Imaging*, vol. MI–1, no. 2, pp. 113–122, 1982.

[62] T. J. Schulz, "Multiframe Blind Deconvolution of Astronomical Images," *J. Opt. Soc. Am.*, vol. 10, no. 5, p. 1064, 1993.

[63] J. L. Harris, "Image Evaluation and Restoration," *J. Opt. Soc. Am.*, vol. 56, no. 5, pp. 569–574, 1966.

[64] B. L. McGlamery, "Restoration of Turbulence-Degraded Images," *J. Opt. Soc. Am.*, vol. 57, no. 3, pp. 293–297, 1967.

[65] M. Rubinstein, "Introduction to Recursive Bayesian Filtering," 2009. [Online]. Available: https://people.csail.mit.edu/mrub/talks/filtering.pdf. [Accessed: 06-May-2019].

[66] J. R. Fienup, "Phase Retrieval Algorithms: a personal tour [Invited]," *Appl. Opt.*, vol. 52, no. 1, pp. 45–56, 2013.

[67] J. R. Fienup, "Phase Retrieval Algorithms: a comparison," *Appl. Opt.*, vol. 21, no. 15, pp. 2758–2769, 1982.

[68] J. C. Zingarelli, "Enhancing Ground Based Telescope Performance with Image Processing," Doctoral Dissertation, Air Force Institute of Technology, 2013.

[69] J. C. Zingarelli and S. C. Cain, "Phase Retrieval and Zernike Decomposition using Measured Intensity Data and the Estimated Electric Field," *Appl. Opt.*, vol. 52, no. 31, pp. 7435–7444, 2013.

[70] R. W. Gerchberg and W. O. Saxton, "A Practical Algorithm for the Determination of the Phase from Image and Diffraction Plane Pictures," *Optik (Stuttg).*, vol. 35, pp. 237–246, 1972.

[71] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River: Pearson Prentice Hall, 2008.

[72] U.S. Air Force, "Air Force Maui Optical & Supercomputing Site (AMOS)," 2012. [Online]. Available: https://www.kirtland.af.mil/About-Us/Fact-Sheets/Display/Article/825970/air-force-maui-optical-supercomputing-site-amos/. [Accessed: 24-Nov-2019].

[73] U.S. Air Force, "Starfire Optical Range at Kirtland Air Force Base, New Mexico," 2012. [Online]. Available: https://www.kirtland.af.mil/About-Us/Fact-Sheets/Display/Article/825974/starfire-optical-range-at-kirtland-air-force-base-new-mexico/. [Accessed: 24-Nov-2019].

[74] J. W. Goodman, *Statistical Optics*. New York: John Wiley & Sons, 2000.

[75] M. Hirsch, S. Harmeling, S. Sra, and B. Schölkopf, “Online multi-frame blind deconvolution with super-resolution and saturation correction,” *Astron. Astrophys.*, vol. 531, pp. 1–11, 2011.

[76] S. B. Howell and E. P. Horch, “High-resolution speckle imaging,” *Phys. Today*, vol. 11, 2018.

[77] S. C. Cain, “Super-Resolution Imaging via Expectation-Maximization Estimation of Near Stellar Neighborhoods,” in *Proc. SPIE*, 2018, vol. 10772.

[78] Y. Deng and D. Chu, “Coherence properties of different light sources and their effect on the image sharpness and speckle of holographic displays,” *Sci. Rep.*, vol. 7, no. 1, p. 5893, Jul. 2017.

[79] G. S. Agarwal, G. Gbur, and E. Wolf, “Coherence properties of sunlight,” *Opt. Lett.*, vol. 29, no. 5, p. 459, 2004.

[80] Planetary Science Communications Team, “Earth,” 2019. [Online]. Available: https://solarsystem.nasa.gov/planets/earth/in-depth/. [Accessed: 11-May-2019].

[81] Planetary Science Communications Team, “Our Sun.” [Online]. Available: https://solarsystem.nasa.gov/solar-system/sun/by-the-numbers/. [Accessed: 11-May-2019].

[82] NASA Earth Observatory, “Three Classes of Orbit,” 2009. [Online]. Available: https://earthobservatory.nasa.gov/features/OrbitsCatalog/. [Accessed: 02-Jun-2019].

[83] S. T. Thurman, “Phase-error correction in digital holography using single-shot data,” *J. Opt. Soc. Am. A*, vol. 36, no. 12, p. D47, 2019.

[84] C. J. Pellizzari, M. F. Spencer, and C. A. Bouman, “Phase-error estimation and image reconstruction from digital-holography data using a Bayesian framework,” *J. Opt. Soc. Am. A*, vol. 34, no. 9, p. 1659, 2017.

[85] P. Hickson, “Atmospheric and adaptive optics,” *Astron. Astrophys. Rev.*, vol. 22, no. 1, pp. 1–38, 2014.

[86] EdmundOptics, “25mm Dia. x 200mm FL Uncoated, Double-Convex Lens.” [Online]. Available: https://www.edmundoptics.com/p/25mm-dia-x-200mm-fl-uncoated-double-convex-lens/3082/. [Accessed: 25-Mar-2020].

[87] C. E. Ebeling, *An Introduction to Reliability and Maintainability Engineering*. Long Grove, IL: Waveland Press, 1997.

[88] E. W. Weisstein, “Survival Function,” *MathWorld*. [Online]. Available: https://mathworld.wolfram.com/SurvivalFunction.html. [Accessed: 12-Jun-2020].

[89] H. L. Van Trees and K. L. Bell, *Detection Estimation and Modulation Theory, Part I: Detection, Estimation, and Filtering Theory*. Hoboken, NJ: John Wiley & Sons, 2013.

[90] M. H. Zweig and G. Campbell, “Receiver-operating characteristic (ROC) plots: a

fundamental evaluation tool in clinical medicine," *Clin. Chem.*, vol. 39, no. 4, pp. 561–577, Apr. 1993.

[91] A. P. Bradley, "The Use of the Area Under the ROC Curve in the Evaluation of Machine Learning Algorithms," *Pattern Recognit.*, vol. 30, no. 7, pp. 1145–1159, 1997.

[92] R. M. Aung and S. C. Cain, "Multi-Frame Blind Deconvolution of Closely Spaced Dim Stellar Objects," in *Advanced Maui Optical and Space Surveillance Technologies Conference*, 2019.

[93] ThorLabs, "Item# 8050-TE-GE," 2016. [Online]. Available: https://www.thorlabs.com/catalogpages/Obsolete/2016/8050M-GE-TE.pdf. [Accessed: 25-Nov-2019].

[94] N. J. Yielding, S. C. Cain, and M. D. Seal, "Statistical photocalibration of photodetectors for radiometry without calibrated light sources," *Opt. Eng.*, vol. 57, no. 01, 2018.

[95] S. Cain, "Non-Linear statistical photo-calibration of photodetectorswithout calibrated light sources," *Appl. Opt.*, vol. 59, no. 9, pp. 2767–2775, 2020.

[96] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Process using MATLAB*, 2nd ed. Gatesmark Publishing, 2009.

[97] H. W. Lilliefors, "On the Kolmogorov-Smirnov test for normality with mean and variance unknown," *J. Am. Stat. Assoc.*, vol. 62, pp. 399–402, 1967.

[98] MATLAB, "Lilliefors test," *MathWorks*, 2020. [Online]. Available: https://www.mathworks.com/help/stats/lillietest.html. [Accessed: 09-May-2020].

[99] M. R. Brown, "One-Dimensional Multi-Frame Blind Deconvolution Using Astronomical Data for Spatially Separable Objects," MS Thesis, Air Force Institute of Technology, 2020.

[100] H. Anton and C. Rorres, *Elementary Linear Algebra*, 8th ed. New York, NY: John Wiley & Sons, 2000.

[101] G. H. Golub and C. F. Van Loan, *Matrix Computation*, 4th ed. Baltimore, MD: The Johns Hopkins University Press, 2013.

[102] M. E. Wall, A. Rechtsteiner, and L. M. Rocha, "Singular Value Decomposition and Principal Component Analysis," *A Pract. Approach to Microarray Data Anal.*, pp. 91–109, 2005.

[103] MATLAB, "Singular Value Decomposition," 2020. [Online]. Available: https://www.mathworks.com/help/matlab/ref/double.svd.html;jsessionid=8d4a53d2fcdbd08430f66fad34a9. [Accessed: 15-May-2020].

[104] MATLAB, "Weibull Distribution," 2020. [Online]. Available: https://www.mathworks.com/help/stats/weibull-distribution.html. [Accessed: 19-May-2020].

[105] G. J. Hahn and S. S. Shapiro, *Statistical Models in Engineering*. New York: John Wiley & Sons, 1967.

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 074-0188*



| 1. REPORT DATE *(DD-MM-YYYY)* | 2. REPORT TYPE | 3. DATES COVERED *(From – To)* |
|---|---|---|
| 18-09-2020 | Doctoral Dissertation | Sep 2018 - Sep 2020 |

**TITLE AND SUBTITLE**

Improving Closely Spaced Dim Stellar Objects Through Improved Multi-Frame Blind Deconvolution

**5a. CONTRACT NUMBER**

**5b. GRANT NUMBER**

**5c. PROGRAM ELEMENT NUMBER**

**6. AUTHOR(S)**

Ronald M. Aung, Maj, USAF

**5d. PROJECT NUMBER**

**5e. TASK NUMBER**

**5f. WORK UNIT NUMBER**

**7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)**
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
Wright-Patterson AFB OH 45433-7765

**8. PERFORMING ORGANIZATION REPORT NUMBER**

AFIT-ENG-DS-20-S-004

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
Air Force Office of Scientific Research
Michael K. Yakes
875 N. Randolph St, Arlington, VA 22203
(571) 263-6926
michael.yakes@us.af.mil

**10. SPONSOR/MONITOR'S ACRONYM(S)**

AFOSR/RTB

**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

**12. DISTRIBUTION/AVAILABILITY STATEMENT**
DISTRUBTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

**13. SUPPLEMENTARY NOTES**
This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

**14. ABSTRACT**
This dissertation focuses on improving the ability to detect dim stellar objects that are in close proximity to a bright one, through statistical image processing using short exposure images. The goal is to improve the space domain awareness capabilities with the existing infrastructure. Two new algorithms are developed. The first one is through the Neighborhood System Blind Deconvolution where the data functions are separated into the bright object, the neighborhood system, and the background functions. The second one is through the Dimension Reduction Blind Deconvolution, where the object function is represented by the product of two matrices. Both are designed to overcome the photon counting noise and the random and turbulent atmospheric conditions. The performance of the algorithms are compared with that of the Multi-Frame Blind Deconvolution. The new algorithms are tested and validated with computer generated data. The Neighborhood System Blind Deconvolution is also modified to overcome the undersampling effects since it is validated on the undersampled laboratory collected data. Even though the algorithms are designed for ground to space imaging systems, the same concept can be extended for space to space imaging. This research provides two better techniques to improve closely space dim object detection.

**15. SUBJECT TERMS**
Closely Spaced Dim Objects, Stellar Neighborhood, Multi-Frame Blind Deconvolution, Statistical Image Processing

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | Dr. Stephen C. Cain, AFIT/ENG |
| U | U | U | UU | 219 | 19b. TELEPHONE NUMBER *(Include area code)* (937) 255-6565, x4716 stephen.cain@afit.edu |

**Standard Form 298 (Rev. 8-98)**
Prescribed by ANSI Std. Z39-18