One-Dimensional Multi-Frame Blind Deconvolution Using Astronomical Data for Spatially Separable Objects

THESIS

Marc R. Brown, Captain, USAF

AFIT-ENG-MS-20-M-008

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this document are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT-ENG-MS-20-M-008

One-Dimensional Multi-Frame Blind Deconvolution Using Astronomical Data for Spatially Separable Objects

THESIS

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 Master of Science in Electrical Engineering

Marc R. Brown, B.S.E.E.

Captain, USAF

26 March 2020

AFIT-ENG-MS-20-M-008

One-Dimensional Multi-Frame Blind Deconvolution Using Astronomical Data for Spatially Separable Objects

THESIS

Marc R. Brown, B.S.E.E.
Captain, USAF

Committee Membership:

Maj David J. Becker, Ph.D
Chair

Stephen C. Cain, Ph.D
Member

Lt Col Scott J. Pierce, Ph.D
Member

AFIT-ENG-MS-20-M-008

Abstract

Blind deconvolution is used to complete missions to detect adversary assets in space and to defend the nations assets. A new algorithm was developed to perform blind deconvolution for objects that are spatially separable using multiple frames of data. This new one-dimensional approach uses the expectation-maximization algorithm to blindly deconvolve spatially separable objects. This object separation reduces the size of the object matrix from an NxN matrix to two singular vectors of length N. With limited knowledge of the object and point spread function the one-dimensional algorithm successfully deconvolved the objects in both simulated and laboratory data. Comparing the one-dimensional blind deconvolution algorithm to the two-dimensional blind deconvolution algorithm saw a decrease in error when comparing the intensities for both low and high signal-to-noise ratio data sets. The new algorithm blindly deconvolved multiple spatially separable objects, to include, a binary star system. The last test for this algorithm was to perform blind deconvolution on data collected in a laboratory setting.

Table of Contents

Page

Page

List of Figures

List of Tables

Table Page

One-Dimensional Multi-Frame Blind Deconvolution Using Astronomical Data for Spatially Separable Objects

I. Introduction

This chapter discusses the background of the problem that space domain awareness (SDA) agencies face, the motivation behind the research, the goals of the research. Lastly, this chapter will outline the organization of the thesis.

1.1 Motivation

The motivation for this research is to aid in deconvolving objects that are spatially separable with an unknown point spread function (PSF) and no additional knowledge of the object. These objects can be satellites and debris in low earth orbit (LEO) or geosynchronous orbit, or stars that are light-years away. Currently, SDA agencies utilize the two-dimensional blind deconvolution algorithm that is computationally intensive on the core processing unit (CPU) [1, 2]. Reducing the amount of Fourier transform computations and the time it takes to perform the algorithm would enable real time blind deconvolution. This real time blind deconvolution would enable on-the-fly adjustments of ground-based systems to make corrections to the system. This research will be able to reduce the number of computations when calculating the object by separating the object into it's x and y components.

1.2 Background

Organizations such as the United States Space Force and Air Force Research Laboratory (AFRL) take pride in their ability to perform blind deconvolution in a manner that is able to aid in assessing threats to the military's assets in space. With a significant number of objects being put into orbit around Earth, the next ten years will test the ability of blind deconvolution algorithms [3]. The ability to do blind deconvolution faster will be put to the test by objects that are smaller, such as satellites and small debris. The protection of these assets in space are a focus for some military organizations. A deconvolution algorithm that is able to deconvolve blindly and rapidly with no degradation in performance or increase in equipment cost is desired.

Images used by organizations that examine this data have aberrations caused by the geometry and material of ground-based telescopes [4]. A ground-based telescope will also have atmospheric turbulence from the heating and cooling of the air causing a "boiling" effect [5]. There are many blind deconvolution algorithms that are used today that are able to perform blind deconvolution though they are very slow and typically are post-processed hours after the data is collected [2]. This post-processing eliminates the ability to make corrections on-the-fly to ensure the collected images are capable of detecting the resident space objects.

This proposed research effort would develop a new blind deconvolution algorithm that would be able to quickly and effectively blindly deconvolve an object or multiple objects. This research will focus on the knowledge that the object may be spatially separable and can be broken into separate components. This research will aid in speeding up the blind deconvolution algorithm and moving the computation power from the CPU to the graphics processing unit to perform one-dimensional convolutions instead of the traditional two-dimensional Fourier transforms.

1.3 Research Goals

The goal of this research is to develop a new blind deconvolution algorithm that utilizes a one-dimensional approach versus the traditional two-dimensional approach. This research will follow the expectation-maximization blind deconvolution algorithm developed by Schulz, using multiple frames of astronomical data and no assumption about the object [6]. This new research will develop a new algorithm for the assumption that the object is spatially separable to estimate the object and the PSF for each frame of data. The simulated experiments will demonstrate spatially separable blind deconvolution for binary star systems and other spatially separable objects. This one-dimensional algorithm will also be able to estimate the object with better accuracy compared to the two-dimensional algorithm. To measure the success of the new algorithm applied to laboratory data, the results will be compared to images of the object gathered through a system without atmospheric turbulence.

1.4 Thesis Organization

Chapter II provides the necessary knowledge for this research to include the atmospheric phase screen, blind deconvolution and necessary probability relationship. Chapter III describes the methodology for how the research was performed in a simulated setting. The results of both the simulated and laboratory experiments are presented in Chapter IV. Lastly, in Chapter V the research is summarized to include the results and any future work that would benefit this research.

II. Background and Literature Review

This chapter explains the technical background required to understand the concepts of this research. First, the chapter discusses how Zernike polynomials and the atmospheric phase screen are generated for this research. The Gerchberg-Saxton phase retrieval method is discussed to explain how the phase was reconstructed. Then the description of the blind deconvolution algorithm is explained. Finally, the relationship of the probability of the incomplete data given the complete data is derived.

2.1 Zernike Polynomials

Zernike Polynomials, introduced by physicist Frits Zernike, are polynomials that are orthogonal on a unit circle [7]. These Zernike polynomials describe the characteristics of light through an optical system. Optical aberrations, such as tilting of light across the x and y axis, can be described as a scaled version of these Zernike polynomials by using an aberration coefficient, called Zernike coefficients. Noll created a modified set of Zernike polynomials that describes the Kolmogoroff's spectrum of turbulence using Zernike polynomials and is defined in Eq. (1). In this equation, $R_n^m(r)$ is the radial function, found in Eq. (2). The values of m and n are integers and are the azimuthal frequency and radial degree, respectively. They also follow the following rule where $m \leq n$, $n - |m| =$ even. Lastly, the radius around the unit circle is denoted as r. This modified set of Zernike polynomials allows for the separation into a radius and phase in the unit circle, $Z_j\left(p, \theta\right)$.

$$\left.\begin{aligned} Z_{even\,j} &= \sqrt{n+1}R_n^m\left(r\right)\sqrt{2}\cos\left(m\theta\right) \\ Z_{odd\,j} &= \sqrt{n+1}R_n^m\left(r\right)\sqrt{2}\sin\left(m\theta\right) \end{aligned}\right\} m \neq 0 \tag{1}$$
$$Z_j = \sqrt{n+1}R_n^0\left(r\right) \qquad\qquad m = 0$$

$$R_n^m(r) = \sum_{s=0}^{(n-m)/2} \frac{(-1)^s(n-s)!}{s![(n+m)/2-s]![(n-m)/2-s]!} r^{n-2s} \tag{2}$$

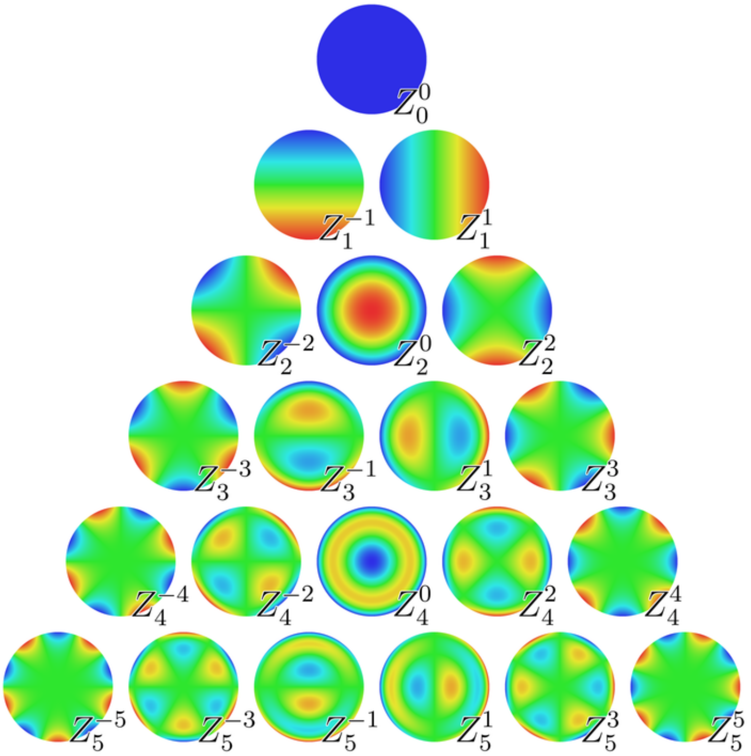

Figure 1: Zernike polynomials for the first 21 Zernikes.

The visual characteristics of the Zernike polynomials can be seen in Figure 1, which displays the Zernike polynomials from Zernike zero to Zernike twenty-one. In this research, the aberration coefficients that are simulated utilize Zernike one through Zernike twelve due to the drop in aberration power with increasing Zernike index.

Zernike zero is piston and affects the light by causing a time delay for photons arriving on the charge-coupled device (CCD). When simulating and collecting laboratory data the piston Zernike is non-distortive and unobservable in the frames of data [8]. The visual representation aids in the ability to describe how a light source is going to react to an optical system. These Zernike polynomials are utilized to generate an atmospheric phase screen to simulate how the atmosphere will react to a light source.

2.2 Atmospheric Phase Screen

These Zernikes are utilized to model optical aberrations caused by the boiling effect of the atmosphere. This atmosphere modeling is used to generate a phase screen with accurate statistics of the atmosphere. The use of Noll's modified set of Zernike polynomials sets up the random phase screen, which will be used for the atmospheric phase screen seen in Eq. (3). Where θ_{atm} is the random phase screen and Z_j is the corresponding Zernike polynomial [5].

$$\theta_{atm}\left(R_p, \theta\right) = \sum_j a_j Z_j\left(p, \theta\right) \tag{3}$$

To generate the phase screen the amplitude, a_j, is calculated by multiplying the Cholesky decomposition, $\underline{\Phi}$, of the covariance matrix by a zero-mean, unit-variance random vector, $\vec{n}$. This amplitude is found in Eq. (4).

$$\vec{a}_j = \underline{\Phi}\vec{n} \tag{4}$$

The Cholesky decomposition is a factorization technique to produce a lower triangular matrix that when multiplied by it's conjugate will generate the original matrix. The covariance matrix of the Zernike polynomials is decomposed to generate the required Cholesky decomposition to produce the amplitude. Following the work in [5], the covariance matrix and Cholesky decomposition is calculated and derived.

The covariance matrix was calculated from the Zernikes, one to nine, and followed Noll's derivation of the Zernike covariance matrix, found in Table 1. The piston Zernike is ignored due to piston causes a time delayed signal, which doesn't effect the way the signal appears on a CCD or photodiode. The piston Zernike is the 0^{th} column and row.

Table 1: Covariance matrix using Noll's derivation calculated out to the tenth Zernike.

m'\m	1	2	3	4	5	6	7	8	9
1	4.572	0.000	0.000	0.000	0.000	0.000	-0.144	0.000	0.000
2	0.000	4.572	0.000	0.000	0.000	-0.144	0.000	0.000	0.000
3	0.000	0.000	0.236	0.000	0.000	0.000	0.000	0.000	0.000
4	0.000	0.000	0.000	0.236	0.000	0.000	0.000	0.000	0.000
5	0.000	0.000	0.000	0.000	0.236	0.000	0.000	0.000	0.000
6	0.000	-0.144	0.000	0.000	0.000	0.063	0.000	0.000	0.000
7	-0.144	0.000	0.000	0.000	0.000	0.000	0.063	0.000	0.000
8	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.063	0.000
9	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.063

Taking the Cholesky decomposition of the covariance matrix in Table 1 gives the Cholesky matrix found in Table 2. The table of random vectors that is used for the simulated example to generate the phase screen is found in Table 3.

Table 2: Cholesky decomposition of the covariance matrix for 10 Zernikes.

m'\m	1	2	3	4	5	6	7	8	9
1	2.138	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2	0.000	2.138	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	0.000	0.000	0.486	0.000	0.000	0.000	0.000	0.000	0.000
4	0.000	0.000	0.000	0.486	0.000	0.000	0.000	0.000	0.000
5	0.000	0.000	0.000	0.000	0.486	0.000	0.000	0.000	0.000
6	0.000	-0.067	0.000	0.000	0.000	0.242	0.000	0.000	0.000
7	-0.067	0.000	0.000	0.000	0.000	0.000	0.242	0.000	0.000
8	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.251	0.000
9	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.251

Multiplying the Cholesky matrix and the random normal vector generates the aberration coefficients found in Table 4. The aberration coefficients are then multi-

Table 3: Zero-mean, unit variance random vector for the example atmospheric phase screen.

n_j	Value
1	3.035
2	0.725
3	-0.063
4	0.715
5	-0.205
6	-0.124
7	1.490
8	1.409
9	1.417

plied by the Zernike polynomials that they correspond to and summed together to generate a random atmospheric phase screen. This atmospheric phase screen is found in Fig. 2 and describes how light will react with the "boiling" effect caused by the atmosphere from the heating and cooling of the air. Using the atmospheric phase screen and Eq.(5), an atmospheric phase screen across the aperture is generated, where $A(x,y)$ is the aperture and $P(x,y)$ is the pupil function. The PSF is built from fourier transforms of this pupil function. The PSF for the example atmospheric phase screen is found in Fig. 3.

$$P(x,y) = A(x,y)\cos(\theta_{atm}) + \sqrt{-1}\sin(\theta_{atm}) \tag{5}$$

Converting the PSF to an optical transfer function (OTF) is essential when utilizing the processor to do the computation versus a graphics card. When utilizing a PSF in the numerous convolution operations required to accomplish blind deconvolution algorithms, accomplishing them on a graphics card can be faster then on a CPU. This is due to the ability of the graphics card to graphically compute the convolution versus the CPU computing the convolutions numerically. Computing the OTF, H, can be accomplished by taking the Fourier transform of the PSF, h, and vice versa. These two equations can be found in Eq. (6) and Eq. (7).

Table 4: Aberration coefficients using the Cholesky matrix and multiplying it by the zero-mean, unit variance random vector.

a_j	
1	6.490
2	1.551
3	-0.031
4	0.348
5	-0.100
6	-0.079
7	0.156
8	0.354
9	0.356

$$H\left(f_x, f_y\right) = \mathscr{F}\left\{h\left(x, y\right)\right\} \tag{6}$$

$$h\left(x, y\right) = \mathscr{F}^{-1}\left\{H\left(f_x, f_y\right)\right\} \tag{7}$$

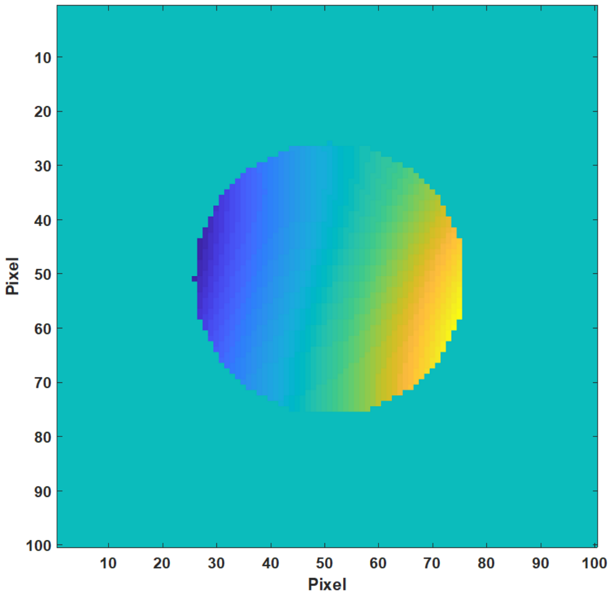

Figure 2: Image of the pupil function with an atmospheric phase screen using the aberration coefficients and Zernike polynomials.

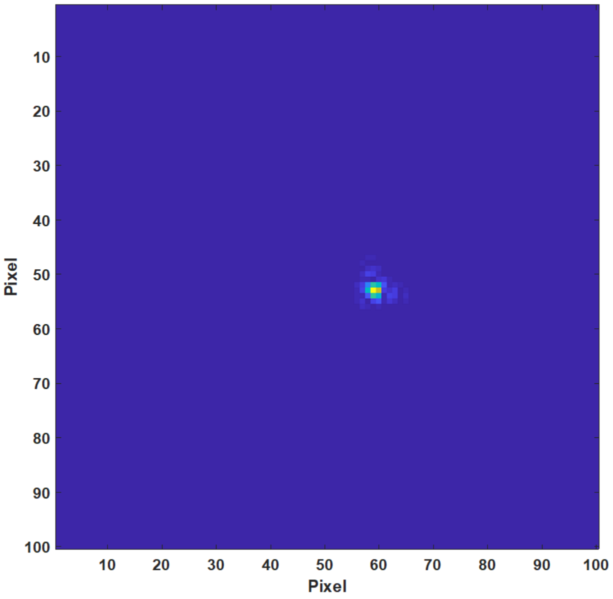

Figure 3: PSF of the example atmospheric phase screen.

2.3 Gerchberg-Saxton Phase Retrieval

There is an unknown phase associated with the unknown PSF. To estimate a spatially separable object or objects, this phase needs to be estimated in order to estimate the PSF. This can be done using methods such as the steepest-descent method, error-reduction algorithm and the one used for this research, the Gerchberg-Saxton phase retrieval algorithm [9, 10, 11]. In [11], the different algorithms mentioned above are compared and explained.

The premise behind the Gerchberg-Saxton phase retrieval approach is that the amplitude in the pupil and detector planes are known, but the phases are unknown. In Algorithm 1, the process describes how a guess at the phase in the detector plane is used to estimate the aperture field via the inverse Fourier transform and vice versa. In this algorithm j is the iteration count, F is the Fourier transform and F^{-1} is the inverse Fourier transform, both of these transforms are two-dimensional. The e is the exponential function and i is the imaginary number $\sqrt{-1}$. A guess of the phase at the detector is used for the initialization step of the Gerchberg-Saxton phase retrieval algorithm. A guess of the aperture is also selected for the initialization step of the phase retrieval algorithm. These guesses are explained in more detail in Chapter III.

Algorithm 1: Gerchberg-Saxton Phase Retrieval

```
Result: PSF = $|\text{Detector Field}|^2$
$j = 1$
while $j \leq 10$ do
    Detector Field $= F\left\{\text{aperture} \cdot \mathrm{e}^{-i \cdot \text{phase}}\right\}$;
    Detector Phase $=$ angle$\{\text{Detector Field}\}$;
    Aperture Field $= F^{-1}\left\{PSF \cdot e^{-i \cdot \text{Detector Phase}}\right\}$;
    Phase $=$ angle$\{\text{Aperture Field}\}$;
    $j = j + 1$;
end
```

2.4 Generalized Expectation-Maximization

The generalized expectation-maximization (GEM) algorithm is a technique to numerically optimize a maximum-likelihood estimation problem [6, 12]. The use of the GEM algorithm is to compute the maximum-likelihood estimates, through iterations, when the observed data is incomplete. This incomplete data is data that has a many to one mapping to the true object and associated PSF [13]. In this case the observed data is measured by the CCD or photo-diode. The problem of estimating the true object from this data is ill-posed because there are many combinations of objects and PSFs that can produce the measured data. Thus, instead of solving for the object, this research endeavors to solve for a spatially separable object. With the two main steps for the expectation-maximization (EM) algorithm consisting of an expectation step and a maximization step, an update equation can be derived to develop an iterative estimation solution for the ill-posed problem. Further examples of the derivation and discussion of the EM algorithm and the likelihood and convergence of the algorithm can be found in [13].

2.5 Blind Deconvolution

Deconvolution is the ability to reconstruct an object or PSF based on the knowledge of the PSF or object, respectively. Due to the complexity of convolution, the derivation of the reconstruction is done in the frequency domain to eliminate the convolution, described in Eq. (8), where o is the object, h is the PSF and i is the image or data . This type of direct deconvolution requires information about two of the three unknowns found in the convolution portion of Eq. (8). Typically the knowledge of the data is known and with the knowledge of the PSF, the object can be reconstructed. This is unable to be done if there are too many unknowns, such as when the object and the PSF are unknown. Then blind deconvolution is used to estimate the PSF and object.

$$i\left(z,w\right)=\sum_{x}\sum_{y}o\left(x,y\right)h\left(z-x,w-y\right)\quad\Leftrightarrow\quad I\left(f\right)=O\left(f\right)H\left(f\right) \tag{8}$$

Blind deconvolution is the ability to estimate an object without the knowledge of the object nor the PSF due to atmospheric turbulence or aberration from the telescope. There are several techniques used to perform blind deconvolution, such as maximum a posteriori (MAP) estimation, which is an iterative approach [14]. A non-iterative approach to blind deconvolution is the APEX method but lacks the ability to blindly deconvolve a more complex blur function that is incorporated due to atmospheric turbulence and other photon noise [15].

This research is based on the two-dimensional blind deconvolution algorithm originally derived by Dr. Timothy Schulz for astronomical images using multiple sets of data [6]. This research is an iterative approach that is similar to MAP called maximum-likelihood estimation.

2.6 Signal-to-Noise Ratio

Signal-to-noise ratio (SNR) is a ratio of the signal intensity versus the background noise. With a low SNR value it becomes difficult to properly estimate an object. To determine the SNR of a particular data set, the power of the signal is divided by the power of the noise. SNR can be calculated by also dividing the amplitudes of the signal and power and squaring it as seen in Eq. (9), where P_{signal} and P_{noise} are the power of the signal and noise, respectively. Additionally, A_{signal} and A_{noise} are the amplitude of the signal and noise, respectively.

$$SNR = \frac{P_{signal}}{P_{noise}} = \left(\frac{A_{signal}}{A_{noise}}\right)^2 \tag{9}$$

SNR can also be calculated as the average signal vs the standard deviation of the signal, seen in Eq. (10), where μ_S is the average of the signal and σ_S is the standard deviation of the signal.

$$SNR = \frac{\mu_S}{\sigma_S} \tag{10}$$

2.7 Relationship of Complete Data Given Incomplete Data

The two-dimensional blind deconvolution algorithm utilizes statistical definitions to determine the probability of complete data given the incomplete data. The incomplete data is the sum of all the complete data and the complete data is indirectly observed with a many-to-one mapping of the complete data to the incomplete data. This is done by proving that the probability of complete data given the incomplete data is a binomial distribution and has an average or expected value of Np, where N is the number of trials and p is the probability of success on any given trial. Using Bayes theorem, the probability of the complete data given the incomplete data can be written as seen in Eq. (11), where d_k is the incomplete data, $\tilde{d}_k$ is the complete data, (z, w) are the detector plane coordinates and (x, y) are the object plane coordinates. The P denotes the probability notation.

$$P\left(\tilde{d}_k\left(z, w|x, y\right)|d_k\left(z, w\right)\right) = \frac{P\left(\tilde{d}_k\left(z, w|x, y\right) \cap d_k\left(z, w\right)\right)}{P\left(d_k\left(z, w\right)\right)} \tag{11}$$

The first step is to solve for the probability of the incomplete data. It is known that the expected value of the incomplete data is the image itself, described in Eq. (12). This distribution follows a Poisson distribution due to how photons arrive to the detector,as seen in Eq. (13).

$$E\left[d_k\left(z, w\right)\right] = i_k\left(z, w\right) = \sum_x \sum_y o\left(x, y\right) h_k\left(z - x, w - y\right) \tag{12}$$

$$P\left(d_k\left(z, w\right)\right) = \frac{\left(\sum\limits_x \sum\limits_y o\left(x, y\right) h_k\left(z - x, w - y\right)\right)^{d_k(z,w)} e^{-\sum\limits_x \sum\limits_y o(x,y) h_k(z-x,w-y)}}{d_k\left(z, w\right)!} \tag{13}$$

To solve for the probability of the intersection in the numerator of Eq. (11), the relationship between the complete and incomplete data in Eq. (14) is used.

$$\begin{gathered} d_1 = \tilde{d}_k\left(z, w|x_0, y_0\right) \\ d_2 = \sum_{x \neq x_0} \sum_{y \neq y_0} \tilde{d}_k\left(z, w|x, y\right) \\ d_k\left(z, w\right) = d = d_1 + d_2 \end{gathered} \tag{14}$$

From the two dimensional blind deconvolution algorithm the expected value is determined for d_1 and d_2, which is found in Eq. (15).

$$
\begin{aligned}
E\left[d_1\right] &= m_1 = o\left(x_0, y_0\right) h_k\left(z - x_0, w - y_0\right) \\
E\left[d_2\right] &= m_2 = \sum_{x \neq x_0} \sum_{y \neq y_0} o\left(x, y\right) h_k\left(z - x, w - y\right)
\end{aligned}
\tag{15}
$$

The probability of the intersection of d_1 and d_2 is found in Eq. (16).

$$
P\left(d_1 \cap d_2\right) = \frac{{m_1}^{d_1} e^{-m_1}}{d_1!} \frac{{m_2}^{d_2} e^{-m_2}}{d_2!}
\tag{16}
$$

The probability found in Eq. (16) can be rewritten as a function of d_1 and d and is seen in Eq. (17).

$$
\begin{gathered}
d_2 = d - d_1 \\
P\left(d_1 \cap d - d_1\right) = P\left(d_1 \cap d\right) = \frac{{m_1}^{d_l} e^{-m_1}}{d_1!} \frac{{m_2}^{d - d_1} e^{-m_2}}{\left(d - d_1\right)!}
\end{gathered}
\tag{17}
$$

The original probability of the incomplete data can be rewritten using the notation from Eq. (17) which leads to Eq. (18).

$$
P\left(d_k\left(z, w\right)\right) = P\left(d\right) = \frac{\left(m_1 + m_2\right)^d e^{-\left(m_1 + m_2\right)}}{d!}
\tag{18}
$$

Lastly, solving for the original conditional probability and simplifying the probability is found to follow a binomial distribution shown in Eq. (19).

$$
\frac{P\left(d_1 \cap d\right)}{P\left(d\right)} = \frac{d!}{d_1!\left(d - d_1\right)!} \left(\frac{m_1}{m_1 + m_2}\right)^{d_1} \left(\frac{m_2}{m_1 + m_2}\right)^{d - d_1}
\tag{19}
$$

III. Methodology

3.1 Overview

This chapter deals with the one-dimensional blind deconvolution algorithm to estimate a spatially separable object and the data set's PSF. This chapter discusses how to determine a spatially separable object based on a visual and numerical approach. Next, the expectation maximization steps are discussed and derived to produce the new update equations for a spatially separable object. Then, the algorithm is developed and discussed to describe how the update equations produce new estimates from old ones. Finally, the generation of a receiver operating characteristic (ROC) curve is discussed and how it was generated for simulated data. The results of simulated and laboratory data are found in Chapter IV. These results include the effects of SNR for both simulated and lab generated data, as well as a ROC curve for these data sets.

3.2 Spatially Separable Object

This section will describe how an object can be spatially separated. This section will also discuss the two ways that were utilized to determine if an object is spatially separable or not. These two ways of determining spatial separability are by visual reconstruction and singular value decomposition (SVD).

3.2.1 Object Separation

A spatially separable object can be separated into a product of it's horizontal and vertical components. This is expressed in Eq. (20), where $o\left(x,y\right)$ is the object, $o_1\left(x\right)$ is the horizontal component of $o\left(x,y\right)$ and $o_2\left(y\right)$ is the vertical component. In this research, separating the object is accomplished by summing the object along the columns then once again along the rows. Summing along the columns will produce

$o_1(x)$ and summing along the rows will produce $o_2(y)$. All image matrices can be separated using this technique, though you lose detailed characteristics about the object when reconstructing the object if the object is not spatially separable.

$$o\left(x,y\right)=o_1\left(x\right)o_2\left(y\right) \tag{20}$$

As seen in Figure 4, the simplicity of a binary star system makes it a good scenario to perform the one-dimensional blind deconvolution algorithm on. This separation can be seen in the stem plots in Figure 5. This can be easily reconstructed back to the truth image as seen before. If the object isn't spatially separable, this scenario can be seen in Figure 6, reconstructing the object back into the original form doesn't produce the correct object. Thus, this limits the ability to perform the one-dimensional blind deconvolution algorithm to objects that can be spatially separated into their respective x and y components. This would be objects such as a binary star system, point sources and simple objects such as squares and other quadrilaterals.

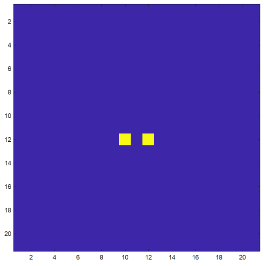

Figure 4: A spatially separable object, binary star system.

3.2.2 Singular Value Decomposition

A test can be performed on the object to determine whether the object is truly separable or not. This test consists of performing an singular value decomposition (SVD) on $o\left(x,y\right)$ [16]. The two objects that will be tested for separability using this method are found in Figure 4 and Figure 6.

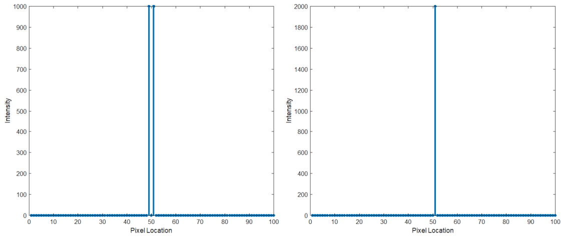

Figure 5: Stem plot of the binary star system that has been separated into two components.

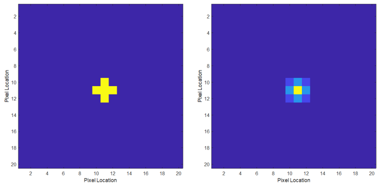

Figure 6: Left:True object; Right: Reconstruction of separated object

The results of these two scenarios can be found in Table 5. In this table the SVD was taken and the main diagonal was converted to a vector to easily verify the separability of the object. As can be seen in the table, the binary star system is a separable object due to the the predominant value in the first row followed by zeros in the remaining rows. This describes the object as only needing 1 value to reconstruct the object. Unlike in the cross shaped object, there is a requirement of two values to reconstruct the object correctly. Only using the 1 SVD value for the object in Fig. 6 will produce similar results due to the complete description of the object is missing.

Table 5: SVD vectors for the binary star system and cross configuration.

i^{th} Singular Value	Binary	Cross
1	1414.21	2000.00
2	0	1000.00
3	0	0
$\vdots$	$\vdots$	$\vdots$
N	0	0

3.3 Expectation-Maximization Algorithm

The EM algorithm is one approach to solve for the ill-posed problem of an unknown object and an unknown PSF introduced from the atmosphere and telescope aberrations. This approach allows each iteration to establish a more refined estimation of the object and it's PSF. With the statistical expected value and maximization of the complete data, a likelihood estimation of the object and PSF can be calculated. There are two main steps for the EM approach called the expectation step and the maximization step or e-step and m-step, respectively [13]. Other blind deconvolution EM algorithms for astronomical data can be found in [17, 6]. The steps and derivation found in the following subsections outline how this new blind deconvolution algorithm is performed.

3.3.1 EM Steps

The new one-dimensional blind deconvolution algorithm requires a new set of statistical models and equations in order to perform the algorithm properly. Below is a list of steps that are required to derive the update equations for the one-dimensional blind deconvolution algorithm.

1. Statistical Model for Incomplete Data

2. Statistical Model for Complete Data

3. Generate Probability Function for Incomplete Data

4. Generate log-likelihood function

5. Derive expectation step

6. Maximize expectation step

7. Solve for Expected Value of Incomplete Data given Complete Data

8. Solve for Update Equations

3.3.2 Statistical Model for Incomplete Data

The incomplete data is the sum of all the complete data and is mathematically described in Eq. (21). Where d_k describes the incomplete data, $\tilde{d}_k$ describes the complete data, (z, w) is the coordinates in the detector plane, and lastly (x, y) is the coordinates in the object plane.

$$d_k\left(z, w\right) = \sum_x \sum_y \tilde{d}_k\left(z, w|x, y\right) \tag{21}$$

To describe the incomplete data more accurately there needs to be a statistical description of the incomplete data. The expected value of the incomplete data is the image intensity detected for that frame of data, shown in Eq. (22). Where i_k is the image intensity and is the convolution of the spatially separable objects and it's respective PSF for that frame, described in Eq. (23), where h_k is the PSF of that specific image or frame of data and E is the expected value operator.

$$E\left[d_k\left(z, w\right)\right] = i_k\left(z, w; o_1, o_2, h_k\right) \tag{22}$$

$$i_k\left(z, w; o_1, o_2, h_k\right) = \sum_x \sum_y o_1\left(x\right) o_2\left(y\right) h_k\left(z - x, w - y\right) \tag{23}$$

3.3.3 Statistical Model for Complete Data

The complete data is the data that is observed indirectly due to the incomplete data and has a many-to-one mapping of complete data to incomplete data [13]. The complete data can be described to be Poisson-distributed random variables and have a mean expressed in Eq. (24).

$$E\left[\tilde{d}_k\left(z,w|x,y\right)\right]=o_1\left(x\right)o_2\left(y\right)h_k\left(z-x,w-y\right) \tag{24}$$

3.3.4 Generate Probability Function for Incomplete Data

It is assumed that the incomplete data is a Poisson distributed at every point in the detector plane and is statistically independent between different pixels and frames. This produces a probability function for a single pixel of a frame of data, found in Eq. (25).

$$P\left[d_k\left(z,w\right)\right]=\frac{i_k(z,w;o_1,o_2,h_k)^{d_k(z,w)}e^{-i_k(z,w;o_1,o_2,h_k)}}{d_k\left(z,w\right)!} \tag{25}$$

The total probability function for the incomplete data is the product of the individual incomplete data probabilities, due to the independence between each pixel and frame. This total probability function can be found in Eq. (26).

$$P\left[d\left(z,w\right)\right]=\prod_k\prod_z\prod_w\frac{i_k(z,w;o_1,o_2,h_k)^{d_k(z,w)}e^{-i_k(z,w;o_1,o_2,h_k)}}{d_k\left(z,w\right)!} \tag{26}$$

3.3.5 Generating Log-Likelihood Function

Now that the total probability function for the incomplete data has been computed, the natural log of the function is taken. This is done in order to make the function easier to maximize by removing the products and replacing them with sums. This log-likelihood function is found in Eq. (27), where L represents the log-likelihood function.

$$
\begin{aligned}
L\left(o_1, o_2, h_k\right) = \sum_k \sum_z \sum_w d_k\left(z, w\right) \ln\left[i_k\left(z, w; o_1, o_2, h_k\right)\right] - \cdots \\
\cdots - i_k\left(z, w; o_1, o_2, h_k\right) - \ln\left[d_k\left(z, w\right)!\right]
\end{aligned}
\tag{27}
$$

$$
L\left(o_1, o_2, h_k\right) = \sum_k \sum_z \sum_w d_k\left(z, w\right) \ln\left[i_k\left(z, w; o_1, o_2, h_k\right)\right] - i_k\left(z, w; o_1, o_2, h_k\right) \tag{28}
$$

The data factorial at the end of the log-likelihood equation has no affect on the maximization step due to no dependence on the object or PSF. Due to this Eq. (27) is simplified to Eq. (28). This summarizes the log-likelihood of the incomplete data. Similar to this mathematical model, the complete data log-likelihood is shown in Eq. (29). The image intensity was also substituted in to complete the equation. The notation L^{CD} denotes the complete data log-likelihood function.

$$
\begin{aligned}
L^{CD}\left(o_1, o_2, h_k\right) = \sum_k \sum_z \sum_w \sum_x \sum_y \tilde{d}_k\left(z, w|x, y\right) \left\{\ln\left[o_1\left(x\right)\right] + \ln\left[o_2\left(y\right)\right] + \cdots \right. \\
\left. \cdots + \ln\left[h_k\left(z-x, w-y\right)\right]\right\} - o_1\left(x\right) o_2\left(y\right) h_k\left(z-x, w-y\right)
\end{aligned}
\tag{29}
$$

3.3.6 Derive Expectation Step

Now that the log-likelihood function has been computed, the expected value of the log-likelihood function can be taken, shown in Eq. (30). The expected value of the log-likelihood function is denoted as $Q\left(o_1, o_2, h\right)$, seen in Eq. (31). The expected value of the complete data given the incomplete data will be calculated in a later section.

$$Q\left(o_1, o_2, h_k\right) = E\left[L^{CD}\left(o_1, o_2, h_k\right)|d_k\left(z, w\right)\right] \tag{30}$$

$$\begin{aligned} Q\left(o_1, o_2, h_k\right) = \sum_k \sum_z \sum_w \sum_x \sum_y E\left[\tilde{d}_k\left(z, w|x, y\right)|d_k\left(z, w\right)\right]\left\{\ln\left[o_1\left(x\right)\right] + \cdots \right. \\ \left. \cdots + \ln\left[o_2\left(y\right)\right] + \ln\left[h_k\left(z-x, w-y\right)\right]\right\} - o_1\left(x\right) o_2\left(y\right) h_k\left(z-x, w-y\right) \end{aligned} \tag{31}$$

3.3.7 Maximize Expectation Step

With the expectation step complete, the maximization step (m-step) can be computed by taking the derivative of the expectation step function with respect to each of the components that are to be solved, o_1, o_2, and h_k and setting them equal to zero. The first derivative that will be taken is the object focused around the x component, $o_1\left(x\right)$, found in Eq. (32). The m-step also focuses around only one point. In this computation, x_0, is used as the singular point. The same goes for the y component, y_0.

$$\begin{aligned} \frac{dQ\left(o_1, o_2, h_k\right)}{do_1\left(x_0\right)} = \sum_k \sum_z \sum_w \sum_y \frac{E\left[\tilde{d}_k\left(z, w|x_0, y\right)|d_k\left(z, w\right)\right]}{o_1\left(x_0\right)} - \cdots \\ \cdots - o_2\left(y\right) h_k\left(z-x_0, w-y\right) = 0 \end{aligned} \tag{32}$$

In order to simplify this into an algorithm that will perform correctly, the use of a Lagrange multiplier is required. Thus the assumption in Eq. (33) and the property of the PSF found in Eq. (34) are used to simplify the equation for the new $o_1\left(x_0\right)$. This simplifies Eq. (32) to Eq. (35), where K denotes the number of data sets or image frames that are used for the algorithm. The exponential *new* notation signifies the update equation that will be overwritten after each iteration of the algorithm.

$$\sum_{y} o_2\left(y\right) = 1 \tag{33}$$

$$\sum_{z}\sum_{w} h_k\left(z - x_0, w - y\right) = 1 \tag{34}$$

$$o_1^{new}\left(x_0\right) = \sum_{k}\sum_{z}\sum_{w}\sum_{y} \frac{E\left[\tilde{d}_k\left(z, w|x, y\right)|d_k\left(z, w\right)\right]}{K} \tag{35}$$

For maximizing the object separated into it's y component the Lagrange multiplier is incorporated into the equation due to the assumption made earlier. This gets the following equation found in Eq. (36). Solving for $o_2\left(y_0\right)$ gives Eq. (37). The variable γ is introduced to ensure that $o_2\left(y_0\right)$ sums to 1. In Eq. (38)-(41), the variable γ is solved in terms of the expected value of the incomplete data.

$$\frac{dQ\left(o_1, o_2, h_k\right)}{do_2\left(y_0\right)} = \sum_{k}\sum_{z}\sum_{w}\sum_{x} \frac{E\left[\tilde{d}_k\left(z, w|x, y_0\right)|d_k\left(z, w\right)\right]}{o_2\left(y_0\right)} - \gamma = 0 \tag{36}$$

$$o_2^{new}\left(y_0\right) = \frac{\sum\limits_{k}\sum\limits_{z}\sum\limits_{w}\sum\limits_{x} E\left[\tilde{d}_k\left(z, w|x, y_0\right)|d_k\left(z, w\right)\right]}{\gamma} \tag{37}$$

$$\sum_{k}\sum_{z}\sum_{w}\sum_{x} \frac{E\left[\tilde{d}_k\left(z, w|x, y\right)|d_k\left(z, w\right)\right]}{o_2\left(y\right)} - \gamma = 0 \tag{38}$$

$$\gamma o_2\left(y\right) = \sum_{k}\sum_{z}\sum_{w}\sum_{x} E\left[\tilde{d}_k\left(z, w|x, y\right)|d_k\left(z, w\right)\right] \tag{39}$$

$$\gamma \sum_{y} o_2\left(y\right) = \sum_{k}\sum_{z}\sum_{w}\sum_{x}\sum_{y} E\left[\tilde{d}_k\left(z, w|x, y\right)|d_k\left(z, w\right)\right] \tag{40}$$

$$\gamma = \sum_{k}\sum_{z}\sum_{w}\sum_{x}\sum_{y} E\left[\tilde{d}_k\left(z, w|x, y\right)|d_k\left(z, w\right)\right] \tag{41}$$

The last portion of the maximization step is solving for the PSF. Using a property of convolution the shift aspect of the convolution was transferred to the objects to simplify the calculations of the new PSF. This new update equation for the PSF can be found in Eq. (43), where the notation "*old*" denotes the current estimate of that particular variable.

$$\frac{dQ\left(o,h\right)}{dh\left(x_0,y_0\right)}=\sum_z\sum_w\frac{E\left[\tilde{d}\left(z,w|x_0,y_0\right)|d\left(z,w\right)\right]}{h\left(x_0,y_0\right)}-o_1\left(z-x_0\right)o_2\left(w-y_0\right)=0 \quad (42)$$

$$h^{new}\left(x_0,y_0\right)=\frac{\sum\limits_z\sum\limits_w E\left[\tilde{d}\left(z,w|x_0,y_0\right)|d\left(z,w\right)\right]}{\sum\limits_z\sum\limits_w o_1^{old}\left(z-x_0\right)o_2^{old}\left(w-y_0\right)} \quad (43)$$

3.3.8 Solve for Expected Value of Complete Data given Incomplete Data

The expected value of the complete data given the incomplete data is the expected value of a binomial distributed random variable. As discussed in Chapter II, the mean or expected value is equal to $N \cdot p$. The Eq. (44)-(48) establishes the values of the individual variables while Eq. (49) states the expected value of the complete data given the incomplete data. The full derivation has been completed in Chapter II for the binomial distribution random variable for complete data given incomplete data.

$$E\left[\tilde{d}_k\left(z,w|x,y\right)|d_k\left(z,w\right)\right]=\mu=Np \tag{44}$$

$$N=d_k\left(z,w\right) \tag{45}$$

$$p=\frac{m_1}{m_1+m_2} \tag{46}$$

$$m_1=o_1\left(x_0\right)o_2\left(y_0\right)h_k\left(z-x_0,w-y_0\right) \tag{47}$$

$$m_2=\sum_{x\neq x_0}\sum_{y\neq y_0}o_1\left(x\right)o_2\left(y\right)h_k\left(z-x,w-y\right) \tag{48}$$

$$E\left[\tilde{d}_k\left(z,w|x,y\right)|d_k\left(z,w\right)\right]=d_k\left(z,w\right)\frac{o_1\left(x_0\right)o_2\left(y_0\right)h_k\left(z-x_0,w-y_0\right)}{\sum\limits_x\sum\limits_y o_1\left(x\right)o_2\left(y\right)h_k\left(z-x,w-y\right)} \tag{49}$$

3.3.9 Solve for Update Equations

With the knowledge of the expected value of the complete data given the incomplete data, the update equations can be solved. Simply replacing the expected value with the value that was found in Eq. (49), will generate the new update equation. The new update equations are found in Eq. (50)-(53).

$$o_1^{new}\left(x\right)=\frac{\sum\limits_k\sum\limits_z\sum\limits_w\sum\limits_y\frac{d_k(z,w)o_1^{old}(x)o_2^{old}(y)h_k^{old}(z-x,w-y)}{i_k^{old}(z,w)}}{K} \tag{50}$$

$$o_2^{new}\left(y\right)=\frac{\sum\limits_k\sum\limits_z\sum\limits_w\sum\limits_x\frac{d_k(z,w)o_1^{old}(x)o_2^{old}(y)h_k^{old}(z-x,w-y)}{i_k^{old}(z,w)}}{\gamma} \tag{51}$$

$$\gamma=\sum_k\sum_z\sum_w d_k\left(z,w\right) \tag{52}$$

$$h_k^{new}\left(z-x,w-y\right)=\frac{\sum\limits_z\sum\limits_w\frac{d(z,w)o_1^{old}(z-x_0)o_2^{old}(w-y_0)h^{old}(x_0,y_0)}{i^{old}(z,w)}}{\sum\limits_z\sum\limits_w o_1^{old}\left(z-x_0\right)o_2^{old}\left(w-y_0\right)} \tag{53}$$

These three equations are the new update equations that will be used through several iterations to estimate the object and PSF of the incomplete data.

3.4 Phase Retrieval

With the EM update equations derived for the one-dimensional spatially separable object, the next step is to perform a phase retrieval in order to reconstruct an estimation of the PSF. The phase retrieval method that is used in this research is the Gerchberg-Saxton phase retrieval algorithm. In the one-dimensional blind deconvolution algorithm designed, the PSF is not assumed to be separable and follows closely with the phase retrieval in [11]. The number of iterations to accurately estimate the PSF is important due to the amount of time this portion of the algorithm takes. After running the algorithm one hundred times, the Gerchberg-Saxton phase retrieval algorithm took 52% of the total time to estimate phase for the PSF using only 10 iterations of the phase retrieval algorithm. The total number of iterations of 10 is used throughout all of the research as anything more than 10 significantly increases run-time of the algorithm with only minor performance increase for the estimation of the PSFs. A visual representation of the total time is found in Fig. 7, which shows the percentage of time required to run the Gerchberg-Saxton phase retrieval algorithm with 10 iterations for phase retrieval and 750 iterations of the one-dimensional blind deconvolution algorithm, totaling 7500 iterations for phase retrieval. The time outside the Gerchberg-Saxton algorithm is the remaining setup and object estimation using the PSFs.

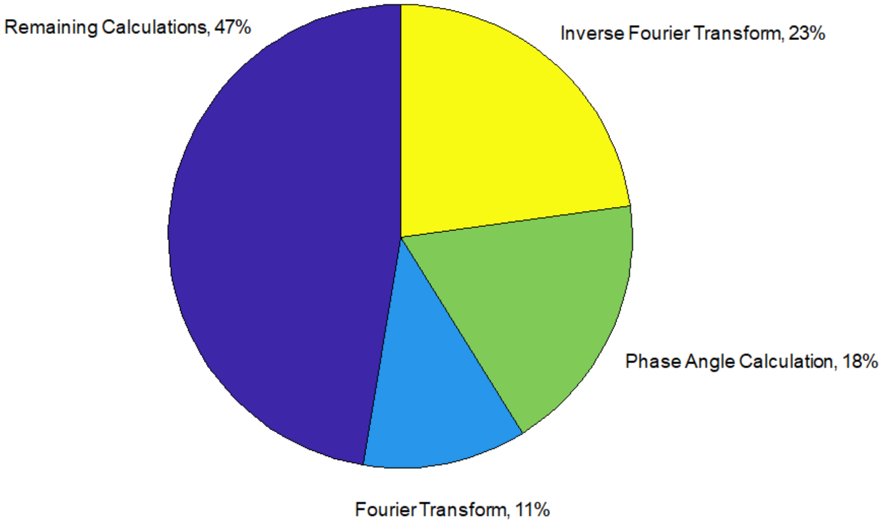

Figure 7: Pie chart of how much time it takes to execute the Gerchberg-Saxton phase retrieval algorithm with 10 iterations.

3.5 One-Dimensional Blind Deconvolution Algorithm

The final step to begin estimating the spatially separable objects and the different PSFs for each respective frame of data is to setup the algorithm with the EM update equations and the Gerchberg-Saxton phase retrieval algorithm. The first step is to take a guess at the initial values for the object and the two PSFs. These initial values are needed due to the update equations requiring a previous iterations values. For this research, to include the simulated data and the lab created data, all the initial values were kept constant throughout the different runs of the algorithm. The guess for the object to start with was a uniform value of ones the size of the frames of data. These objects were scaled based on the update equations to ensure that $o_2\left(y\right)$ summed to 1. The guess of the PSFs could be a uniform number throughout the frame of data. This was not as useful as taking a logical guess of the PSF, by saying that it is a parabolic focus error. This parabolic focus error can be seen in Fig. 8.

With all variables initialized, the algorithm is designed to estimate the object and

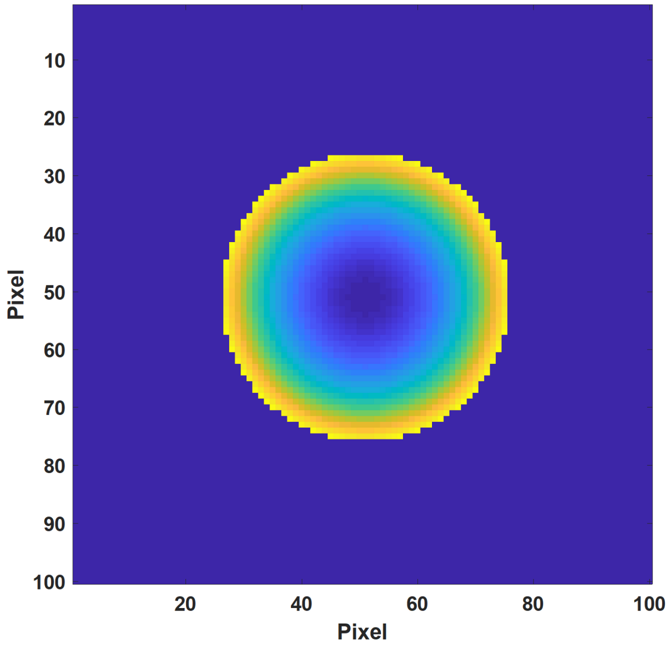

Figure 8: The guess at the focus error for the initialization of the algorithm.

PSF. The flowchart for the algorithm can be found in Fig. 9. The algorithm starts by reading in the frames of data from frame 1 to frame K. Then guesses for the object and all the PSFs are chosen, as discussed in the previous paragraph. With the guess of the object and the PSFs, the new estimated object is calculated based on these guesses using the EM approach. With the new object calculated, the Gerchberg-Saxton phase retrieval algorithm is performed to generate the new PSFs for the blind deconvolution algorithm. This new object and PSF is ran back through the algorithm again to obtain a new object based on the updated object and PSFs. After several iterations of this algorithm, the object and PSFs converge to a estimation of the object or objects and PSFs.

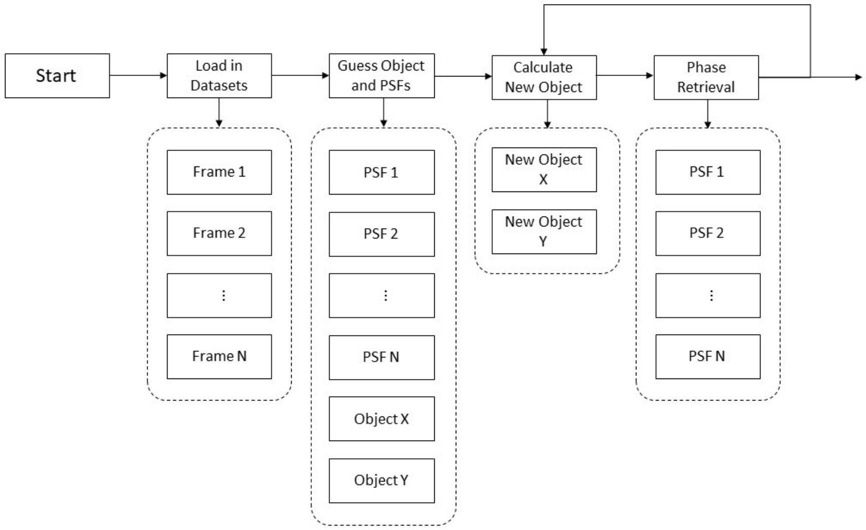

Figure 9: Flowchart for the one-dimensional blind deconvolution algorithm.

IV. Results

This chapter explains the results of several different scenarios utilizing the one-dimensional blind deconvolution algorithm. The chapter also explains the resulting error between the actual object and the estimated object and compares those results to the two-dimensional blind deconvolution algorithm. Results are shown for objects that are more complex but are still spatially separable. The last set of data that was ran through the one-dimensional algorithm is a set of data that was collected in a laboratory setting. Lastly, the chapter concludes with a discussion of what factors can affect the algorithms performance.

4.1 Simulated High and Low Signal-to-Noise Ratio

The ability for the algorithm to perform well in both high and low SNR is required due to the nature of astronomical data. Objects whether they are in LEO, geosynchronous orbit or light years away can have differing SNR. Objects in LEO can have a low SNR due to reflectively of the object and the angle of the sun with respect to the object and view point through the telescope. This can cause objects that are light-years away to have a significantly higher SNR than these objects in LEO. More information on SNR can be found in Chapter II. This section, will discuss how the data was collected and analyze the results of the simulation for both scenarios, low and high SNR. The diameter versus Fried's seeing parameter for the atmospheric turbulence was a factor of four times greater for the simulation.

4.1.1 High Signal-to-Noise Ratio

In this scenario, the setup for the simulation was to convolve a binary star system with a random PSF. This binary star system has an amplitude that is an order of magnitude larger than the standard deviation of the noise. The PSF that the binary star system was convolved with is found in Figure 10. The resulting frames of data from the convolution are found in Figure 11. The peak SNR for the high SNR simulation for the two data sets that were used was 6.25.

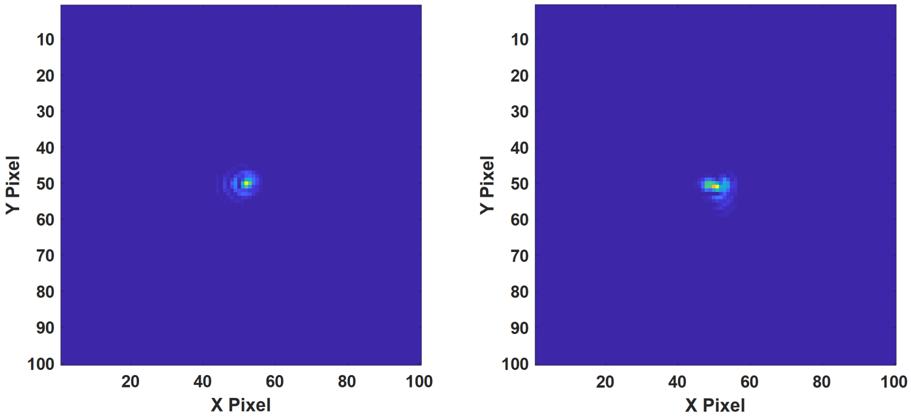

Figure 10: Two simulated PSFs to be convolved with binary star system.

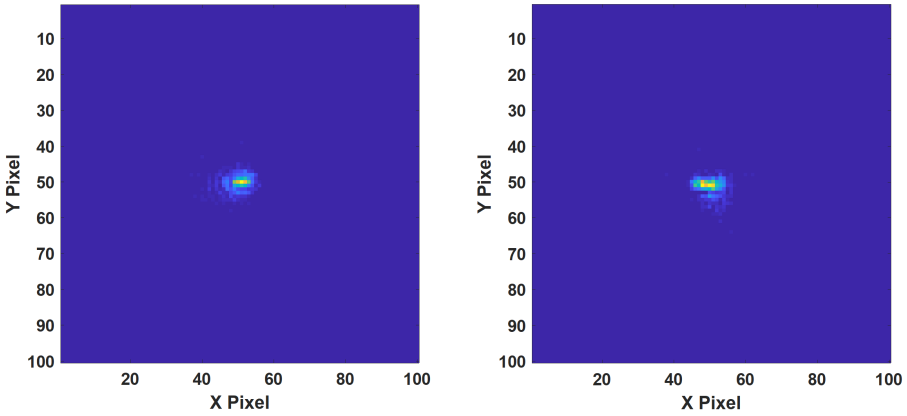

Figure 11: Two simulated frames of data from simulated PSFs.

Running the two frames of data through the one-dimensional blind deconvolution algorithm generates results that are similar to results from the original two-dimensional algorithm. This blind deconvolution estimation of the object, which has been magnified to elaborate on details, can be seen in Figure 12. To accompany the object estimation, the algorithm also estimates the PSF, which can be seen in Figure 13. Visually the two PSFs are very similar and may have minor differences but the minor differences in the PSF are not significant compared to the estimation of the object.

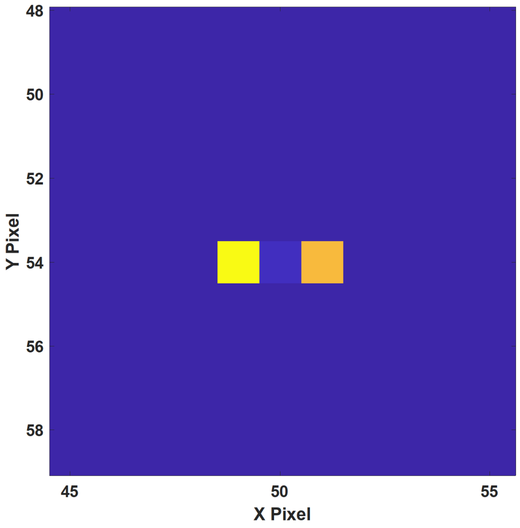

Figure 12: Zoomed in on object estimated using two frames of data following the one-dimensional algorithm.

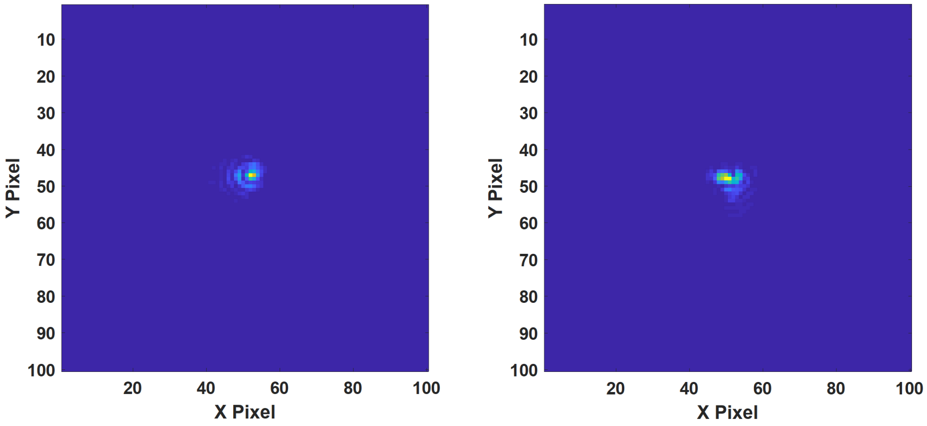

Figure 13: Estimation of the two PSF utilized for the high SNR simulation.

For the object reconstruction the percent relative error, using Eq. (54), of the left star of the binary star system is 37.5%. Which means that the intensity of the left binary star that was simulated was 37.5 dimmer or brighter than the reference for that particular star. The percent relative error of the right portion of the binary star system is 12.8% for the intensity of the right star compared to the reference. This demonstrates that the algorithm was able to accurately estimate the location and separation based on visual inspection and intensity of the object based on numerical calculations of the differences between the reference and the estimated objects.

$$\text{Error} = \left| \frac{\text{Measured - Actual}}{\text{Actual}} \right| * 100 \tag{54}$$

4.1.2 1-D vs 2-D Blind Deconvolution Algorithm

To determine the accuracy and performance of the new blind deconvolution algorithm the two algorithms are compared using the same data sets. In Figure 14, the absolute error of the two algorithms are plotted against each other to determine which one had the fewest errors in both intensity and resolution of the binary star. The x and y tilt of the atmospheric turbulence was ignored and the estimated objects were shifted to match the true object. To show how the two errors compare to each other they have been sorted based on either the one-dimensional algorithm or the two-dimensional algorithm. The trials are shifted concurrently, for example, trial 60 of the one-dimensional algorithm is compared to trial 60 of the two-dimensional algorithm. The left plot sorts the absolute errors of each trial from lowest error to greatest error of the one-dimensional algorithm. The right plot sorts the absolute errors of each trial from lowest error to greatest error of the two-dimensional algorithm. It is visually apparent that the one-dimensional algorithm performed better in both instances where the two-dimensional algorithm performed well or poorly.

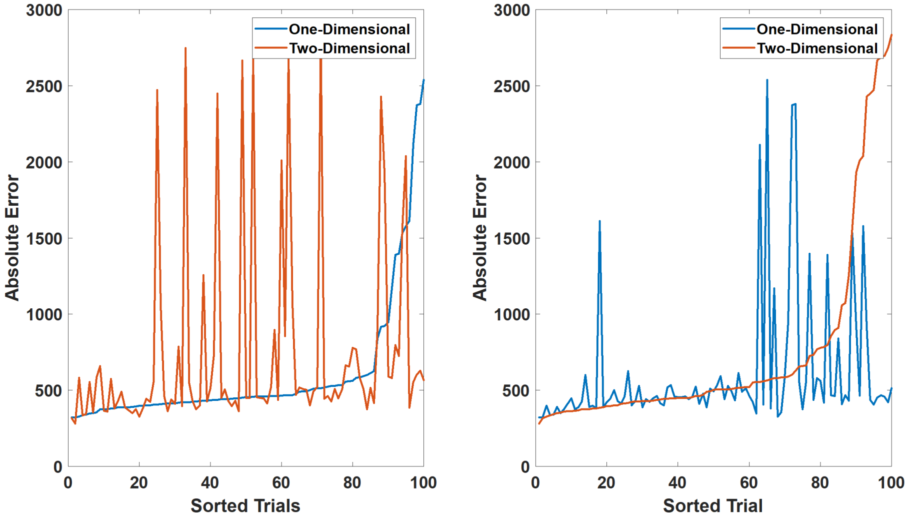

Figure 14: Left: Sorted plot from lowest to highest error of the one-dimensional algorithm using high SNR with the same two-dimensional algorithm's trial plotted against it; Right: Sorted plot from lowest to highest error of the two-dimensional algorithm using high SNR with the same one-dimensional algorithm's trial plotted against it

4.1.3 Low Signal-to-Noise Ratio

A similar simulation was performed where the binary star's intensity was significantly reduce. The average SNR for the low SNR simulation for the two data sets that were used was 3.63. In Figure 15, the absolute error of the two algorithms is plotted and clearly shows that the one-dimensional algorithm performs significantly better in low SNR scenarios.

Both errors for high SNR and low SNR are then averaged and are found in Table 6. Numerically it can also be seen that the one-dimensional blind deconvolution algorithm performs better than the two-dimensional algorithm, when testing for the error of the intensities based on the true data, in both high and low SNR situations.

Table 6: Error of the simulated results for both high and low SNR.

	Low SNR	High SNR
One-Dimensional	259.61	602.12
Two-Dimensional	329.57	741.55

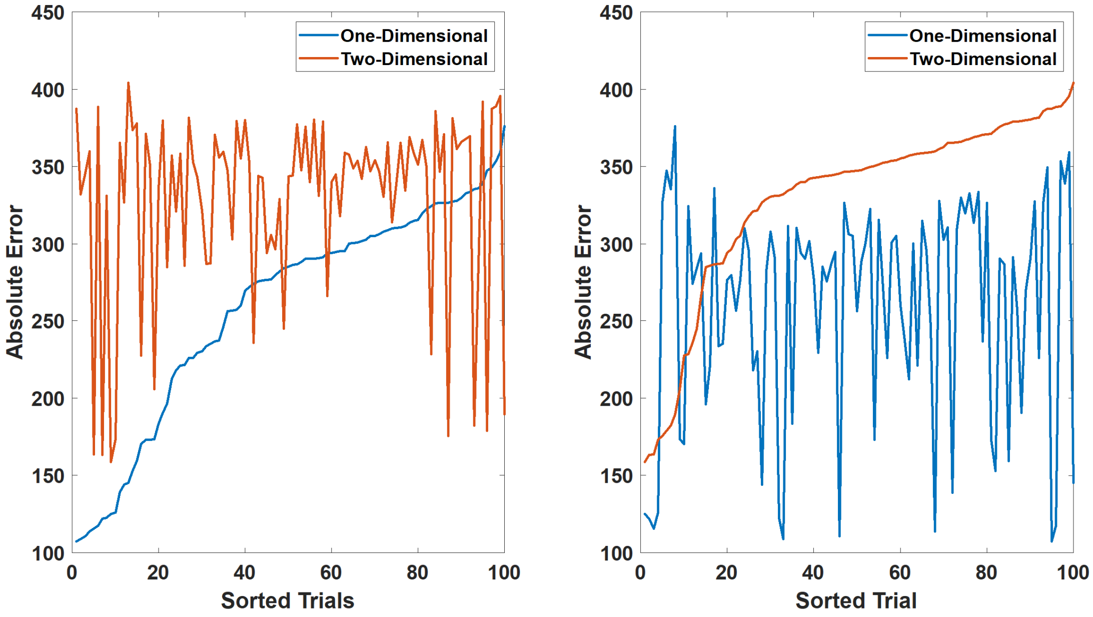

Figure 15: Left: Sorted plot from lowest to highest error of the one-dimensional algorithm using low SNR with the same two-dimensional algorithm's trial plotted against it; Right: Sorted plot from lowest to highest error of the two-dimensional algorithm using low SNR with the same one-dimensional algorithm's trial plotted against it

4.2 Blind Deconvolution of Other Spatially Separable Objects

The previous section concluded that the one-dimensional blind deconvolution algorithm was able to estimate the object of a binary star system through simulation. In the majority of cases, the one-dimensional blind deconvolution algorithm performed better than the two-dimensional blind deconvolution algorithm. Thus for the rest of the research the results will only display the unique results of the one-dimensional algorithm and no further comparisons. This section will discuss how the new algorithm performs when the object is not as simple as a binary star system. Through simulations, two additional objects were estimated using the one-dimensional algorithm to see how the algorithm would perform the estimation. These two objects are four point sources separated by one blank pixel in a square pattern and a bar the length of 5 pixels, these objects can be found in Figure 16 and Figure 17. The goal for these unique shapes is to display the capability of the one-dimensional algorithm to estimate the object, if the object is spatially separable, to have the correct spacing and pixel count.

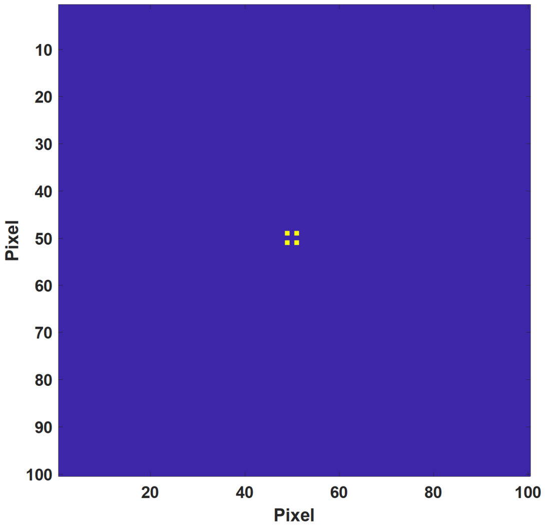

Figure 16: True image of an object consisting of 4 point sources that are separated by one pixel in a square pattern.

These objects were subjected to atmospheric turbulence similar to that of the simulated binary star simulation performed in the previous section. The data sets for the two scenarios in Figure 16 and Figure 17 are found in Figure 18 and Figure 19. Similar to the binary star system simulation, there is no visible evidence of what these objects will look like. Thus, the one-dimensional blind deconvolution is required to estimate these objects.

The one-dimensional blind deconvolution algorithm was able to successfully esti-

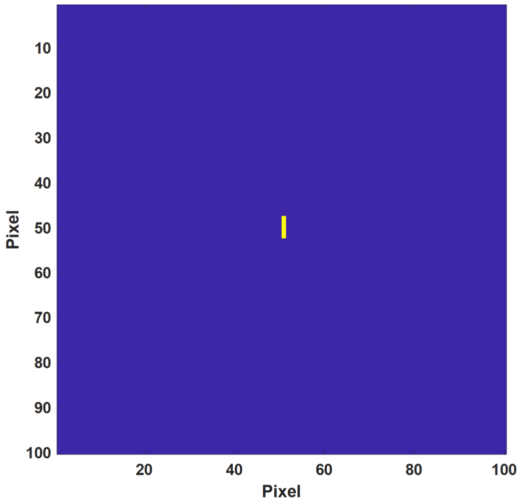

Figure 17: True image of an object consisting of 5 pixels orientated in a vertical line with no spatial separation between the different pixels.

mate what the four point source object would look like. Though the intensities of the four point sources differ from the true data, the shape and spacing of the objects match exactly what the truth describes. These results of the estimation of the four point sources can be found in Figure 20.

The last simulated spatially separable object that was estimated using the one-dimensional algorithm was on the light bar scenario and this blind deconvolution estimated the object well enough to determine the size of the object. Though the intensities are off and not equal throughout the light bar, it still describes the object as being five pixels in length in a vertical position. The results of the estimation of the 5 pixel light bar can be found in Figure 21.

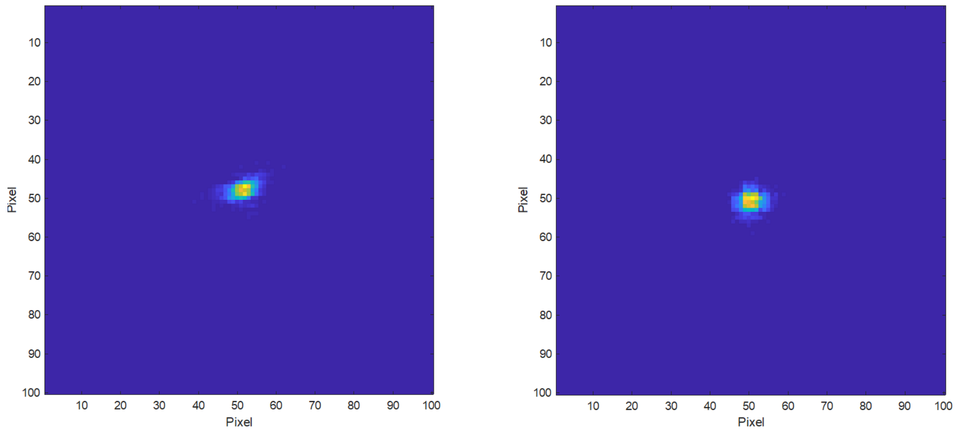

Figure 18: The two data sets used for the one-dimensional blind deconvolution of the four point sources with atmospheric turbulence convolved with the object.

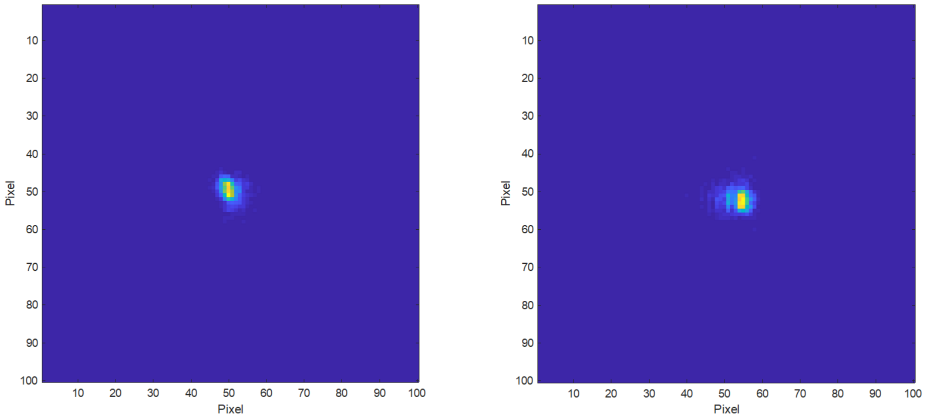

Figure 19: The two data sets used for the one-dimensional blind deconvolution of the light bar with atmospheric turbulence convolved with the object.

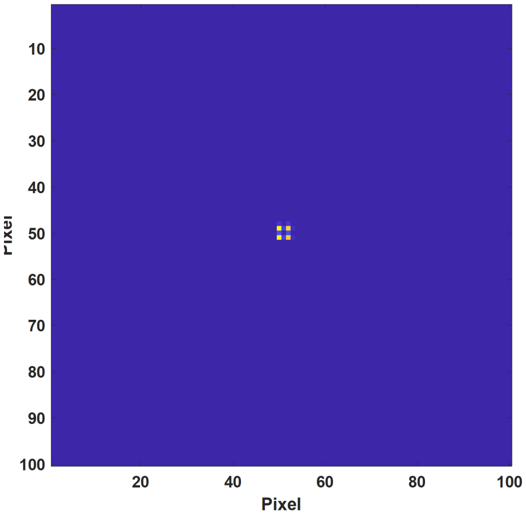

Figure 20: One-dimensional blind deconvolution of the four point source scenario.

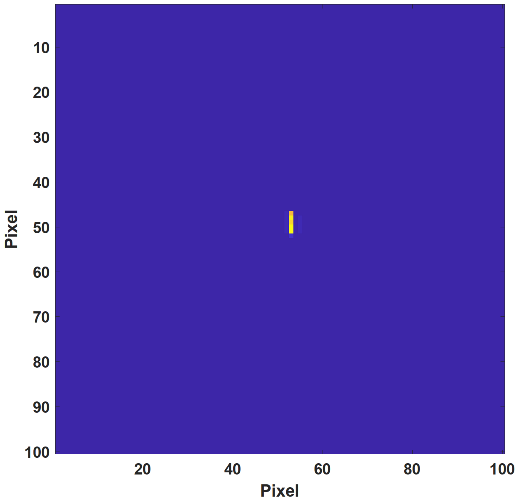

Figure 21: One-dimensional blind deconvolution of the light bar scenario.

4.3 Laboratory Generated Data

This section will discuss the deconvolution of data collected using a laboratory setup. This final scenario was collected using a monitor, camera, a lens for focusing and a heat source. The camera was situated approximately 1.5 meters away from the two pixels displayed on a computer monitor. To focus the image onto the detector a focusing lens was positioned to accomplish this. This lens was positioned between the detector and the monitor. To generate realistic atmospheric turbulence a heat source was introduced around the lenses to create the boiling effect of the atmosphere. The characteristics of the lens, camera and monitor are found in Table 7. The image that was used for the laboratory setup can be found in Figure 22, which is two white pixels separated by one black pixel. This figure is zoomed in to easily see the two pixels, their location and the separation.

Table 7: Specifications for laboratory setup

Equipment		Specifications
Lens	Diameter (mm)	50.00+/-.025
	Focal Length (mm)	200.00+/-4.00
	f/#	4.00
	Lens Type	Achromatic
	Wavelength Range (nm)	750-1550
Camera	Pixels	3296x2472
	Pixel Size (µm)	5.5x5.5
	Exposure Time (ms)	15
	ADC Gain	1023
	Aperture Diameter (mm)	25.4 +/-.2
Monitor	Screen Resolution	1920x1080
	Aspect Ratio	16:9
	Pixel Size (mm)	0.276

Introducing the heat source around the lens and the camera system simulated a turbulent atmosphere that would tilt and de-focus the light around the lens and camera. This tilt and de-focus can be seen in Figure 23. Looking at these two data sets it can be difficult or nearly impossible to determine if two objects are present

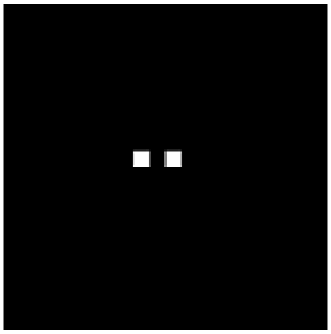

Figure 22: Two pixels loaded to the monitor to be collected by the camera in the laboratory setting.

around the targeted brightness.

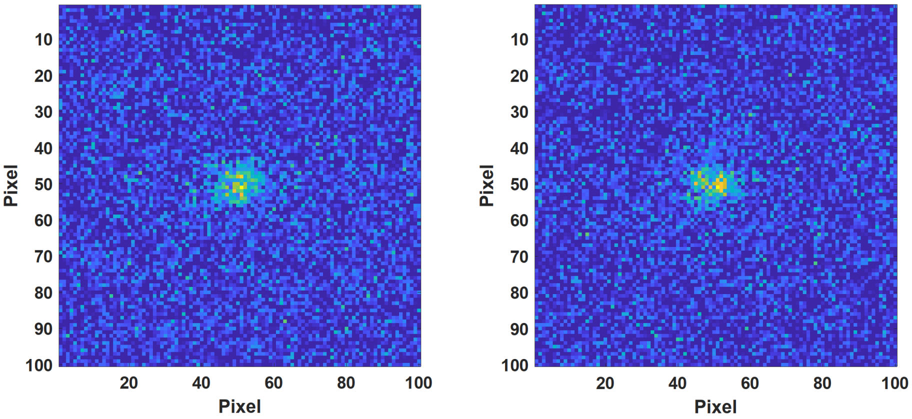

Figure 23: Two data sets collected using the laboratory setup that has atmospheric turbulence added using a heat source.

When these two data sets were deconvolved using the one-dimensional blind deconvolution algorithm it was found that the number of iterations needed to be reduced by a factor of 10. This is due to how quickly the algorithm will converge to a point source, which is not the desired outcome. When deconvolved the images do not appear as point sources due to the geometry of the camera, specifically the size of the individual detectors and the spacing between them. In Figure 24, the figure on the right is the true image that was collected without any turbulence added from monitor through to the lens then to the camera. The figure on the left is the deconvolved object and as can be seen from the figure the pixel diameter for both figures is approximately 5 with a slightly difference in pixel spacing. The estimation of the laboratory generated data is off by one pixel in pixel separation due to the way that the light was scattered across the detector. Increasing the number of iterations would give a separation pixel count that matched closer to the true data but you lose out on the details about the right object and may end up missing an object.

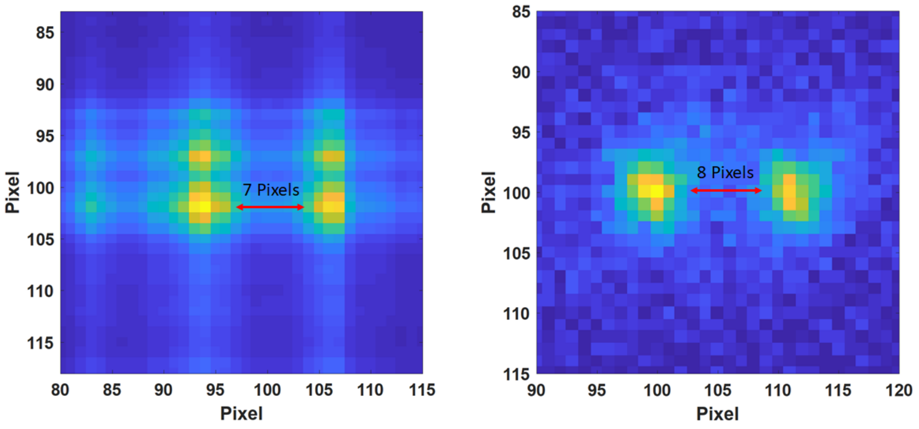

Figure 24: Left: One Dimensional Blind Deconvolution of two point sources collected in a lab setting. Right: True data with no atmospheric simulated turbulence added using a heat source.

4.4 Limiting Factors that Effect Results

This section will discuss the limiting factors that will cause the results of the algorithm to perform poorly. These factors include the SNR of the data, the difference in the PSF caused by the turbulence in the atmosphere, and the separation of the object based on the geometry of the detectors.

4.4.1 Signal-to-Noise Ratio Differences

Demonstrated in the previous sections in this chapter were simulations where the differing SNRs were tested to see how well the one-dimensional algorithm performed at estimating the object. For the high SNR data, the estimation of the object was as expected and performed better than the two-dimensional algorithm in most cases. Though, when changing the intensity of the object to cause a lower SNR value it was visibly clear through the error that both algorithms struggled with lower SNR data. As mentioned in the previous sections, objects whether they are in LEO, geosynchronous orbit or light years away can have dramatically differing SNRs. Objects in LEO may be dimmer than objects that are light years away, while both objects can be spatially separable.

4.4.2 Atmospheric Turbulence Differences

The differences in the PSF caused by the atmospheric turbulence, the boiling effect of the air and the effect it has on light, causes slight variations in the way the algorithm performs. When the correlation between two PSFs are low this enables the algorithm to perform better at resolving the object or objects. When looking at the equation for the image if the PSFs are to similar it leads to an issue where there are too many unknowns and not enough equations to solve for the object.

4.4.3 Geometry of the Detector

The final factor that will effect the results of the estimation of the object is how far the objects are separated. A detector may not be able to detect another object if the individual detector pixel size is larger than the spacing of the object on the detector plane. With further image processing and optical theory, the object can be isolated to a location on a detector based on the pattern on a detector as described in [3].

V. Conclusions

This chapter details the conclusions that were drawn from the results of both the simulated data and the laboratory data. This chapter will also discuss future work that can be done in order to further speed up the deconvolution algorithm or increase the performance of the one-dimensional algorithm.

5.1 Conclusions

It has been shown, through simulation, that the one-dimensional blind deconvolution algorithm was able to perform better at determining the intensity of the objects. Also, the one-dimensional blind deconvolution algorithm performed better in low SNR situations than the two-dimensional blind deconvolution algorithm, when resolving a binary star system.

Thus, since the one-dimensional blind deconvolution algorithms performs better it was proven that the blind deconvolution algorithm was able to estimate objects that were more complex than a binary star system. This algorithm was able to estimate the correct location and size of the unique objects.

Lastly, the one-dimensional blind deconvolution algorithm was performed in a laboratory setting. This algorithm was able to estimate the object comparing it to an image that has no introduced atmospheric turbulence. Much more work can be done in order to speed up and increase the performance of the algorithm.

5.2 Future Work

There is significant amount of work that can still be done in order to improve the algorithm to perform significantly faster than the two-dimensional blind deconvolution algorithm. One particular avenue to decrease the amount of time for the algorithm would be to separate the PSF into it's own separable components. This would not only reduce the number of Fourier transforms that would need to be used but the elimination of the two dimensional Fourier transforms would be removed, which is processing intensive.

One other aspect that was not researched was the ability to use this algorithm on actual images collected by a ground-based telescope. As long as the imaged object is spatially separable the one-dimensional blind deconvolution algorithm should not have any issues resolving the object. Noise may be a factor that may distort the results and further research will aid in determining the ability for the algorithm to be implemented in a ground-based system.

Bibliography

1. Daniel L Pimentel-Alarcón, Ashish Tiwari, Douglas A Hope, and CO Stuart M Jefferies. Leveraging machine learning for high-resolution restoration of satellite imagery. 2018.

2. Charles L. Matson and Kathy Borelli. Parallelization and automation of a blind deconvolution algorithm. *Proceedings - HPCMP Users Group Conference, UGC 2006*, pages 327–330, 2006.

3. Tyler J. Hardy. Optical Theory Improvements to Space Domain Awareness. *Defense Technical Information Center*, 2016.

4. James R Fienup, Joseph C Marron, Timothy J Schulz, and John H Seldin. Hubble space telescope characterized by using phase-retrieval algorithms. *Applied optics*, 32(10):1747–1767, 1993.

5. Isaac B Putnam and Stephen C Cain. Modeling a temporally evolving atmosphere with zernike polynomials. Technical report, Air Force Institute of Technology, Wright-Patterson AFB, OH, 2012.

6. Timothy J. Schulz. Multiframe blind deconvolution of astronomical images. *Journal of the Optical Society of America A*, 1993.

7. Robert J Noll. Zernike polynomials and atmospheric turbulence TL - 66. *Journal of the Optical Society of America*, 66 VN - r(3):207, 1976.

8. T. Söderström. Linear Quadratic Gaussian Control. 32(31):319–365, 2002.

9. Robert A Gonsalves. Phase retrieval and diversity in adaptive optics. *Optical Engineering*, 21(5):215829, 1982.

10. Heinz H Bauschke, Patrick L Combettes, and D Russell Luke. Phase retrieval, error reduction algorithm, and fienup variants: a view from convex optimization. *JOSA A*, 19(7):1334–1345, 2002.

11. J. R. Fienup. Phase retrieval algorithms: a comparison. *Applied Optics*, 21(15):2758, 1982.

12. José M Bioucas-Dias. Bayesian wavelet-based image deconvolution: a gem algorithm exploiting a class of heavy-tailed priors. *IEEE Transactions on Image Processing*, 15(4):937–951, 2006.

13. Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

14. Filip Sroubek and Jan Flusser. Multichannel blind deconvolution of spatially misaligned images. *IEEE Transactions on Image Processing*, 14(7):874–883, 2005.

15. Alfred S Carasso. Apex blind deconvolution of color hubble space telescope imagery and other astronomical data. *Optical Engineering*, 45(10):107004, 2006.

16. H Andrews and CLIII Patterson. Singular value decomposition (svd) image coding. *IEEE transactions on Communications*, 24(4):425–432, 1976.

17. Stephen C. Cain. Super-resolution imaging via expectation-maximization estimation of near stellar neighborhoods. *SPIE Optical Engineering + Applications*, 10772(September):12, 2018.

Acronyms

AFRL Air Force Research Laboratory. 1

CCD charge-coupled device. 5, 7, 12

CPU core processing unit. 1, 2, 8

EM expectation-maximization. 12, 19, 28, 29

GEM generalized expectation-maximization. 12

LEO low earth orbit. 1, 32, 44

MAP maximum a posteriori. 13

OTF optical transfer function. 8

PSF point spread function. vii, 1, 2, 7, 8, 11, 12, 13, 17, 19, 21, 23, 24, 25, 27, 28, 29, 32, 33, 44, 45, 47

ROC receiver operating characteristic. 17

SDA space domain awareness. 1

SNR signal-to-noise ratio. vii, ix, 13, 14, 17, 32, 34, 35, 36, 44, 47

SVD singular value decomposition. ix, 17, 18, 19

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)* 03/26/2020	**2. REPORT TYPE** Master's Thesis	**3. DATES COVERED** *(From - To)* Sep 2018 - Mar 2020

4. TITLE AND SUBTITLE
One-Dimensional Multi-Frame Blind Deconvolution Using Astronomical Data for Spatially Separable Objects

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Brown, Marc R, Capt, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
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-MS-20-M-008

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Lt Col Scott Pierce
Maui High Performance Computing Center
550 Lipoa Pkwy # 100
Kihei, HI 96753

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RDSMC

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A.
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES
This work is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT
Blind deconvolution is used to complete missions to detect adversary assets in space and to defend the nation's assets. A new algorithm was developed to perform blind deconvolution for objects that are spatially separable using multiple frames of data. This new one-dimensional approach uses the expectation-maximization algorithm to blindly deconvolve spatially separable objects. This object separation reduces the size of the object matrix from an NxN matrix to two singular vectors of length N. With limited knowledge of the object and point spread function the one-dimensional algorithm successfully deconvolved the objects in both simulated and laboratory data.

15. SUBJECT TERMS
multi-frame blind deconvolution, spatially separable objects, astronomical data deconvolution, one-dimensional blind deconvolution

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Maj David J. Becker
					19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	62	937-255-3636x4371