

The Plenoptic Sensor for Smart Anisoplanatic Aberration Correction

Simulated Performance in MATLAB

S. A. Stouten

Master of Science Thesis

The Plenoptic Sensor for Smart Anisoplanatic Aberration Correction

Simulated Performance in MATLAB

MASTER OF SCIENCE THESIS

For the degree of Master of Science in Systems and Control at Delft
University of Technology

S. A. Stouten

February 19, 2021

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of
Technology

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF
DELFT CENTER FOR SYSTEMS AND CONTROL (DCSC)

The undersigned hereby certify that they have read and recommend to the Faculty of
Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis
entitled

THE PLENOPTIC SENSOR FOR SMART ANISOPLANATIC ABERRATION
CORRECTION

by

S. A. STOUTEN

in partial fulfillment of the requirements for the degree of
MASTER OF SCIENCE SYSTEMS AND CONTROL

Dated: February 19, 2021

Supervisor(s):

prof.dr.ir. M. Verhaegen

Dr. O. Soloviev

Reader(s):

Dr. S.F. Pereira

Dr. M. Kok

Abstract

Turbulent layers high in the atmosphere cause anisoplanatic phase aberrations, that are responsible for image degradation [6, 33, 46]. Adaptive Optics (AO) aims to correct for these aberrations by sensing them with a wavefront sensor and performing the correction with phase conjugate devices [18].

Various wavefront sensors have been designed and successfully implemented over the years, including the curvature sensor, interferometers, the Shack-Hartmann (SH) sensor and the pyramid sensor [7, 32, 40, 51]. In recent work the plenoptic sensor, an intermediate between the SH and pyramid sensor designs [10, 32, 37], has been proposed to improve on modern wavefront sensing. It has been reported to be beneficial in presence of strong or complex wavefronts, such as deep turbulence conditions [18, 47].

The goal of this thesis is to investigate the SH and plenoptic wavefront sensors, with the specific application on the correction of anisoplanatic aberrations that vary throughout the field of view. The contribution of this thesis to the field of AO is the development of a simulation toolbox in MATLAB. This toolbox is designed to simulate the SH and plenoptic sensors, providing a comparative study between the two.

The comparative simulations reinforced results obtained from a corresponding Literature Survey. The SH sensor can outperform the plenoptic sensor on many occasions. If both sensors share the same microlens array (MLA) the plenoptic sensor scores worse in terms of performance metrics. By adjusting the MLA of the plenoptic sensor its dynamic range and sensitivity can be improved, such that it outperforms the SH sensor. Additionally, it was shown that the plenoptic sensor performs best for strong aberrations, simulated using randomly-generated Kolmogorov screens. On the other hand, it fails in the presence of weak aberrations where the SH performs best.

The developed toolbox allows for iso- and anisoplanatic aberrations to be reconstructed in a single frame, by differentiating between aperture plane reconstruction and phase screen retrieval. Multiple approaches are implemented in the toolbox, such that different reconstruction methods can be selected depending on experimental conditions.

Table of Contents

Preface	ix
Acknowledgements	xi
1 Introduction	1
2 Adaptive Optics	3
3 Wavefront Sensors	5
3-1 Brief Outline Wavefront Sensors	5
3-2 Shack-Hartmann Sensor	6
3-2-1 Working Principles	6
3-2-2 Sensor Drawbacks	9
3-2-3 Adjustments to the SH Sensor	11
3-2-4 Performance Metrics	13
3-3 Plenoptic Sensor	14
3-3-1 Working Principles	16
3-3-2 Sensor Drawbacks	21
3-3-3 Adjustments to the Plenoptic Sensor	22
3-3-4 Performance Metrics	24
3-4 Comparison of the Shack-Hartmann and Plenoptic Sensors	25
3-4-1 Practical Restrictions	26
3-4-2 Theoretical Comparison	26
3-4-3 Practical Comparison of Metrics	28
3-4-4 Comparing Linearity	30
3-4-5 Conclusion	32

4	Isoplanatic versus Anisoplanatic Aberrations	33
4-1	Atmospheric Turbulence	33
4-2	Anisoplanatic Aberrations	36
5	Simulation Toolbox	39
5-1	Discrete Modelling of Waves and Wave Propagation	40
5-2	Supporting Toolbox Functions	44
5-2-1	Digitizing Sensor Image	45
5-2-2	Field Shifting - <i>manipmat.m</i>	46
5-2-3	Preliminary Conditions - <i>checkprelim.m</i>	48
5-2-4	Generating Input Phase Aberrations - <i>getpupil.m</i>	51
5-2-5	Focal Plane Propagation - <i>planeprop.m</i>	54
5-2-6	Retrieving Phase Gradients - <i>getPG.m</i>	58
5-2-7	Reconstruct Phase - <i>getphase.m</i>	67
5-2-8	Retrieve Phase Screen - <i>getscreen.m</i>	70
5-2-9	Correct Aberration - <i>correctphase.m</i>	76
5-2-10	Plenoptic Ray Tracing - <i>plenopticRT.m</i>	77
5-3	Modelling the Shack-Hartmann and Plenoptic Sensors	80
5-3-1	Shack-Hartmann sensor	80
5-3-2	Plenoptic 4F sensor	83
5-4	Isoplanatic Simulation Results	85
5-4-1	Isoplanatic Zernike Modes	85
5-4-2	Isoplanatic Kolmogorov Screens	89
5-4-3	Iterative Isoplanatic Phase Correction	93
5-4-4	Conclusion on Isoplanatic Results	101
5-5	Comparing the Shack-Hartmann and Plenoptic 4F Sensor	101
5-6	Anisoplanatic Simulation Results	107
6	Conclusion	113
6-1	Recommendations for Future Research	115
A	Background Information	117
A-1	Geometrical versus Wave Optics	117
A-1-1	Geometrical Optics	117
A-1-2	Wave Optics	120
A-1-3	Free Space Propagation	124
A-2	The Discrete Fourier Transform	125
A-3	The Point-Spread Function	129
A-4	Zernike Polynomials	130
A-5	Basic Modal Reconstruction	132
A-5-1	Implementation with Square Sub-images	133
A-6	Basic Zonal Reconstruction	135
A-6-1	Implementation with Southwell Geometry	138
A-7	The Minimum-Variance Unbiased Least Squares Estimator	141

Bibliography	147
Glossary	153
List of Acronyms	154

List of Algorithms

1	Focal Plane Fresnel Propagation	56
1	Focal Plane Fresnel Propagation (continued)	57
1	Focal Plane Fresnel Propagation (continued)	58
2	Image Cluster Detection	63
2	Image Cluster Detection (continued)	64
3	Plenoptic Pixel Reshuffling	66
4	Retrieve Phase Gradients from Sensor Image	67
5	Reconstructing Aperture Plane Phase	68
6	Retrieving Phase Screen	75
6	Retrieving Phase Screen (continued)	76
7	Correcting Reference Phase Aberrations	77
8	Plenoptic Phase Screen Reconstruction using Ray Tracing	79
9	Simulating SH Sensor Image and Phase Reconstruction	82
10	Simulating plenoptic 4F (P4F) Sensor Image and Phase Reconstruction	84

Preface

This document presents the report written for my MSc Thesis Project, demonstrating my contribution to the field of Adaptive Optics. My interest for this field was sparked by one of the assignments of the course Filtering & System Identification, given by prof.dr.ir Michel Verhaegen. The way that Control Engineering and System Identification could be used to correct images of stars, thousands of kilometres away, inspired me to meet with dr. Oleg Soloviev. He inspired me to follow courses on Control for High-Resolution Imaging and the Adaptive Optics Project, where I learned the workings of wavefront sensors, wavefront reconstruction and the basics of simulating such systems. With my background in Mechanical Engineering and Systems & Control I was inspired by the field of optics, as it was something different than I experienced in previous studies and it would pose an interesting challenge. From this perspective I chose the plenoptic sensor for my MSc Thesis Project, it had not been presented in previous courses and so I could extend my understanding of physics and mathematics beyond what I already knew.

This thesis documents a literature survey on wavefront sensing, putting the emphasis on two specific sensors that are selected for a comparative study. As support of this study a simulation toolbox is developed in order to simulate the selected sensors. This toolbox implements an anisoplanatic aberration model such that the influence of atmospheric turbulence on the sensor images can be simulated. The implementation of this anisoplanatic model, together with the toolbox development as a whole represents by contribution to the field of Adaptive Optics and computational imaging.

This thesis is written for readers interested in Adaptive Optics, wavefront sensors and wavefront correction. Chapter 1 introduces the topic, gives an overview of the work in wavefront sensing and defines the research questions. Chapter 2 introduces the basics of Adaptive Optics, Chapter 3 covers the studies in wavefront sensing, focussing on the working principles of the Shack-Hartman and plenoptic sensors. The representation of atmospheric turbulence is detailed in Chapter 4, introducing the effect of anisoplanatism on the imaging process. The development of the simulation toolbox is elaborated in Chapter 5, presenting important algorithms simulation results. The final chapter, Chapter 6 concludes the work done, answers the research questions and provides recommendations for future study and improvements.

Delft, University of Technology
February 19, 2021

S. A. Stouten

Master of Science Thesis

S. A. Stouten

Acknowledgements

The work covered in this thesis could not have been possible without the many people that provided me with support. To all of you whom I have bothered with endless questions I want to express my deepest appreciation, for your help but also your patience.

More specifically, I would like to thank dr. Oleg Soloviev for being my daily supervisor, guiding me with expert advice and encouraging me with unending enthusiasm. Your support has been invaluable during my thesis. I would also like to thank prof. dr. Gleb Vdovin for the supportive critique and help using the LightPipes software package. Furthermore I would like to express my thanks to prof. dr. ir Michel Verhaegen for the helpful comments and insightful suggestions. Furthermore, I would like to thank all members of the Smart Imaging Lab for all the interesting presentations, group meetings and fruitful discussions throughout the year. Additional thanks go out to [44] for providing the photo used for the cover of this thesis.

Last but not least I would like to thank Lisa de Vries from the bottom of my heart. Without her everlasting patience and support, this thesis could not have been completed as it is today.

Delft, University of Technology
February 19, 2021

S. A. Stouten

“Remember to look up at the stars and not down at your feet. Try to make sense of what you see and wonder about what makes the Universe exist. Be curious. And however difficult life may seem, there is always something you can do and succeed at. It matters that you don’t just give up.”

— *Stephen Hawking*

Chapter 1

Introduction

Throughout the ages scientists have used telescopes to investigate the cryptic cosmos, imaging enormous stellar objects millions of kilometres away. These systems have grown both in size and power, but even the largest telescopes suffer from atmospheric turbulence [9, 27]. Inhomogeneities in Earth's atmosphere introduce *phase aberrations* to light that reaches the surface, severely affecting image quality [33]. A common solution is to employ Adaptive Optics (AO) systems to increase the image quality of the telescope [13, 17, 27, 33, 41]. Using a *wavefront sensor* (WFS) these systems can compensate for these aberrations, resulting in a sharp image.

For most applications the Shack-Hartmann (SH) wavefront sensor is used, a sensor that has been developed thoroughly for decades [2, 47]. This sensor performs excellently under weak aberrations, but its performance depends on local complexities and fails under strong and complex aberrations [18, 25, 26, 47].

To improve on the design of the SH sensor a number of new wavefront sensors emerged. Designs that showed promise within this development were the pyramid and plenoptic sensor. The pyramid sensor showed an increased sensitivity and adjustable pupil sampling as well as increased closed-loop performance [35, 40]. The plenoptic sensor further improved performance by increasing the pupil sampling and the dynamic range capabilities [47].

Standard methods allow wavefront sensors to interpret the recorded image and produce a reconstruction of the aberrations, sensed in the aperture plane. In practice these aberrations are not introduced here but outside the wavefront sensor (WFS) and thus sensed aberrations may differ depending on the imaging location. If sensed aberrations do not change throughout the field of view (FOV) one considers them *isoplanatic aberrations*. In reality this is often not the case and the aberrations vary over the FOV, making them *anisoplanatic aberrations* [6, 33, 46].

For the purpose of astronomy the solution to this anisoplanatic problem starts by describing atmospheric turbulence. The statistics of this turbulence can be represented by the common model of Kolmogorov theory [6, 13, 17, 31, 39]. The three-dimensional volume of atmospheric

turbulence itself can be approximated by dominant layers, which can be represented by phase screens [33, 46].

The aim of this thesis is to investigate if the plenoptic sensor is a viable competitor to the SH wavefront sensor and could serve as a better alternative in the presence of anisoplanatic aberrations. The theoretical part of this work is focussed on the working principles of various wavefront sensors and performs a comparative study on two of these sensors: the SH and plenoptic WFS. This comparison leads to the conclusion that the SH sensor does not yield effortlessly and can still outperform the plenoptic sensor on many occasions. For specific sensing tasks at hand however, the plenoptic sensor can be designed to triumph.

The practical part of this thesis is focussed around the development of a simulation toolbox to test these optical systems and perform similar comparisons. The purpose of this toolbox is to simulate the WFS imaging process, reconstruct the phase aberrations and correct for them. In the context of this thesis the toolbox is used to compare the SH and plenoptic sensors performance, not develop faster algorithms for wavefront reconstruction.

The narrative of this report is guided by the following research questions, that have been established for this thesis at the time of literature survey:

How does the plenoptic sensor improve modern wavefront sensing and will it provide a better alternative to the Shack-Hartmann sensor?

How can anisoplanatic phase reconstruction be conducted in a single frame?

The report is structured to highlight this narrative and the purpose of the toolbox, starting with an introduction to the basics of AO in Chapter 2. A brief overview of common wavefront sensor is given in Chapter 3, followed by an in-depth investigation of the SH and plenoptic sensors. Chapter 4 discusses the nature of atmospheric turbulence and introduces anisoplanatic aberrations to the imaging process. These two chapters collect the literature survey conducted for this thesis. The development of the toolbox, important algorithms and results are presented in Chapter 5. The thesis comes to a close in Chapter 6, where the conclusions and future recommendations are presented.

Chapter 2

Adaptive Optics

Most Adaptive Optics (AO) systems are defined by three important parts: 1) the wavefront sensor (WFS), 2) the deformable mirror (DM) and 3) the control system [12, 13, 17, 40]. An AO system is added to an existing imaging system, such as a telescope or microscope to increase the performance in a number of ways. An overview of the basic components of an AO system applied to a telescope is shown in Figure 2-1.

Light that is emitted from a distant star arrives at the outer layers of Earth's atmosphere with a plane and undistorted wavefront. Atmospheric turbulence between these outer layers and Earth's surface interfere with incoming light and the wavefront arriving at the telescope aperture is no longer plane but distorted [17]. The incoming light can be defined as a complex field $U = Ae^{j\phi}$, where A is the magnitude and ϕ the phase of light, with j the imaginary unit. The aberrations that result from atmospheric turbulence end up in the phase term $\phi(\rho, t)$, that is a function of both spatial aperture coordinates $\rho \in \mathbb{R}^2$ and time t .

The ultimate goal of the AO system is to reconstruct the phase correctly and to compensate for it, such that the light that falls on the image sensor is approximately a plane wave. For more details about the wave-nature of light, the reader is referred to Appendix A-1-2. The wavefront of light $w(x, y)$ can be related to the phase of the corresponding complex field $\phi(x, y)$ using the wavelength of light λ , expressed in [m]:

$$\phi(x, y) \approx \frac{2\pi}{\lambda}. \quad (2-1)$$

The path of light through an AO system can be traced as illustrated in Figure 2-1 for a telescope. Here light is collected by the telescope and limited by the telescope's aperture. The aperture located in the *aperture plane* ensures that all light travelling through it ends up on the image sensor. Light should thus only be limited by the aperture and nowhere else.

Beyond the telescope light comes into contact with the DM, in Figure 2-1 the adaptive mirror, which can be controlled to change its shape and affect incoming wavefronts. The DM introduces a phase delay in a similar manner as lenses do, elaborated in Appendix A-1-2, but also reflects incoming light to change its direction.

After the DM the light beam is split in two by a beam splitter, such that one part goes through the original imaging system forming an image, while the other part of light goes through the WFS. The beam splitter enables one to obtain an image and sense the wavefront at the same time, since a WFS takes in all light to produce its measurements.

These measurements are sent to a control system that translates them into a control action for the DM, closing the AO-loop traced by the dashed line.

The goal of this loop is to drive the distorted wavefront of incoming light to a corrected wavefront that is close to a plane wave. The ultimate goal of the AO system is to perform this loop in real-time, offering continuous real-time aberration correction, such that much sharper images can be obtained.

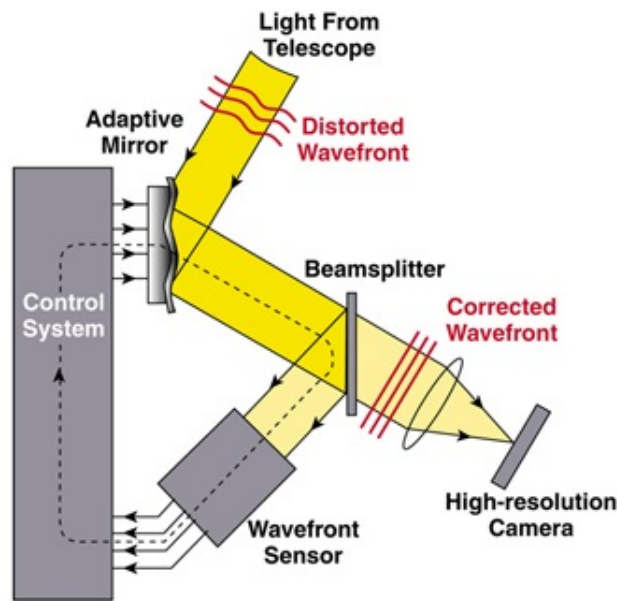


Figure 2-1: Basic adaptive optics system [27].

As noted in [17, 32], an important complication in astronomical imaging and wavefront sensing is that light is scarce and a WFS may not be able to accurately sense the atmospheric aberrations. Stars and other stellar science objects are often indicated by a stellar or visual magnitude, where a larger magnitude means the object is less bright. As an example, a to-be-imaged object with stellar magnitude of 13 or more, will be too faint for the WFS to function properly. For this reason AO operates around the vicinity of bright stars that act as a guiding beacon for objects close to them. This assumes that the light coming from both the guide star and science object experience roughly the same turbulence, such that measurement from the guide star can be used to correct aberrations associated with the imaged object.

This thesis is centred around wavefront sensing and reconstruction, as indicated in Chapter 1, which will be discussed in Chapter 3. For the purpose of aberration correction a highly idealized DM is used but further analysis on DM and control systems are beyond the scope of this work and not discussed. The correction of aberrations is addressed together with the development of the simulation toolbox in Chapter 5.

Wavefront Sensors

The wavefront sensor (WFS) is one of the most important parts of an Adaptive Optics (AO) system. This chapter presents a brief overview of common wavefront sensors and discusses the principles of both wavefront sensing and reconstruction. This chapter concentrates on the Shack-Hartmann (SH) and plenoptic sensors, discussing their working principles, drawbacks and possible adjustments. Following the details of each sensor a comparison is drawn between the two on multiple grounds, on which the conclusion of this chapter is based. The information presented in this chapter follows from a literature survey conducted for this thesis.

3-1 Brief Outline Wavefront Sensors

This section presents a brief overview of wavefront sensors. As stated by [51], a WFS can be defined as any optical system that provides wavefront information on an incident beam of light. Such a device differs from an ordinary image sensor, such as a CCD or CMOS image sensor, that only records the intensity distribution of incoming light. It has been stated by [40], that no optical device can record the wavefront of light directly, but instead a WFS infers information on the wavefront of incident light based on the recorded intensity pattern. The WFS then produces an estimate of the measured wavefront that can be translated by a computer into a control signal for the deformable mirror (DM) [27]. Typical examples of wavefront sensors are Curvature sensors, Interferometers, the SH WFS and the pyramid sensor [7, 32, 40, 51] which will be touched upon below. For more details on the representation of light and its propagation through thin lenses, the reader is referred to Appendix A-1.

The four examples of wavefront sensors are briefly outlined using [11, 40, 47, 51], starting with the curvature sensor. This sensor can be used to detect local focus or defocus of incoming light. Two images are taken, one at a distance l in front of the focal plane and one at a distance l behind the focal plane, with the focal distance $f \gg l$. By comparing both images additional curvature distributions with respect to a plane wave can be extracted.

An interferometer uses a reference beam of light with known intensity and phase to interfere with the aberrated beam of the same wavelength. Then, by observing the resulting interfered

intensity pattern information on both the intensity and phase distributions can be revealed.

The SH WFS uses a microlens array (MLA) to spatially samples incoming light. Each lenslet images a spatial part of the wavefront into a focussed spot on the image sensor. This spot's deviation from the reference is used to determine the local tilt of the imaged wavefront part.

Finally, the pyramid sensor uses a four-sides glass pyramidal prism and a set of lenses to split a focussed beam into four images on an image sensor. The locations of illuminated pixels then reveal information on the incident wavefront slopes.

In addition to common sensors, the plenoptic sensor has been proposed to surpass the SH WFS when exposed to strong and complex wavefronts [18, 47]. As explained in [10, 32, 37], the plenoptic sensor can be interpreted as an intermediate design between the SH and pyramid sensors. The design contains a main objective lens, an MLA and an image sensor, such as a CCD detector. The illuminated pixels and the corresponding locations reveal the phase information, just as the pyramid sensor does, but the MLA makes it possible to sample the incoming light from multiple points of view.

The plenoptic sensor will be compared to wavefront sensors of similar design, such that a thorough comparison can be done on both design and sensor performance. The SH WFS has been used in AO extensively and received many years of study [2, 25, 26, 47], making it a sensor worth investigating. The pyramid sensor has been proposed as a better alternative to the SH sensor through improved design [2, 9, 40]. The glass prism at the centre of its design is expensive to manufacture and prone to misalignment errors, as reported by [2, 22]. It is clear that the pyramid WFS involves significantly more mechanical and design complexities than the SH and plenoptic sensors. Additionally, the designs of the pyramid sensor and plenoptic WFS are very similar. With these arguments the pyramid sensor is omitted from further investigation and the remainder of this chapter will explore the SH and plenoptic sensors further.

Throughout this thesis the terms *wavefront reconstruction* and *phase reconstruction* are used interchangeably, as they refer to the same *reconstruction algorithm*. In this argument both phase and wavefront reconstruction only differ in factor $2\pi/\lambda$ through approximation (2-1).

3-2 Shack-Hartmann Sensor

The SH WFS has been successfully implemented in AO systems for the past decades [2, 47]. It gained widespread application through its reliability and accuracy, low sensitivity to both noise and misalignments and a simple implementation [26, 32, 47]. Furthermore it does not require any reference wavefronts during the sensing process [26].

The SH sensor is investigated on a number of topics. First, the working principles of the WFS will be elaborated, discussing the image formation process and phase reconstruction. Second, drawbacks of the sensor design are discussed, followed by several possible adjustments. Finally, performance metrics for the SH sensor are derived and listed.

3-2-1 Working Principles

Image Formation The sensor itself consists of an array of identical microlenses, often called lenslets due to their size, with an imaging sensor placed at the lenslet focal distance behind

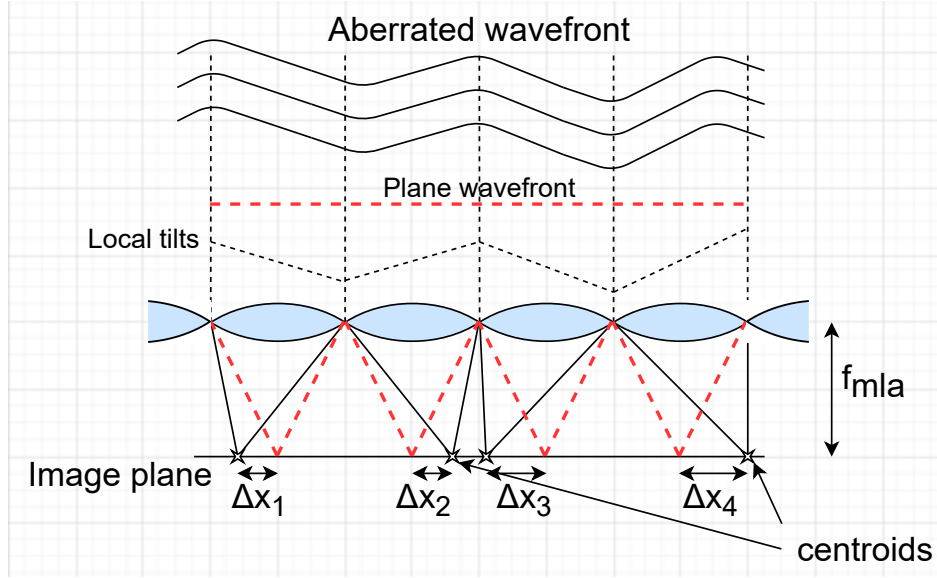


Figure 3-1: Inspiration taken from [26]. An aberrated wavefront incident on a Shack-Hartmann sensor is focussed by the individual (blue) microlenses. The wavefront is sampled locally, meaning the local part of the wavefront directly incident on the microlens is focussed. The local tilts are directly related to the displacements Δx_i of the local centroids, indicated by the stars in the image plane. Through Δx the local tilt with respect to a plane wave (red) can be found, which would be focussed in the focal point of each microlens.

the array [5, 25, 26]. A two-dimensional illustration of the lenses and the imaging process is shown in Figure 3-1. The MLA, as further explained in [25], spatially divides incoming wavefronts into discrete sections, which are focussed by their corresponding lenslet into a spot. Each lenslet images light onto a specific section on the image sensor, the lenslet *sub-image* [51, 52]. From Figure 3-1 it can be concluded that each sub-image must be the same size as the lenslet itself, such that the image is divided into equally sized sub-images.

As highlighted in Section 3-1, the deviation of the spot from the centre of the sub-image is proportional to the local tilts of the wavefront. If for a certain lenslet the imaged wavefront section is flat, the spot will be imaged in the focal point, directly behind the geometric centre of the lenslet. In Figure 3-1 this is illustrated by the 'Plane wavefront' that is imaged in the geometric centre of each sub-image. If the wavefront section deviates from a flat wave, illustrated by the imaging of the 'Aberrated wavefront' in Figure 3-1, the spot on the image sensor will be displaced with respect to the centre. Actual SH sensor images are shown in Figure 3-3 and Figure 3-4 which will be discussed later.

Unfortunately, lens imperfections and diffraction effects prevent the focussed spot from being a single illuminated pixel. Instead, one obtains the *point spread function* (PSF), which is further elaborated in Appendix A-3. To find the displacements in both x - and y -directions, defined as Δx and Δy respectively, one must find the centroid of the PSF. The standard approach of doing so is computing the *image first moment* [26, 51].

Phase Reconstruction The image first moment equations are defined in (3-1), which result in displacement $(\Delta x, \Delta y)$ of the centroid with respect to the centre of the sub-image. The

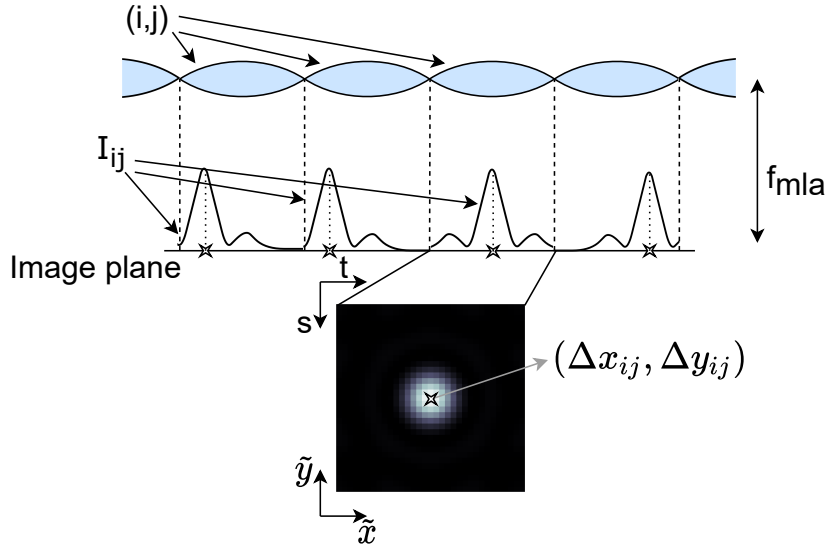


Figure 3-2: Illustration of image first moment calculation for the SH sub-images. Each lenslet (i, j) images light into its sub-image, containing intensity pattern I_{ij} represented by pixels (s, t) . By defining local spatial coordinates (\tilde{x}, \tilde{y}) in each sub-image one can find the spatial displacement $(\Delta x_{ij}, \Delta y_{ij})$.

physical quantities involved in these equations are illustrated in Figure 3-2. For starters, each lenslet and corresponding sub-image is associated with indices (i, j) . Each lenslet (i, j) images the spatially sampled wavefront onto sub-image (i, j) , where the centroid is represented by the 4-point stars analogous to the imaging process shown in Figure 3-1. Each sub-image contains the recorded intensity pattern I_{ij} that is composed of pixels associated with index-pair (s, t) . This means that for lenslet (i, j) one can select pixel (s, t) that contains intensity value $I_{ij}(s, t)$. Finally, to obtain the spatial displacements instead of pixel displacements, one can define a local spatial coordinate grid (\tilde{x}, \tilde{y}) for each sub-image. These coordinates span the physical size of the sub-image with their centre aligned with the sub-image centre. For example, for a square sub-image with half-width r_{mla} , both \tilde{x} and \tilde{y} are defined from $-r_{mla}$ to $+r_{mla}$. The geometric centre of the sub-image should correspond to coordinates $(\tilde{x}, \tilde{y}) = (0, 0)$. With these definitions one can describe each pixel in the whole SH image by indices (i, j, s, t) , such that the entire SH image is described by intensity function $I^{SH}(i, j, s, t)$.

If the MLA contains an $M \times N$ grid of lenslets, each containing a sub-image that consists of $P \times Q$ pixels, displacements $(\Delta x, \Delta y)$ can be found through image first moments:

$$\begin{aligned} \Delta x(i, j) &= \frac{\sum_{s=1}^P \sum_{t=1}^Q \tilde{x}(s, t) \cdot I_{ij}(s, t)}{\sum_{s=1}^P \sum_{t=1}^Q I_{ij}(s, t)}, \\ \Delta y(i, j) &= \frac{\sum_{s=1}^P \sum_{t=1}^Q \tilde{y}(s, t) \cdot I_{ij}(s, t)}{\sum_{s=1}^P \sum_{t=1}^Q I_{ij}(s, t)}. \end{aligned} \tag{3-1}$$

The obtained displacements are related to the wavefront defined in the aperture plane. In order to eliminate possible sensor noise in the retrieval of displacements $(\Delta x, \Delta y)$, pixel intensities $I_{ij}(s, t)$ can be set to zero if they fall below a certain threshold value [26].

This standard approach approximates the aberrated wavefront as a discrete set of local tilts [18, 25, 26, 51]. If the aberrated wavefront is purely a tilted plane wave, the resulting point-spread function (PSF) is shifted from the centre proportional to the wavefront slope. This can be seen by comparing the first three wavefront shapes in Figure A-7, the plane wave 'piston' and the tilted plane waves 'tip' and 'tilt'. By measuring the centroids one measures the displacement in each sub-image caused by the *average slopes* of each wavefront section. This results in a data set of local tilts that correspond to a two-dimensional spatial location [18]. Through geometrical optics, elaborated in Appendix A-1-1, one can extract these average slopes S_x and S_y by dividing displacements Δx and Δy by the lenslet focal distance f_{mla} . Invoking approximation (2-1) allows one to relate each centroid $(\Delta x_{ij}, \Delta y_{ij})$ to the average phase gradient of light incident on lenslet (i, j) :

$$\begin{aligned}\frac{\partial \phi(x_{ij}, y_{ij})}{\partial x} &= \frac{-2\pi}{\lambda f_{mla}} \Delta x_{ij}, \\ \frac{\partial \phi(x_{ij}, y_{ij})}{\partial y} &= \frac{-2\pi}{\lambda f_{mla}} \Delta y_{ij},\end{aligned}\tag{3-2}$$

where the spatially sampled section of lenslet (i, j) is indicated by coordinates (x_{ij}, y_{ij}) . The minus sign corrects for the direction inversion through imaging. If a wavefront with slopes in positive x - and y -directions is imaged by a lens, the focal spot is imaged in negative x - and y -coordinates. Thus the minus sign ensures positive centroid coordinates correspond to negative gradients. With gradients (3-2) the phase in the aperture plane can be reconstructed, using either the modal or zonal reconstruction algorithm elaborated in Appendix A-5 and Appendix A-6.

The spatial sampling of the MLA enables one to specify the type of information encoded in pixel intensity $I^{SH}(i, j, s, t)$. Indices (s, t) specify the position of the pixel in the sub-image which are used to determine the centroids (3-1) and thus the average slopes. Indices (i, j) on the other hand specify which lenslet is considered and thus which average slope is measured. This means that in $I^{SH}(i, j, s, t)$ the *spatial information* is encoded in index-pair (i, j) , whereas *angular information* is encoded in (s, t) .

3-2-2 Sensor Drawbacks

There are a number of drawbacks to be recognized from the SH WFS design. One drawback stated by [47] is that the MLA limits both the lateral resolution and range of measurable wavefront tilts. Each sub-image is the same size as the lenslets so the maximum detectable displacements wavefront slopes are limited by the lenslet size. More complex or strong wavefronts may result in too strong individual displacements, such that spots are focussed in different sub-images and either merge or switch places [25, 47].

As stated by [26] standard approach (3-1) is only suitable if each sub-image contains a single focal spot. Even more so, centroids are assigned to the sub-image they are imaged in, which means that spots moving beyond their respective sub-image are incorrectly assigned to different indices (i, j) . In this context spot merging is likewise problematic as (3-1) detects the *average centroid* of the merged spots.

The impact of this spot merging is reported by [18] and shown in Figure 3-3 and Figure 3-4 for varying turbulence strengths. The leftmost image of Figure 3-3 corresponds to weaker

turbulence and it can be clearly seen that the spots are well-aligned. The rightmost image corresponds to medium turbulence and already shows spots merging along the left side and bottom-right of the image. Finally, in Figure 3-4 strong turbulent conditions are present that result in drastic spot merging and the absence of spots in other sub-images. Using (3-1) under such conditions results in several averaged centroids, as well as zero-centroids in sub-images that are not illuminated - whose spot has moved to different sub-images.

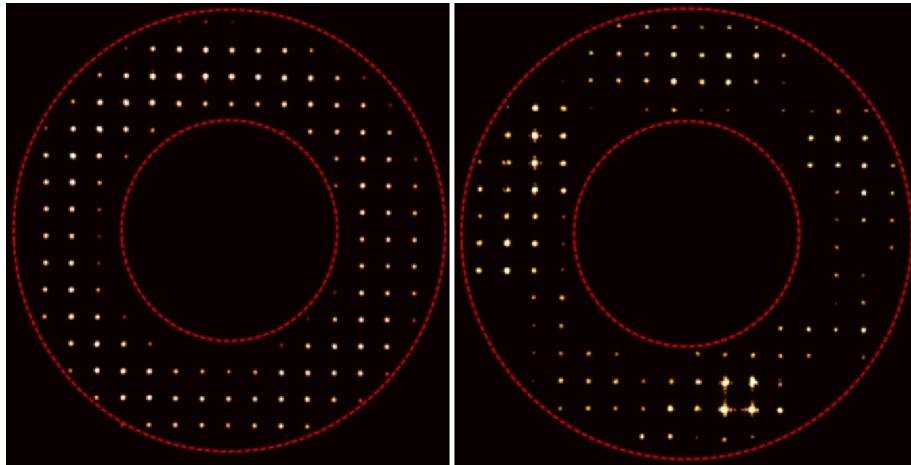


Figure 3-3: Turbulence effects on the Shack-Hartmann pattern. Weak turbulence on the left shows a clear pattern, while medium turbulence on the right begins to show some black-outs and spot merges. The dashed red line shows the annular aperture of the optical system [18].

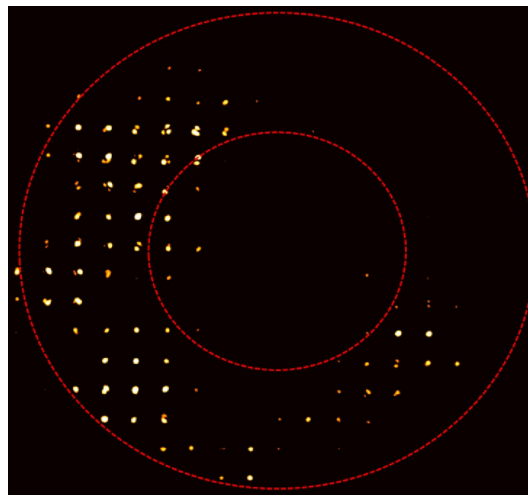


Figure 3-4: Strong turbulence effects on the Shack-Hartmann pattern. Multiple spots per detection area can be seen and more black-outs have occurred. The dashed red line shows the annular aperture of the optical system [18].

A second limitation is light economy, reported by [25]. The MLA spatially samples light, dividing the number of photons over the sub-images. Small lenslets collect less light than larger ones up to the point that noise significantly affects the results. This noise, according to [9, 10, 25], can be divided into photon quantum noise and sensor read noise. The former,

as explained in [25], results from the Poisson distribution describing how many photons arrive at each pixel during exposure. The latter affects all slope-based sensors, as noted in [9, 10]. It can be minimized by reducing the number of detectors, elaborated further in Section 3-2-3.

Another drawback that can be identified for all gradient-based wavefront sensors is the fact that discontinuities in the phase cannot be detected, as reported by [26]. Phase gradients are measured and integrated to obtain the phase, approximating the phase as a continuous surface. These undetectable discontinuities are identified as piston modes, averaged constants that do not affect phase gradients or reconstructions [7, 26, 43].

As a final note towards the limitations, strong wavefronts and atmospheric turbulence have additional effects on the incoming light, such as self-interference, anisoplanatism, branch-points and branch-cuts [18]. The basic notions of atmospheric turbulence and the anisoplanatic model are discussed in Chapter 4 of this thesis, but more complex details and effects are beyond the scope of this study. The reader is referred to [51] for further elaborations.

3-2-3 Adjustments to the SH Sensor

From literature a number of adjustments can be made to the design of the SH sensor. The adjustments listed do not cover all possibilities but do provide an overview on how the performance of the sensor can be adjusted to fit certain requirements. The adjustments have been divided into software and hardware based.

Software Software-based adjustments are algorithmic changes which are coined as smart algorithms [51]. One such algorithm applicable to the SH sensor, is the addition of blob detection in image pre-processing before computing the centroid [51]. Blob detection identifies clusters of illuminated pixels and helps to isolate the focused spots from noise and other random high order phase oscillation structures.

Blob detection is also helpful when multiple point sources are simulated. If each of these sources is imaged incoherently one obtains as many focal spots in each sub-image as there are sources, provided that the focal spots are sufficiently separated. If this is the case, blob detection isolates the spots such that all centroids can be detected in the same sub-image using image first moments (3-1). A minimum condition of sufficient spot separation can be obtained from the *Rayleigh resolution criterion* for diffraction-free imaging. This is elaborated further in Appendix A-3. The benefit of blob detection for imaging multiple point sources is clear and is thus implemented in this work, further detailed in Section 5-2-6.

Removing noise in the data is also reported in [51], by either averaging over multiple integral paths or solving minimum mean square error (MMSE) equations. The latter is already implemented in this work, by reconstructing either the phase or orthonormal Zernike coefficients with the minimum-variance unbiased estimate (MVUE). The former is based on the weak law of large numbers and that each phase gradient is retrieved independently from neighbouring sub-images. Averaging over different integral paths should result in a mean error of zero.

Hardware One hardware adjustment mentioned in Section 3-2-2 is replacing the pixels under each lenslet by a 2×2 quad cell to minimize sensor read noise [7, 9, 10, 11]. Figure 3-5 shows the behaviour of this quad cell for small and large focal spots.

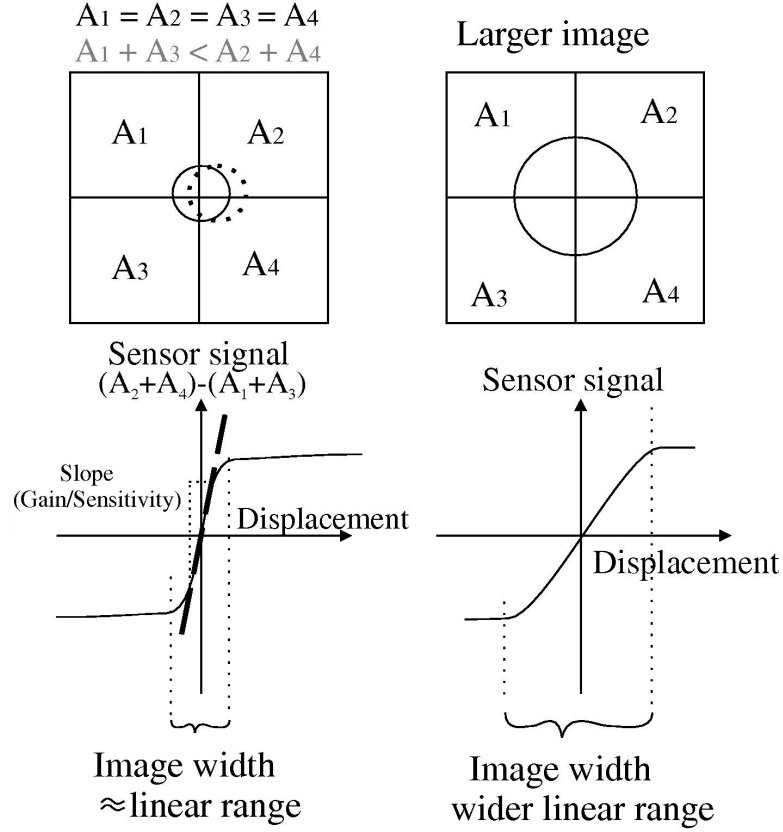


Figure 3-5: Taken from [9]: "The sensor signal is approximately linear for small displacements and saturates for larger displacements. A larger spot size (right) results in a wider range for which the signal is linear, at the expense of the signal gain or sensitivity."

If the sensed displacements of the focal spot are small compared to its spatial width b one can determine said displacements by averaging over sub-aperture areas A_i [7]:

$$\begin{aligned}\Delta x(i, j) &= \frac{b}{2} \frac{A_1 - A_2 - A_3 + A_4}{A_1 + A_2 + A_3 + A_4}, \\ \Delta y(i, j) &= \frac{b}{2} \frac{A_1 + A_2 - A_3 - A_4}{A_1 + A_2 + A_3 + A_4},\end{aligned}\tag{3-3}$$

For flat wavefronts each spot will be imaged in the cell centre and areas $A_1 \dots A_4$ are all equal. Any tilt in the wavefront will shift the spot away from the centre, up to the point that either of the four areas A becomes zero. This is illustrated in the saturation of the sensor signal in Figure 3-5. This saturation results from the spot moving away from area pairs (for example $A_1 + A_3$ or $A_2 + A_4$), such that some values A_i become zero. Larger spots have the ability to move further away from the centre before saturation occurs, increasing the linearity range. As reported by [9] one minimizes sensor and read noise with quad-cells, but severely undersamples the image, possibly reducing the dynamic range of the SH sensor.

3-2-4 Performance Metrics

The general performance of a WFS can be quantified by a number of metrics [18, 47]. It is assumed that the small angle approximation holds, such that $\tan(\theta) \approx \theta$ (A-2).

Dynamic range The dynamic range represents the maximum measurable wavefront tilt [18]. For the SH sensor this tilt is limited to the lenslet size and focal distance. As reported by [18, 47] and mentioned in Section 3-2-1, each sub-image is the same size as the lenslet itself and the maximum detectable tilt is therefore confined to this size:

$$\left. \frac{\partial w(x, y)}{\partial u} \right|_{max}^{SH} = \frac{d_{mla}}{2f_{mla}}, \quad u = \{x, y\}, \quad (3-4)$$

where $w(x, y)$ is the wavefront defined by spatial x - and y -coordinates, d_{mla} and f_{mla} are the size and focal distance of the MLA lenslets.

Sensitivity The sensitivity quantifies the minimum detectable wavefront tilt [18]. In the standard image first moment approach these tilts are obtained at *sub-pixel accuracy*. This means that the SH sensitivity depends on multiple contributions, such as spot size, detector noise and the intensity distribution on the sub-image [9, 26]. An alternative is to express the sensitivity proportional to the *pixel accuracy*, dropping the statistical influences from its description. This pixel accuracy is equal to the minimum detectable slope if one is restricted to integer displacements. Such a sensitivity has been adopted in [18] and could be interpreted as a maximum sub-pixel accuracy:

$$\left. \frac{\partial w(x, y)}{\partial u} \right|_{min}^{SH} \propto \frac{d_{pix}}{2f_{mla}}, \quad u = \{x, y\}, \quad (3-5)$$

where d_{pix} the size of the image sensor pixels. If the image sensor contains an uneven number of pixels the sensitivity of the SH WFS is described by:

$$\left. \frac{\partial w(x, y)}{\partial u} \right|_{min}^{SH} \propto \frac{d_{pix}}{f_{mla}}, \quad u = \{x, y\}. \quad (3-6)$$

Another alternative is to quantify the sensitivity as the minimum wavefront curvature that can be unambiguously detected, as reported by [47]. This results in an expression for the diffraction limit of the SH sensor:

$$d_{mla} > \sqrt{2\lambda f_{mla}}, \quad (3-7)$$

which physically means that the Fresnel number should be larger than 2. For the practical case of $d_{mla} \gg \sqrt{2\lambda f_{mla}}$, moving away from the diffraction limit, the detectable minimum curvature R simplifies to:

$$R > 2f_{mla}. \quad (3-8)$$

Gradient sample size: The gradient sample size represents the number of phase gradients that are collected from a single sensor image, an independent sample. The centroids in the SH sensor are computed for each lenslet in the MLA as defined by (3-1). According to [18] the gradient sample size is equal to the number of illuminated sub-images. As long as the aperture plane is properly defined and all light falls on the MLA, the gradient sample size is equal to the number of sub-images fitting on the image sensor:

$$N_{gradient}^{SH} = \left(\frac{w_{sens}}{d_{mla}} \right)^2 = M^2, \quad (3-9)$$

where w_{sens} is the image sensor width and the MLA consists of an $M \times M$ lenslet grid.

Conclusion on Performance Metrics The SH sensor is known for its high accuracy for weaker aberrations, which is reflected by the expressions for sensitivity in (3-5) and (3-6) that are relatively low, considering pixel size d_{pix} in the order of 100 micrometers. This sensitivity can be adjusted freely by changing the resolution of the image sensor.

The dynamic range is limited by ratio d_{mla}/f_{mla} , which can be increased by either increasing d_{mla} or decreasing f_{mla} . The best option is increasing d_{mla} since a decrease in f_{mla} also degrades the SH sensitivity. Unfortunately, increasing the lenslet size also means increasing the sections of spatially sampled wavefronts, resulting in larger sections of wavefront being approximated by one average slope. This will lead to increased wavefront reconstruction errors, unless the aberrations are restricted to purely tilted waves.

3-3 Plenoptic Sensor

The plenoptic WFS is based on the plenoptic camera [30] and the modern description of the *light-field* presented in [1]. A plenoptic camera samples this four-dimensional light field uniquely through its particular design, enabling it to refocus the image at different depths, after the image has been captured [30, 51]. Using the plenoptic camera to obtain wavefront measurements is reported by several authors [10, 36, 52] where its design provided advantages compared to the SH sensor, which only records the two-dimensional focal plane image.

The plenoptic sensor and camera contain three basic elements: 1) The objective lens, 2) an image sensor such as a CCD or CMOS sensor and 3) an MLA [10, 18, 30, 36, 47, 51]. The MLA is placed behind the objective lens in front of the image sensor, such that each lenslet creates a *sub-image* of the pupil on the image sensor [36]. This pupil is the aperture plane of the plenoptic sensor, the front focal plane of the objective lens. Figure 3-6 shows these basic elements and how the incident wavefront is related to the sub-images.

Literature distinguishes between two configurations of the plenoptic sensor, the $3F$ and $4F$ configurations. The plenoptic $3F$ (P3F) sensor discussed in [36] places the MLA in the back focal plane of the objective lens. The plenoptic $4F$ (P4F) sensor discussed in [53] places the microlens array at $f_{obj} + f_{mla}$ behind the objective lens, such that objective lens back focal plane and array front focal plane overlap. This creates Keplerian configuration between objective lens and each lenslet in the MLA [18, 53]. Literature also reports the design of the plenoptic sensor to be intermediate between the SH and pyramid sensor designs [10, 32, 36,

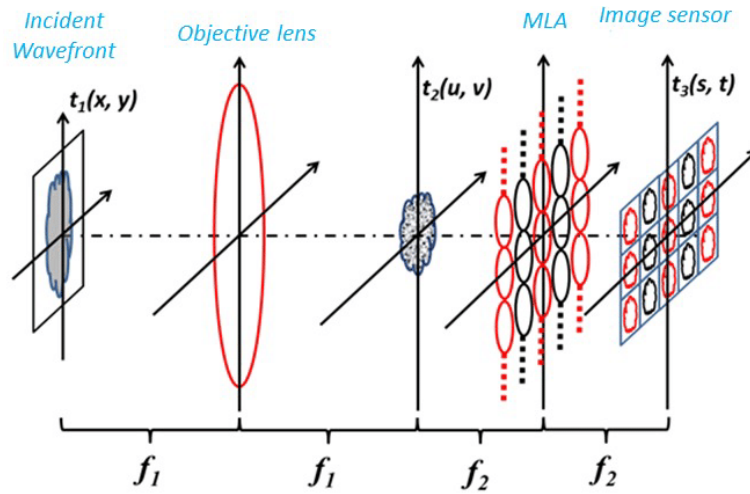


Figure 3-6: The basic elements of a 4F plenoptic sensor. The focal distances of the objective lens and MLA lenses are indicated by f_1 and f_2 , respectively. The light field complex amplitude is expressed by functions t_i , where $t_1(x, y)$, $t_2(u, v)$ and $t_3(s, t)$ correspond to the complex amplitude of the object lens front and back focal plane and MLA image plane, respectively [51].

37, 47]. The structure of this section is the same as in Section 3-2. The working principles of the plenoptic sensor will be elaborated first, discussing the image formation and phase reconstruction. The design drawbacks and possible adjustments are discussed next, followed by derivation of the performance metrics for the plenoptic sensor.

The physical image obtained from a plenoptic WFS is shown in Figure 3-7, using the same annular aperture as in Figure 3-3 and Figure 3-4. Comparing Figure 3-7 shows clear differences between the plenoptic and SH sub-images, details that will be discussed below.

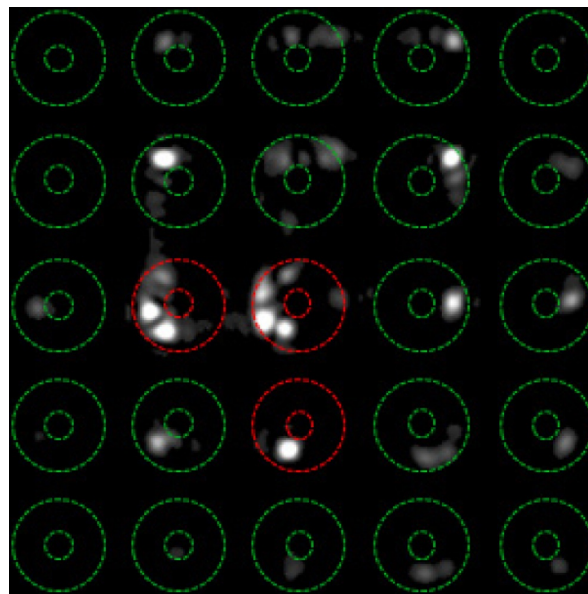


Figure 3-7: A zoomed-in plenoptic image that shows the sub-images and which parts of the aperture plane are imaged in each sub-image, depending on the slopes in the aperture [18].

3-3-1 Working Principles

Image Formation As explained in [36] the MLA samples light such that each lenslet images the aperture plane from a slightly different point of view, depending on the angle of arrival. Figure 3-8 illustrates the image formation of three aberration-free plane wavefronts, each emitted incoherently from a star at optical infinity. The aperture plane of the plenoptic sensor is located f_{obj} in front of the objective lens.

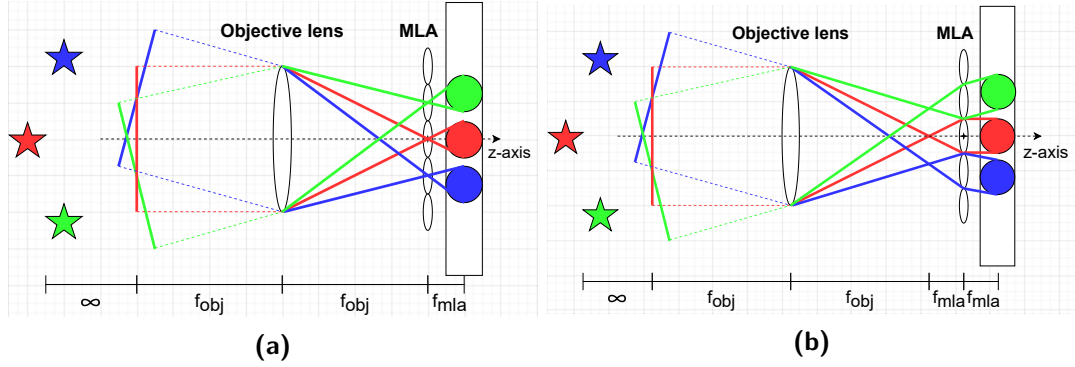


Figure 3-8: Image formation in the P3F (a) and P4F (b) sensors for different tilted plane waves. The objective lens separates the incoming wavefronts depending on the angle of arrival and three sub-images are fully illuminated. The process is shown for both the 3F (a) and 4F (b) setups.

The aperture plane contains three tilted plane waves whose angles of arrival depend on the angular location of the star. These angles, which can be related to phase gradients through approximation (A-6), result in each plane wave to be focussed by different lenslets and thus three sub-images are illuminated. With light being considered incoherent, each of the three plane waves fully covers the aperture and thus each sub-image is fully illuminated.

By means of *principal rays*, defined by [29], one can associate each lenslet with an angle of arrival as illustrated by Figure 3-9. The collection of these angles for all lenslets in the MLA is referred to as the *quantized angular spectrum* [18]:

$$(\alpha, \beta) = \frac{d_{mla}}{L} (m, n), \quad (3-10)$$

where d_{mla} is the lenslet size and L is the distance between objective lens and MLA, equal to f_{obj} and $f_{obj} + f_{mla}$ for the 3F and 4F configurations respectively. Indices m and n quantify the displacement for each lenslet with respect to the optical axis, in x - and y -directions. This means that for a square $M \times M$ MLA with uneven M the centre lenslet is given by $(m, n) = (0, 0)$, the one above it $(1, 0)$ and the lenslet in the top-left corner by $(-\frac{M-1}{2}, \frac{M-1}{2})$.

The quantized angular spectrum (3-10) can be interpreted by drawing the principal ray from the objective lens to each lenslet, which is illustrated for three lenslets in Figure 3-9. If a tilted plane wave is focussed exactly in the lenslet centre, for the 3F configuration, or in the front focal plane along its principal ray, for the 4F configuration, one obtains the same image formation as illustrated by Figure 3-8. The angle between principal ray and optical axis is equal to the wavefront gradient (A-5) and thus proportional to the phase gradient (A-6). The

geometric centres of each sub-image can be determined using the angular spectrum and the sensor geometry, as shown in Figure 3-9.

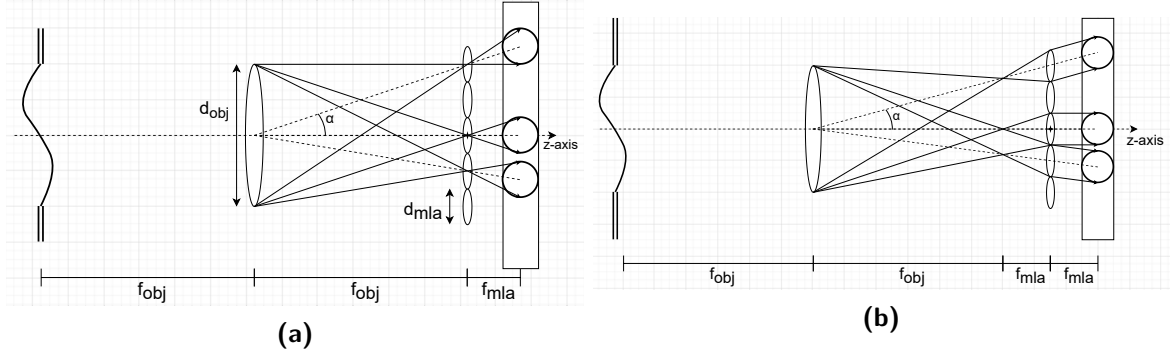


Figure 3-9: Illustration of the geometrical locations of several plenoptic sub-images, shown for both the P3F (a) and P4F (b) sensors.

In practice the aberration-free example of Figure 3-8 is not feasible. A more realistic scenario is illustrated by Figure 3-10, where atmospheric turbulence introduces an arbitrary aberration profile. This profile is indicated by the black dashed lines, located at some distance from the aperture plane in Figure 3-10. One point source is considered at optical infinity that emits a plane wave. The turbulence affects this wave such that the wavefront in the aperture contains three distinct slopes, indicated by the coloured line-segments. Analogous to Figure 3-8 these segments are imaged by separate lenslets and end up on separate sub-images. However, as the slopes only cover a part of the aperture the sub-images are only partly illuminated, as opposed to the example in Figure 3-8.

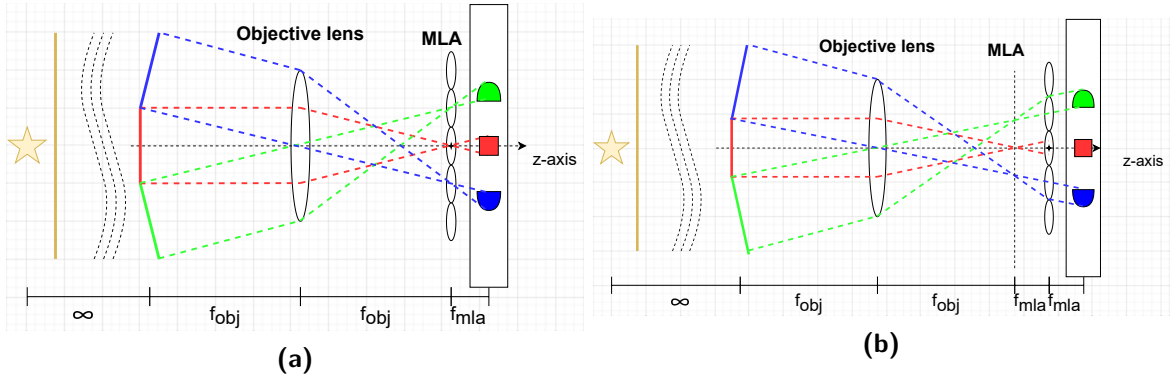


Figure 3-10: Image formation in the P3F (a) and P4F (b) sensors for an aberrated wavefront. The objective lens separates the light based on the wavefront slopes and three partly-illuminated sub-images appear on the image sensor. The illustration is inspired by Figure 3 and Figure 4 in [18].

Examples Figure 3-8 and Figure 3-10 illustrate how slopes in the aperture plane affect the patterns in the sub-images. To determine which pixels on the sensor image belong to which sub-image one needs the geometric centre as well as its size. Determining this size has been thoroughly discussed in literature, using either the *image-side f-number* [1, 30, 36] or

numerical aperture [18, 47]. Under the paraxial approximation (A-2) both can be reduced to the same equality:

$$\frac{d_{obj}}{L} = \frac{d_{mla}}{f_{mla}}, \quad (3-11)$$

where L is the separation between objective lens and MLA, equal to f_{obj} and $f_{obj} + f_{mla}$ for the 3F and 4F configurations respectively. If this equality is upheld, the size of sub-images will be maximized without overlap [30, 47] and be equal to the lenslet size d_{mla} . This equality is given physical meaning in [30], stating that the sub-images are cropped if the left-hand side of (3-11), the objective lens side, is smaller than the MLA side. Conversely if the objective lens side is larger the sub-images will be larger than d_{mla} and may overlap, introducing *cross-talk* to the image. Throughout this thesis (3-11) will be referred to as the *plenoptic equality*.

To extract slope measurements from the plenoptic sensor image and perform subsequent reconstruction one needs to address the relation between aperture plane and sensor image, defined through sensor geometry. These steps will be detailed below.

Phase Reconstruction Figure 3-11 shows that each pixel in the plenoptic image is related to a specific region in the aperture plane. As stated in [36], all rays that pass through the green square in the pupil arrive at one of the sub-images depending only on the angle of arrival, quantified by (3-10). This means that illuminated pixels are assigned angles based on their sub-image location, revealing the wavefront slopes of corresponding spatial locations in the aperture.

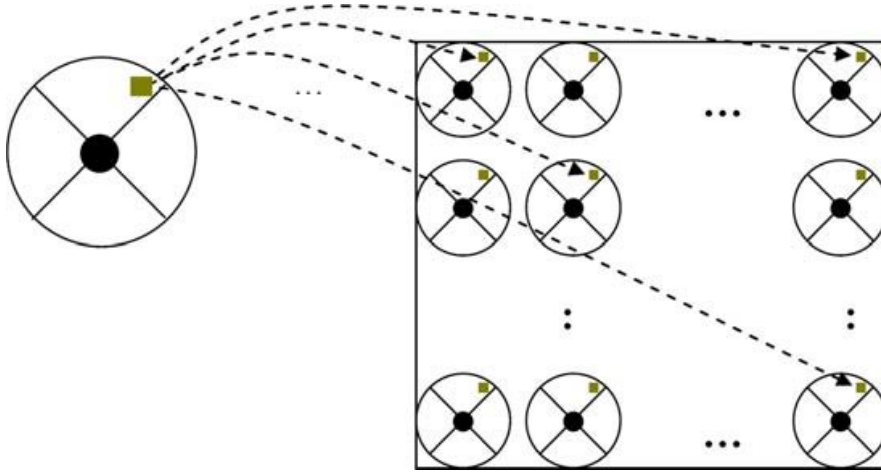


Figure 3-11: Correspondence between pupil and pupil images. Every pupil coordinate is re-imaged on the corresponding position of each pupil image, depending on the arriving angle of the incoming ray [36].

For purposes of consistency with Section 3-2-1 each lenslet in the MLA and corresponding sub-image is associated with index-pair (i, j) . Each pixel in a sub-image is associated with index-pair (s, t) , in the same fashion as illustrated in Figure 3-2 for the SH sensor. This means that a similar intensity function $I^{plen}(i, j, s, t)$ can be defined for the plenoptic image, where each pixel is represented by indices (i, j, s, t) .

The microlens array in the plenoptic sensor has a different function than in the array in the SH sensor, which is reflected in the information encoded in the image pixels. The SH MLA spatially samples incoming wavefronts, as explained in Section 3-2-1. In the plenoptic sensor the objective lens focusses incoming light onto the MLA and individual lenslets image it based on the angular spectrum (3-10). As such the microlens array performs *angular sampling* of incident wavefronts. Each pixel in plenoptic sub-image (i, j) is assigned the corresponding angular spectrum, whereas its position (s, t) in said sub-image defines the region in the aperture plane this pixel refers to. This means that in the plenoptic intensity $I^{plen}(i, j, s, t)$ *spatial information* is encoded by index-pair (s, t) and *angular information* by (i, j) , opposite to the SH image encoding.

The SH WFS design enables one to obtain slopes that represent a two-dimensional spatial location, as addressed in Section 3-2-1. In the plenoptic image each sub-image is related to the entire two-dimensional aperture, such that the plenoptic slope measurements represent a four-dimensional entity: a two-dimensional spatial location and two-dimensional angular information. Recent literature describes two different models that use this information for slope acquisition, one for the CAFADIS camera by Rodríguez-Ramos et al. [36, 37, 38] and one using geometrical ray tracing by Davis, Ko and Wu [18, 51, 52], both elaborated below.

Slope Acquisition 1 - CAFADIS Model The model used by Rodríguez-Ramos et al. in [36, 37, 38] employs a modified image first moment calculation to find the slopes, using the relation between pupil and image in Figure 3-11. This approach is based on the slope model of Clare and Lane [10], which in turn is a generalized version of Ragazzoni's method of retrieving the slopes for a pyramid WFS. The lenslet and sub-image pixel notation, associated with indices (i, j) and (s, t) respectively, is retained from previous notations. The modification of the image first moment equations (3-1) is manifested in averaging over all sub-images, instead of pixels in each sub-image. Through geometry the obtained slopes are the gradients of the wavefront in the aperture plane, which are found by:

$$S_x(s, t) = \frac{-d_{mla}}{L} \frac{\sum_{i=1}^M \sum_{j=1}^N \delta_M(j) I_{ij}(s, t)}{\sum_{i=1}^M \sum_{j=1}^N I_{ij}(s, t)}, \quad (3-12)$$

$$S_y(s, t) = \frac{d_{mla}}{L} \frac{\sum_{i=1}^M \sum_{j=1}^N \delta_N(i) I_{ij}(s, t)}{\sum_{i=1}^M \sum_{j=1}^N I_{ij}(s, t)},$$

where the lenslets are represented by index-pair (i, j) in an $M \times N$ MLA. In a similar fashion to (3-2) direction inversion is corrected for. Additionally the row- and column indices (i, j) are switched as they correspond to $-y$ - and $+x$ -directions respectively. Each sub-image contains intensity pattern $I_{ij}(s, t)$. The δ -function converts lenslet indices (i, j) to index-displacements from the optical axis, such that (3-12) is in accordance with angular spectrum (3-10):

$$\delta_A(k) = k - \frac{A+1}{2}, \quad (3-13)$$

The CAFADIS model uses the full plenoptic image to obtain two-dimensional slope information of the aperture plane, similar to the SH centroid method [18]. This suggests that the CAFADIS slope acquisition is not optimal, since the four-dimensional information is used to obtain information on a two-dimensional spatial location.

Slope Acquisition 2 - Ray Tracing Model The model used by Ko, Davis and Wu in [18, 51, 52] is based on geometrical optics, where the information encoded in each plenoptic pixel defines a distinct light ray. Davis, Ko and Wu have used several algorithms in their work on the plenoptic sensor, with the 'Single phase screen reconstruction' algorithm being the most straightforward and applicable one. This algorithm does not place the slopes in the aperture plane, as the SH and CAFADIS acquisition models do, but assumes atmospheric turbulence is described by a dominant two-dimensional aberration profile located somewhere along the optical path. This model is illustrated in Figure 3-12, where a two-dimensional dominant turbulence affects a collimated beam of light rays. The aberrated light rays that arrive at the plenoptic aperture plane are represented by lower-case dash-dotted \mathbf{k} -vectors. The bottom half of Figure 3-12 shows the AO-corrected step, where the dominant turbulence is reconstructed and compensated for, such that light rays passing this turbulence arrive at the plenoptic sensor approximately collimated, represented by upper-case vectors \mathbf{K} .

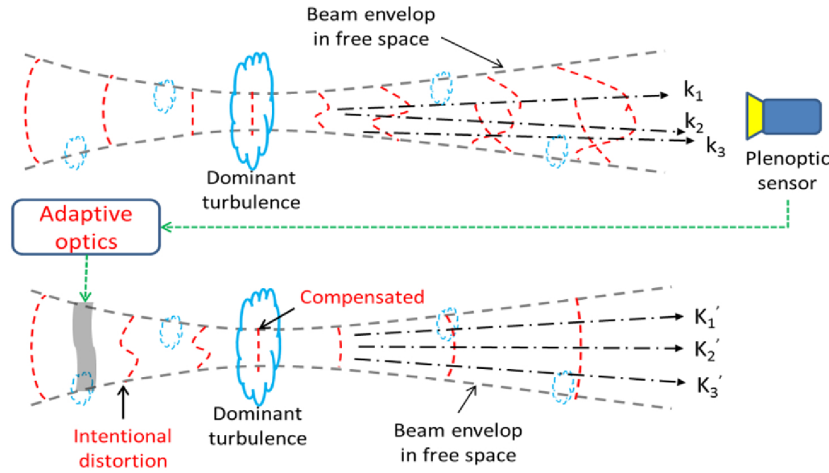


Figure 3-12: An illustration of the Ray Tracing model for the plenoptic sensor, showing how individual light rays travel from dominant turbulent phase screen to the sensor. The top half shows how the aberrated rays arrive at the plenoptic sensor. The bottom half shows the closed-loop with an AO system that corrects for this phase screen. Taken from [52]: "Phase compensation algorithm diagram for beam distortion caused by dominant turbulence occurring over a short distance"

The algorithm is described by the following steps, interpreted from [52]. The final step stated in [52] describes the phase reconstruction and is removed from these steps, as phase reconstruction is considered later in this section.

1. Record the plenoptic image and obtain separate sub-images, one for each lenslet.
2. Transform all pixels into a cluster of vectors with endpoint the spatial pixel location (proportional to (s, t)) and direction equal to the angular spectrum of their corresponding sub-image (proportional to (i, j)). The thickness of the vector is proportional to intensity value $I^{plen}(i, j, s, t)$, the ray intensity.
3. Adjust the vector directions according to relative locations in the sub-image due to the "vignetting" effect.

4. Back-propagate the vectors through the MLA, objective lens up to the location of the dominant turbulence.
5. From the back-propagated cluster remove rays that fall beyond a reasonable spatial range or have abruptly different directions with respect to their neighbours.
6. Propagate the remaining cluster of vectors to the aperture plane of the plenoptic sensor, assuming no turbulence affects the light rays between dominant screen and aperture plane.
7. Combine both clusters of light rays and extract their directions to form the wavefront gradients of the dominant turbulence.

Finally, after obtaining the sensed slopes with either the CAFADIS or Ray Tracing model one can invoke approximation (A-6) to obtain the phase gradients. The gradients allow one to reconstruct the phase aberration using either the modal reconstruction algorithm elaborated in Appendix A-5 or the zonal algorithm elaborated in Appendix A-6. The complete algorithm of the Ray Tracing model is detailed in Section 5-2-10 with an example illustration.

There are two major differences between the phase reconstruction of the CAFADIS and Ray-Tracing models. First, the CAFADIS model produces a guaranteed number of detected slopes due to averaging the pixels over all sub-images. The Ray Tracing model does not offer such a guarantee but Ko and Davis report a minimum number of gradients in [18] as a tractable alternative. This minimum is equal to the guaranteed number of the CAFADIS model, whereas the maximum goes up to the total number of illuminated pixels on the whole sensor image. More details on the number of obtained slopes is found in Section 3-3-4.

The second difference is that the Ray Tracing model reconstructs the phase aberration at the location of assumed dominant turbulence. This model accounts for light propagation through the sensor but also in free space before it arrives at the aperture plane. On the other hand, the CAFADIS model places the slopes and its reconstruction in the aperture plane of the plenoptic sensor, without considering free space propagation. At first glance it may seem convenient to explicitly include free space propagation, but this also means that all light propagation is fixed in the domain of geometrical optics. This means that diffraction and other wave-nature effects are not considered in the Ray Tracing model. The CAFADIS model enables free choice of the propagation model, leaving more design flexibility for implementation.

3-3-2 Sensor Drawbacks

The plenoptic sensor is significantly more complex in its design but does share some design aspects with the SH sensor, so similar drawbacks can be expected. For example, the plenoptic sensor is also limited by the physical dimensions of the MLA, but now in terms of angular resolution. Each lenslet effectively senses one slope according to the angular spectrum (3-10), such that the number of detectable slopes is reduced to the number of lenslets in the MLA. Compared to the SH design it is easier to detect stronger slopes by simply adding more lenslets to the MLA, increasing the angular range. Conversely, the plenoptic sensor has more difficulty detecting smaller slope variations, since the difference between angles in the angular spectrum (3-10) is limited by the lenslet size.

Another major drawback is the plenoptic equality (3-11) that prevents cross-talk in the plenoptic image [1, 18, 30, 47]. This equality constrains the sizes and focal distances of all lenses in the plenoptic sensor, restricting the degrees of freedom in designing the plenoptic sensor.

A third drawback is the finite size of the objective lens that limits the size of the wavefront to be sensed. The physical dimensions of the objective lens define the aperture plane and determines which part of incident light ends up on the image. The SH WFS experiences the same drawback but for the physical dimensions of the MLA. In practice the MLA is larger than the objective lens and the SH sensor is less constrained in design than the plenoptic sensor, which indicates that the plenoptic sensor may suffer more from this drawback.

Finally, the plenoptic sensor suffers from the general slope-based WFS drawbacks discussed in Section 3-2-2, such as light economy, image sensor noise and the inability to detect discrete jumps in the wavefronts [7, 9, 10, 11, 25, 26].

3-3-3 Adjustments to the Plenoptic Sensor

In a similar fashion to the SH sensor, a number of adjustments for the plenoptic WFS design can be noted, divided into software- and hardware based. The listed improvements do not cover all possibilities but provide an overview of possible adjustments to the current design.

Software Considering the similarities between SH and plenoptic sensors, software-based improvements implemented on one can often be carried over to the other, with minor alterations.

One such example is reshuffling the pixels in the plenoptic image to separate the images of multiple point sources, analogous to blob detection described for the SH sensor. This reshuffling is based on the equivalence between the SH and plenoptic WFS from a geometrical optics point of view. The pixels on the plenoptic image are reshuffled by switching lenslet indices (i, j) with pixel indices (s, t) , switching the type of information encoded in the pixels. This results in a Hartmannogram-like image, enabling one to reconstruct the phase for multiple imaged point sources, as was described for the SH sensor. This pixel reshuffling is implemented in the toolbox and further elaborated in Section 5-2-6.

Other adjustments are improvements to the reconstruction speed listed by [51]. Here Wu obtains the slopes from the Ray Tracing model and notes that the full reconstruction algorithm is reliable but computationally demanding. Several scenarios and applications do not require the full knowledge of the reconstruction but profit from fast algorithms. One of these algorithms discussed is the 'Tree Reconstruction Algorithm' that uses graph theory and a fixed number of DM actuators to select the most informative pixels in the plenoptic image, using a minimal number of measurements for reconstruction and correction. Another algorithm listed is the "Checkerboard Reconstruction Algorithm" that is based on the same plenoptic reshuffling principle as explained above, but is again used to obtain the most informative pixels rather than separation of multiple imaged point sources.

Hardware The first drawback above stated the low sensitivity of the plenoptic WFS when compared to the SH sensor. For weak aberrations all light will be collected by the centre lenslet or its neighbours, such that small slope variations will remain undetected. Vdovin,

Soloviev and Loktev discussed the addition of a scattering pupil function in [47], to improve light distribution over the MLA. By increasing the number of illuminated pixels one decreases the reconstruction error for a single detected slope.

This approach is illustrated in Figure 3-13 for the original plenoptic sensor (left), the sensor with a scattering pupil instead of an objective lens (middle) and the plenoptic sensor with a scattering pupil directly behind the objective lens (right). Vdovin, Soloviev and Loktev concluded that the scattering pupil function preserved the wavefront uniqueness and could reconstruct the original light ray directions.

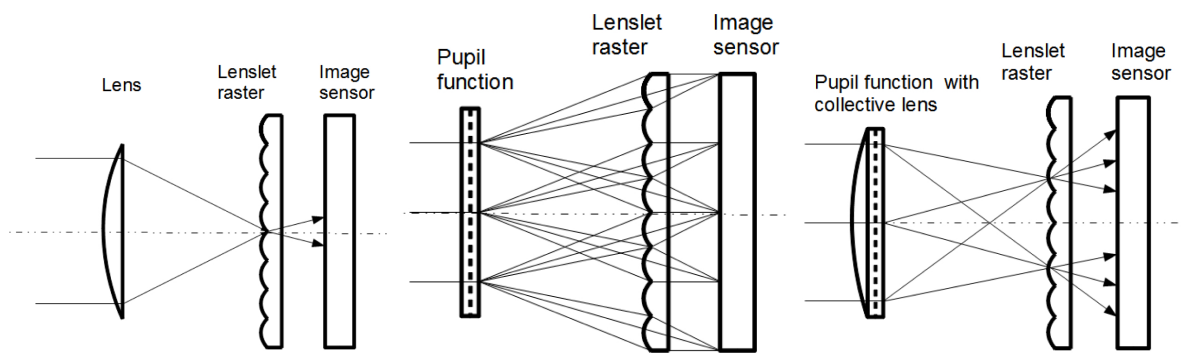


Figure 3-13: Taken from [47]: "Plenoptic WF sensor described in [10] (left), WF sensor with scattering pupil (middle), same with a collective lens (right)"

Another hardware-based adjustment for the plenoptic sensor is proposed in [38] where the phase reconstruction is handled by a custom processing chip referred to as a Field Programmable Gates Array (FPGA) instead of a Graphics Processing Unit (GPU). These chips can be specifically tailored for the computations at hand, speeding up the signal processing capabilities significantly. Naturally these chips can be implemented in the SH design, as reported by [26].

Finally, the light beam focussed by the objective lens can be modulated as illustrated by Figure 3-14 and elaborated in [32]. This adjustment has been reported for the pyramid WFS as well [2, 10, 35], hinting at similarities between both sensors. Modulation is accomplished by rotating the beam using a tip-tilt mirror conjugated to the aperture plane. It was concluded by [32] that the linearity of the plenoptic sensor was roughly similar to the SH linearity when modulation was added, greatly improving the plenoptic sensor sensitivity.

The study conducted by [32] compares the plenoptic and SH sensors on the minimum detectable slopes. This is an interesting perspective and as such this work will be further examined in the comparison between the SH and plenoptic sensor.

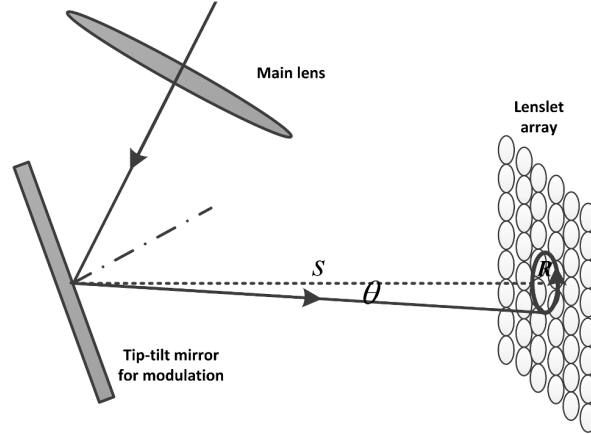


Figure 3-14: Modulation for the plenoptic sensor. The modulation amplitude and angle are given by R and θ respectively, with s the optical axis from the centre of the mirror through the centre of the MLA. Taken from [32]: "The schematic diagram of the circular modulation"

3-3-4 Performance Metrics

To compare the plenoptic and SH wavefront sensors, the same performance metrics discussed in Section 3-2-4 and quantified by [18, 47] should be used. In this section the metrics are elaborated for the plenoptic sensor assuming that the small angle approximation holds (A-2).

Dynamic range The maximum detectable wavefront tilt that can be detected is limited by the finite size of both image sensor and MLA. Each lenslet corresponds to an angular coordinate through (3-10) and thus the dynamic range lies at the maximum of this quantified angular spectrum. If the MLA consists of an $M \times N$ lenslet grid, the maximum detectable tilt is found to be:

$$\left. \frac{\partial w(x, y)}{\partial u} \right|_{max}^{plen} = \frac{d_{mla}}{L} (m, n) \Big|_{max} = \frac{d_{mla}}{L} \frac{\max(M, N) - 1}{2}, \quad u = \{x, y\}, \quad (3-14)$$

where $w(x, y)$ is the incoming wavefront and L is separation between objective lens and MLA, equal to f_{obj} and $f_{obj} + f_{mla}$ for the 3F and 4F configurations, respectively.

Sensitivity Both slope acquisition models place the wavefront slopes based on the sub-image location and angular spectrum (3-10). Since this spectrum is expressed as a function of indices m and n , the plenoptic sensitivity can be described by taking the difference between spectra, or in other words:

$$\left. \frac{\partial w(x, y)}{\partial u} \right|_{min}^{plen} = \frac{d_{mla}}{L}, \quad u = \{x, y\}. \quad (3-15)$$

Gradient sample size The plenoptic gradient sample size follows the same definition as the SH gradient sample size: the number of phase gradients collected from a single sensor image. The CAFADIS slope model finds the aperture plane slopes by averaging over all sub-images,

such that the number of slopes is equal to the number of illuminated pixels in a single sub-image:

$$N_{gradient}^C = \frac{\pi}{4} \left(\frac{d_{mla}}{d_{pix}} \right)^2, \quad (3-16)$$

where d_{mla} and d_{pix} are the spatial sizes of the lenslets and image sensor pixels, respectively, both expressed in [m]. The premultiplication of $\pi/4$ takes a circular objective lens into account. If the objective lens is square this premultiplied factor should be left out.

The Ray Tracing model treats each illuminated pixel as an individual light ray and a potential slope measurement, but the actual gradient sample size can only be determined after propagating all rays back to the dominant turbulence location and discarding unrealistic rays. This dynamic metric is noted by [18] and a minimum sample size is reported as tractable alternative. This minimum is equal to the number of illuminated pixels if only one lenslet images the incoming light, which is equal to the actual gradient sample size of the CAFADIS model.

$$N_{min,gradient}^{RT} = \frac{\pi}{4} \left(\frac{d_{mla}}{d_{pix}} \right)^2. \quad (3-17)$$

Conclusion of Performance Metrics Literature reports the dynamic range of the plenoptic sensor as a significant improvement on the SH design, which is reflected in (3-14). Simply by adding more lenslets to the MLA, or increasing the lenslet size the dynamic range can be increased.

The plenoptic sensitivity is more problematic on the other hand, as it is determined by the ratio of lenslet size with distance between objective lens and MLA, d_{mla}/L . The sensitivity can be improved either by considering smaller lenslets or increasing the focal length of the objective lens and/or MLA. Unfortunately, the plenoptic equality (3-11) prevents one to freely change either d_{mla} or L , without changing the other lens parameters, making a decrease in the sensitivity difficult.

3-4 Comparison of the Shack-Hartmann and Plenoptic Sensors

Taking the sets of equations in Section 3-2-4 and Section 3-3-4 into account, a comparison between the SH and plenoptic sensor can be performed. Before arriving at such a comparison a few practical restrictions are set to provide helpful insights. These restrictions are not to be kept strict, rather provide insight on what to expect in practical cases. After briefly addressing these practicalities, the performance metrics are placed parallel and a comparison is made. This details section three comparisons that support each other toward the conclusion. First, a theoretic discussion is performed on the defined performance metrics in Section 3-2-4 and Section 3-3-4. Then, two practical comparisons are included, one being the work of Ko and Davis [18] that uses very similar metrics and one the work of Pengzhi, Jieping, Yonghui and Hongjun [32] that investigates the sensitivity of both sensors.

3-4-1 Practical Restrictions

Lens and Pixel Sizes From previous figures and schematics depicting the plenoptic sensor it is clear to see that the objective lens is the largest lens in the sensor. The MLA lenslets are in their turn larger than the pixels that make up the image sensor. Considering this, one can define a practical restriction to be:

$$d_{obj} > d_{mla} > d_{sens}. \quad (3-18)$$

Minimum Image Sensor Sizes Both sensors differ in the minimum size required for the image. For the SH sensor it is the same size as the MLA, but the plenoptic sensor requires a larger image sensor to fit all sub-images, as shown in Figure 3-9. Take the MLA to be an $M \times M$ grid of lenslets, with M uneven such that the centre lenslet is aligned with the optical axis. The minimum image sizes for the SH and plenoptic sensors are then found to be:

$$h_{sens}^{SH} = M \cdot d_{mla}, \quad (3-19)$$

$$h_{sens}^{plen} = d_{mla} \left(M + \frac{f_{mla} (M - 1)}{L} \right), \quad (3-20)$$

where L is the separation between objective lens and MLA, equal to f_{obj} and $f_{obj} + f_{mla}$ for the 3F and 4F configurations. One can derive (3-20) with the geometry in Figure 3-9 combined with the quantized angular spectrum (3-10) for the outer lenslet.

3-4-2 Theoretical Comparison

For clarity the performance equations derived in Section 3-2-4 and Section 3-3-4 are summarized in Table 3-1. The MLA consists of a square $M \times M$ lenslet grid each of size or pitch d_{mla} and with focal distance f_{mla} . The size of the objective lens and its focal distance are given by d_{obj} and f_{obj} , respectively. The size of each pixel, assumed square, is given by d_{pix} . The distance between objective lens and MLA in the plenoptic sensor is given by constant L , equal to f_{obj} and $f_{obj} + f_{mla}$ for the 3F and 4F configurations respectively. The maximum and minimum detectable wavefront tilts are given by the dynamic range and sensitivity respectively. As a third performance metric the gradient sample size $N_{gradient}$ is given, the number of phase gradients each sensor outputs.

Performance metrics	Shack-Hartmann sensor:	Plenoptic sensor:
Dynamic range	$\frac{d_{mla}}{2f_{mla}}$	$\frac{(M-1)d_{mla}}{2L}$
Sensitivity	$\frac{d_{pix}}{2f_{mla}}$ (even) or $\frac{d_{pix}}{f_{mla}}$ (uneven)	$\frac{d_{mla}}{L}$
Gradient sample size	M^2	$\frac{\pi}{4} \left(\frac{d_{mla}}{d_{pix}} \right)^2$

Table 3-1: Selected performance metrics for the SH and plenoptic sensors, based on [18, 47]

The most significant improvement of the plenoptic design upon the SH sensor, as stated by literature, is the increase in dynamic range and possibility to sense more complex wavefronts.

Comparing the dynamic ranges in both sensors gives insight on the conditions to make this claim true:

$$\frac{(M-1)d_{mla}}{2L} \geq \frac{d_{mla}}{2f_{mla}} \rightarrow M \geq \frac{L}{f_{mla}} + 1.$$

Solving this inequality for both the P3F and P4F sensor configurations results in:

$$M \geq \frac{f_{obj}}{f_{mla}} + 1 \quad (3F), \quad (3-21)$$

$$M \geq \frac{f_{obj} + f_{mla}}{f_{mla}} + 1 \rightarrow M \geq \frac{f_{obj}}{f_{mla}} + 2 \quad (4F), \quad (3-22)$$

where the term f_{obj}/f_{mla} will always be larger than one, shown by rewriting the numerical aperture equality (3-11) and considering practical limitation (3-18):

$$\frac{d_{obj}}{f_{obj}} = \frac{d_{mla}}{f_{mla}} \rightarrow \frac{f_{obj}}{f_{mla}} = \frac{d_{obj}}{d_{mla}} > 1, \quad (3F),$$

$$\frac{d_{obj}}{f_{obj} + f_{mla}} = \frac{d_{mla}}{f_{mla}} \rightarrow \frac{f_{obj} + f_{mla}}{f_{mla}} = \frac{d_{obj}}{d_{mla}} > 1 \quad (4F).$$

The comparison of the dynamic ranges shows an important difference between the SH and plenoptic sensor. The plenoptic dynamic range increases as more lenslets are added to the MLA, whereas the SH dynamic range depends solely on the finite sizes of these lenslets. As long as all plenoptic sub-images fit on the image sensor one could increase the number of lenslets in the MLA, increasing the plenoptic dynamic range effortlessly. If the image sensor is limited its size will determine the maximum increase to the plenoptic dynamic range, depending on how many sub-images fit on the image sensor.

The next claim is that the plenoptic sensor can sense more complex wavefronts, meaning that more gradient samples are available for reconstruction. In a similar fashion to the dynamic range, one can compare $N_{gradient}$ for both sensors:

$$N_{gradient}^{plen} = \frac{\pi}{4} \left(\frac{d_{mla}}{d_{pix}} \right)^2 \geq M^2 = N_{gradient}^{SH},$$

where $N_{gradient}^{plen}$ represents both the CAFADIS gradient sample size and the Ray Tracing minimum sample size. The inequality can be simplified into a condition for the plenoptic gradients sample size to be larger or equal to that of the SH sensor:

$$\frac{d_{mla}}{d_{pix}} \geq \frac{2M}{\sqrt{\pi}}. \quad (3-23)$$

Within practical restriction (3-18) this inequality is not difficult to meet for the plenoptic sensor, as long as the pixels are significantly smaller

While the dynamic range and gradient sample size should be kept as large as possible, the sensitivity should be kept small in order to sense small variations in the wavefront slopes. For the SH sensor this means to have small pixels, but for the plenoptic sensor it means using small MLA lenslets. Both of these changes are prone to sensor noise, as stated in [9, 10, 25]. Clare and Lane [10] stated that the slope measurement error increases as the lens diameter decreases, partly due to a decrease in light that reaches each sub-image. This notion is also

supported by [9, 25]. Furthermore, an increase in the number of image plane detectors or pixels also increases the contribution of read noise to the measurements, as stated by [9]. On the other hand, it is reported by Vdovin, Soloviev and Loktev in [47] that the relative wavefront reconstruction error decreases as more pixels are available.

The elaboration of the theoretical comparison, as well as the analysis of performance metrics in Section 3-2-4 and Section 3-3-4, reveals a number of trade-offs in the design of both sensors which will be summarized here:

1. Increasing dynamic range in the SH sensor is best done by increasing the lenslet size, but this increases the wavefront section corresponding to the detected average slope, leading to wavefront reconstruction errors. In the plenoptic sensor the dynamic range is increased best by adding more lenslets to the MLA without serious repercussions, other than the financial impact of needing a different MLA.
2. Decreasing the SH sensitivity is best done by decreasing the physical size of the image sensor pixels, leading to stronger contributions of sensor noise. The freedom of decreasing the plenoptic sensitivity is restricted by the plenoptic equality, but is best approached by taking the smallest possible MLA lenslets. This in turn reduces the dynamic range unless more lenslets are added to the MLA.
3. The gradient sample size of the SH sensor is determined by the number of lenslets in the MLA, which is limited to the dimensions of MLA available. For the plenoptic sensor it is equal to the number of pixels contained in each sub-image, which in turn is limited to the dimensions of image sensor available.

3-4-3 Practical Comparison of Metrics

Another interesting comparison is the performance of both sensors when the same MLA is used, which was done by Ko and Davis [18]. The formulation of the performance metrics by Ko and Davis is slightly different from the equations outlined in Table 3-1. In this thesis a minimum image sensor size is considered as boundary condition, while Ko and Davis use an image sensor with fixed size and express the SH sensitivity in pixel accuracy. Despite the slight difference in formulations the obtained results documented in Table 3-2 and Table 3-3 are very similar.

Ko and Davis consider two scenarios in [18]: Scenario 1, where both sensors share the same MLA and Scenario 2, where the plenoptic MLA and objective lens are altered to match the SH performance more closely. The design parameters and resulting performance metrics have been collected in Table 3-2 and Table 3-3, for scenario 1 and 2 respectively.

The number of lenslets in the MLA has not been reported, but can be estimated by matching the parameters and metrics to equations in Table 3-1 and comparing MLA specifications online [14]. This leads to the assumptions that the MLA in Scenario 1 contains a 33×33 lenslet grid and the plenoptic MLA in Scenario 2 a 55×55 grid.

	Shack-Hartmann 1	Plenoptic 1
Image sensor pixel size d_{pix}	5.5 μm	5.5 μm
Image sensor resolution	1820 \times 1820 pixels	1820 \times 1820 pixels
MLA pitch d_{mla}	300 μm	300 μm
MLA focal length f_{mla}	18.8 mm	18.8 mm
Beam diameter	10 mm	10 mm
Objective lens diameter d_{obj}	N/A	10 mm
Objective lens focal length f_{obj}	N/A	627 mm
Sensitivity	0.293 mrad	0.479 mrad
Maximum tilt	8.00 mrad	7.97 mrad
Gradient sample size	1111	2975

Table 3-2: Design parameters and resulting performance metrics for the Shack-Hartmann and plenoptic sensor under Scenario 1: A common MLA [18].

	Shack-Hartmann 1	Plenoptic 2
Image sensor pixel size d_{pix}	5.5 μm	5.5 μm
Image sensor resolution	1820 \times 1820 pixels	1820 \times 1820 pixels
MLA pitch d_{mla}	300 μm	185 μm
MLA focal length f_{mla}	18.8 mm	11.5 mm
Beam diameter	10 mm	10 mm
Objective lens diameter d_{obj}	N/A	10 mm
Objective lens focal length f_{obj}	N/A	622 mm
Sensitivity	0.293 mrad	0.298 mrad
Maximum tilt	8.00 mrad	8.01 mrad
Gradient sample size	1111	1131

Table 3-3: Design parameters and resulting performance metrics for the Shack-Hartmann and plenoptic sensor under Scenario 2: The plenoptic MLA is changed to match the performance of the SH sensor from Scenario 1 in Table 3-2 [18].

With the performance metrics matched in Scenario 2, one could improve the plenoptic dynamic range by adding more lenslets to the MLA. The gradient sample size can be improved by increasing the image sensor resolution, without compromising the sensitivity that is matched to the SH sensor. On the other hand, increasing the number of pixels on the image sensor improves the SH sensitivity - assuming one uses the same pixel size in both sensors.

Ko and Davis [18] conclude that in the matched case of Scenario 2, Plenoptic Sensor 2 outperforms Shack-Hartmann Sensor 1. Under strong turbulent conditions and complex wavefronts the SH image first moment method is prone to branch-cuts, cross-talk and other turbulent effects. The plenoptic design is much less susceptible to these problems and will thus perform better in this case, despite performance metrics being matched.

Another conclusion that can be drawn from this study is that given the exact same MLA, a plenoptic sensor will not outperform the SH sensor in terms of dynamic range and sensitivity.

3-4-4 Comparing Linearity

The linearity of both sensors has been investigated in [32], where the sensed slope is compared to the input wavefront slope. The comparison was performed using a SH sensor with an 16×16 MLA and sub-images containing 8×8 pixels. The plenoptic sensor was used without modulation, had a 8×8 MLA and 16×16 pixels in the sub-images, such that the gradient sample size was equal in both sensors. Furthermore, the comparison was undertaken in open-loop and for different lenslet sizes, shown in Figure 3-15.

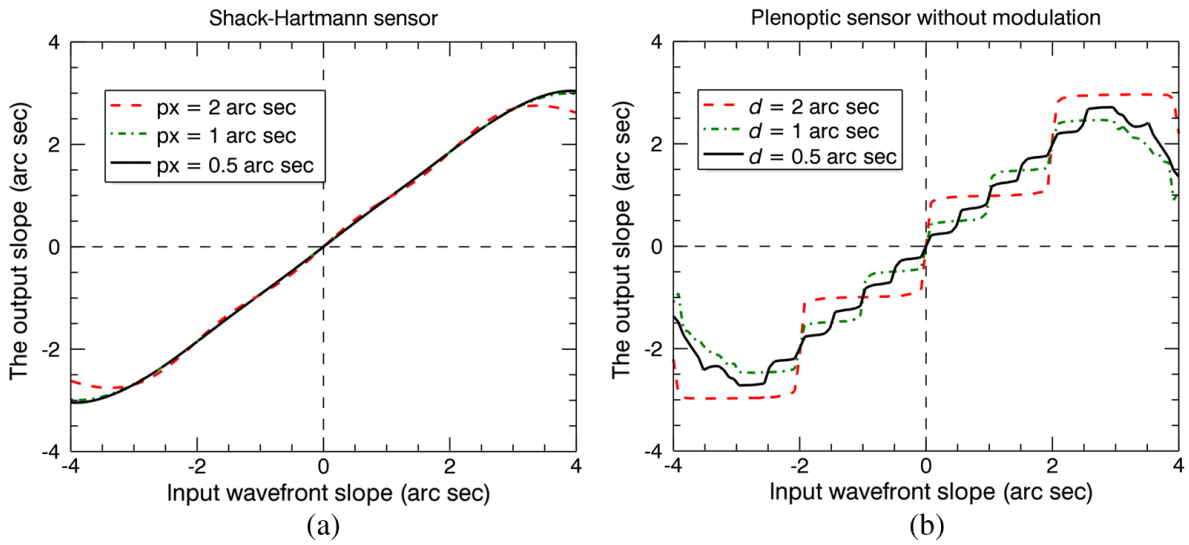


Figure 3-15: Comparison of the SH and plenoptic sensor linearity as a function of MLA lenslet size, depicted as px and d , expressed in arcseconds. Taken from [32]: "Linearity of the wavefront sensor: (a) Shack-Hartmann sensor and (b) the plenoptic sensor without modulation".

By adding modulation one can improve the sensitivity of the plenoptic sensor, as was described in Section 3-3-3. This adjustment is based on the similar improvement in the pyramid WFS [2, 9, 32]. The sensitivity improvement is further investigated by [32], where modulation lead to a significant increase of the linearity, up to the point that the linearities of both SH and modulated plenoptic sensor were roughly the same. The improved linearity of the plenoptic sensor is shown in Figure 3-16 for a lenslet size of 1 arc sec or $194 \mu\text{m}$. It is readily seen that adding the slightest modulation improves the sensitivity. However, further improving the linearity is detrimental for the sensing of the slopes, as can be seen by the green line in Figure 3-16 for a modulation amplitude of $d = 194 \mu\text{m}$. It is concluded by [32] that modulation does improve the plenoptic linearity, but should not be chosen too large.

The closed-loop correction performance was also investigated for both sensors, shown in Figure 3-17 for the unmodulated and in Figure 3-18 for the modulated plenoptic sensor. The modulation amplitude is not explicitly mentioned but based on the results in Figure 3-16 one should assume the amplitude lies between $0.125d$ and $0.5d$.

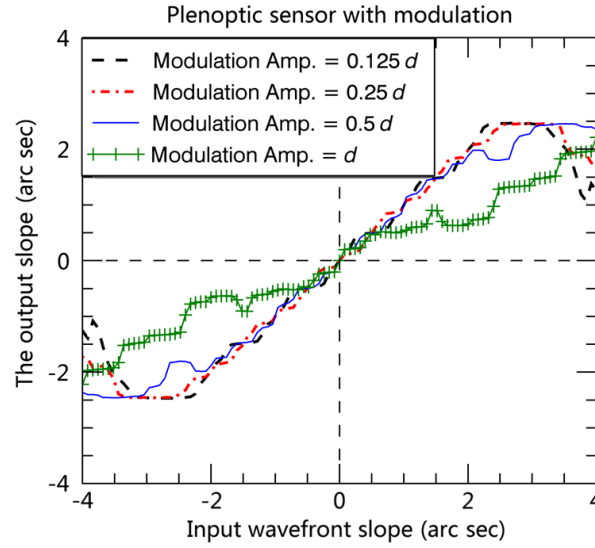


Figure 3-16: Taken from [32]: "Linearity of the plenoptic sensor with circular modulation".

In each figure the root mean square (RMS) error of the residual aberration is shown on the vertical axis, with the stellar magnitude of the imaged object on the horizontal axis. Objects with a larger stellar magnitude are less bright leading to an increase in the RMS error. Furthermore the strength of turbulence is given by D/r_0 , the ratio of telescope diameter and the Fried parameter. This ratio is varied across the three figures in of Figure 3-17 and Figure 3-18, where a higher value indicates stronger turbulence.

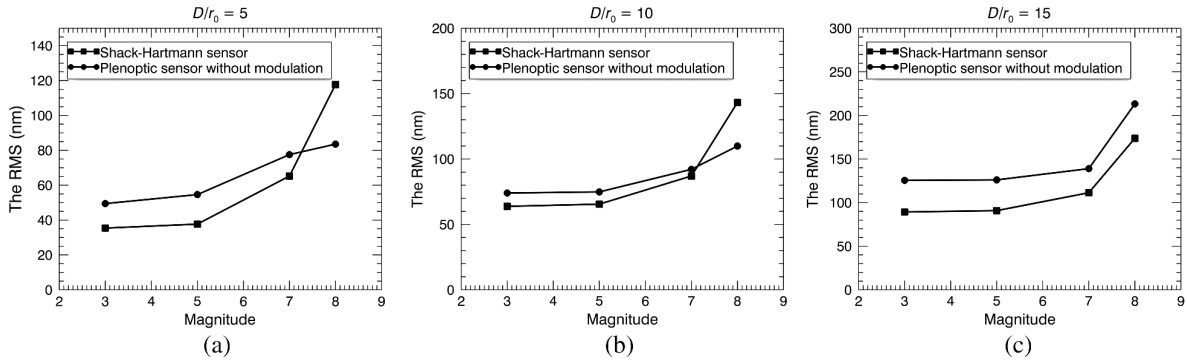


Figure 3-17: Results obtained from [32]: "Comparison of the RMS (nm) of Shack-Hartmann and plenoptic sensor without modulation for different magnitudes, in turbulence of severity: (a) $D/r_0 = 5$, (b) $D/r_0 = 10$, (c) $D/r_0 = 15$."

It should be noted that this comparison tests the sensitivity of both sensors, thus will not reflect the dynamic range benefits of the plenoptic sensor. This is also observed in the closed-loop performance, where the SH sensor clearly outperforms the unmodulated plenoptic sensor.

According to [32], it is reasonable that the Shack-Hartmann sensor performs better, because it has a better linearity and thus sensitivity compared to the plenoptic linearity. Only for stars fainter than magnitude 7 (81 photons per sub-image) the plenoptic sensor catches up - save for the case of strong atmospheric turbulence with $D/r_0 = 15$.

Modulation in the plenoptic sensor improves its closed-loop performance to roughly equal that of the Shack-Hartmann sensor. Unfortunately it performs worse than the SH sensor with decreasing brightness. Again, this is reasonable according to [32] as the modulated plenoptic sensor is significantly more sensitive to signal-to-noise ratio (SNR). Modulation disperses the spot on the plenoptic image, meaning that for low brightness and thus low SNR the reconstruction error in the modulated plenoptic sensor increases more drastically.

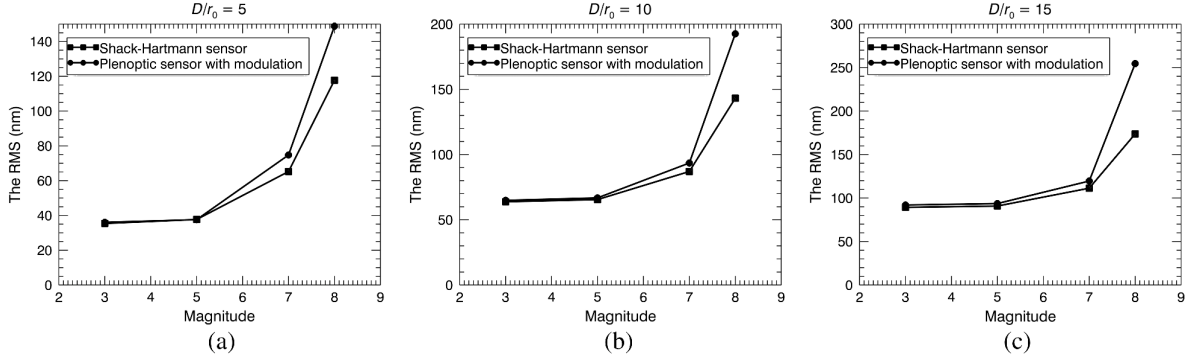


Figure 3-18: Results obtained from [32]: "Comparison of the RMS (nm) of Shack-Hartmann and plenoptic sensor with modulation for different magnitudes, in turbulence of severity: (a) $D/r_0 = 5$, (b) $D/r_0 = 10$, (c) $D/r_0 = 15$."

3-4-5 Conclusion

The conducted and reported investigations do not point out which WFS is better. The design of the plenoptic sensor facilitates relatively easy adjustments to the dynamic range, but it will suffer in the presence of weaker aberrations due to low sensitivity. The comparative studies in [18, 32] have shown that the SH sensor still outperforms the plenoptic sensor in experiments, unless specific adjustments or considerations are taken, such as decreasing lenslet size, adding more lenslets to the microlens array or modulating the beam focussed on the plenoptic MLA. The range of adjustments is determined by the freedom of the MLA design.

The adjustments can be quantified through a comparison of the performance metrics. Equating the metrics dynamic range, sensitivity and gradient sample size for both sensors results in an inequality for each metric:

$$\begin{aligned}
 M &\geq \frac{L}{f_{mla}} + 1, \\
 \frac{d_{pix}}{2f_{mla}} &\geq \frac{d_{mla}}{L}, \\
 \frac{d_{mla}}{d_{pix}} &\geq \frac{2M}{\sqrt{\pi}},
 \end{aligned} \tag{3-24}$$

where satisfaction to one inequality means the plenoptic sensor outperforms the Shack-Hartmann sensor on that metric.

This concludes the literature survey contributed to wavefront sensing. The next chapter will discuss a brief study in aberrations and atmospheric turbulence, introducing the concepts of *isoplanatic* and *anisoplanatic aberrations*. The simulation toolbox, the author's own contribution, is presented in Chapter 5.

Isoplanatic versus Anisoplanatic Aberrations

The principles of wavefront sensors discussed in Chapter 3 reconstruct the phase in the aperture plane using the sensor image, but no effects of external turbulences have been taken into account. For purposes in astronomy and telescopes these external influences originate from layers of *atmospheric turbulence*. The goal of this chapter is to highlight important concepts of this form of turbulence and present the reader with solutions of modelling atmospheric turbulence. Of these solutions one model is selected to simulate turbulence in the simulation toolbox, further elaborated in Chapter 5. Furthermore, the effect of *anisoplanatism* is introduced and how it is accounted for in the selected model for atmospheric turbulence.

4-1 Atmospheric Turbulence

As stated by [52], atmospheric turbulence is a natural phenomenon causing inhomogeneities in the refractive index of air. As light travels through such inhomogeneous media the directions of light rays are adversely affected. In terms of wave optics these inhomogeneities introduce phase aberrations to the light [33]. The nature of atmospheric turbulence is a complex, highly dynamic and statistical phenomenon as indicated by [6, 13, 17, 51]. For the purpose of this thesis it is only highlighted in a few important equations, but for profound elaborations the reader is referred to aforementioned sources.

Kolmogorov Statistics A common statistical model to describe atmospheric turbulence is *Kolmogorov theory* that assumes small-scale structures in turbulent air to be locally homogeneous and spatially isotropic [6, 13, 17]. To quantify the strength of turbulence one uses the *refractive-index structure coefficient* $C_n^2(h)$, explained in [13]. For horizontal propagation it is assumed constant, but for vertical propagation it depends on altitude h and can be found

using the Hufnagel-Valley model, for example:

$$C_n^2(h) = 0.00594 \left(\frac{v}{27}\right)^2 (10^{-5}h)^{10} \exp\left(\frac{-h}{1000}\right) + 2.7 \cdot 10^{-16} \exp\left(\frac{-h}{1500}\right) + A \exp\left(\frac{-h}{100}\right), \quad (4-1)$$

where v is the root-mean-square wind speed in [m/s] and $A = C_n^2(0)$ is the nominal value of the coefficient at height 0. The coherence of light can be indicated by the coherence length or *Fried parameter* r_0 [13] and can be found by integrating $C_n^2(h)$ from the ground up to the altitude of observation L , as done in [17]:

$$r_0 = \left[0.423 \left(\frac{2\pi}{\lambda}\right)^2 \sec(\zeta) \int_0^L C_n^2(h) dh \right]^{-3/5}, \quad (4-2)$$

where ζ is the angular distance between guide star and science object to image. Typical values of r_0 for visible light range from less than 5 cm in strong daytime to 20 cm at good observation sites at night. Larger values for r_0 result in better seeing conditions [13, 17]. Interestingly enough, as stated by [17], the Fried parameter r_0 can be interpreted as the aperture diameter for which the mean-square wavefront error is approximately 1 rad^2 .

Alternative Solutions Describing atmospheric turbulence with Kolmogorov statistics results in a robust and dynamic three-dimensional representation, but is often too complex for direct implementation. A number of alternatives have been derived over the years, such as defining an optical transfer function (OTF) for each point in the field of view (FOV) using an anisoplanatic function [6]. This method makes thorough use of underlying statistics, but requires lengthy experiments and is parameter-unfriendly. Another method is the classical Adaptive Optics (AO) control approach, where identification experiments lead to a linearized state-space description of the complex atmospheric dynamics [17]. Such approaches can be very effective, but a complex and lengthy analysis of the problem remains. A rather straightforward and more tractable solution is to model layers of atmospheric turbulence as *phase screens* [33, 46, 52, 54]. Its description allows easy combination with the simulation framework.

Phase Screen Model Wu and Chui use multiple phase screens to simulate layers of distortion [54]. The dynamic three-dimensional volume of atmospheric turbulence is thus approximated by a number of dominant layers. The effect of such a layer on passing light can be represented by a two-dimensional transparent screen that imposes a phase aberration. By placing multiple phase screens in series, complex three-dimensional turbulence can be tractably simulated. In some cases dominant layers of turbulence can be identified such that atmospheric turbulence can be represented by only a few or even a single phase screen, as described by [52].

Such a single phase screen model is clearly illustrated by [46] and shown in Figure 4-1. In this figure the 'Pupil Plane' is the entry of the wavefront sensor (WFS), throughout this work coinciding with the aperture plane. Imaged stars are simulated as point sources in the 'Object Plane'. These sources emit incoherent light that passes the 'Phase Plane' and ends up in the

pupil plane, where it is imaged by the WFS. The phase aberration representing atmospheric turbulence is defined in the phase plane as a two-dimensional phase surface, indicated by the curved line. In this specific illustration two point sources - blue and green - are considered in the object plane, each corresponding to the optical path of the same colour. Each object illuminates a section of the phase plane and thus propagates this section to the pupil plane. If light is emitted incoherently the pupil plane will contain two distinct phase aberrations - one for each object - that will be imaged simultaneously on the sensor image.

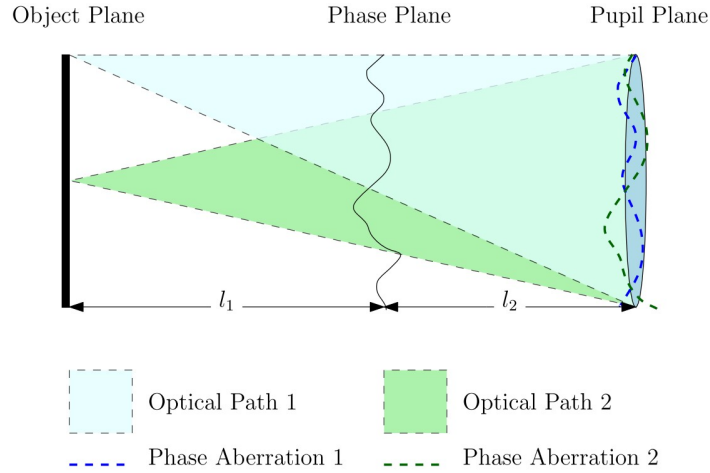


Figure 4-1: A schematic representation of the single phase screen model, showing how objects illuminate different sections of this screen and propagate different phases onto the 'Pupil Plane'. The propagation of light through free space is illustrated by projecting it onto the 'Pupil Plane'. Obtained from [46].

Kolmogorov Screens By combining the phase screen model with Kolmogorov theory one can generate random two-dimensional phase screens with Kolmogorov statistics. Common approaches to generating such *Kolmogorov screens* involve the turbulence structure function and Kolmogorov power spectral density (PSD) [17, 20, 31, 39]. The structure function for Kolmogorov phase fluctuations is reported by [31] and written as:

$$D_\phi(r) = 6.88 \left(\frac{r}{r_0} \right)^{5/3}, \quad (4-3)$$

where r is the distance between two discretely-sampled phase points and r_0 is the Fried parameter (4-2). Combining the structure function with a refractive-index PSD allows one to write the PSD for phase fluctuations, which is elaborated by Schmidt in [39]. For Kolmogorov turbulence this PSD can be written in terms of two-dimensional angular frequency $\kappa = 2\pi (f_x \vec{i} + f_y \vec{j})$ [rad/m] and is expressed as:

$$\Phi_\phi^K(\kappa) = 0.49 r_0^{-5/3} \kappa^{-11/3}. \quad (4-4)$$

The phase PSD and the phase covariance function are related through the Fourier transform [51], which means that C_ϕ can be obtained by an inverse Fourier transform of (4-4). In practice this is not tractable since (4-4) grows as frequency κ goes to zero. This means that

the PSD and thus the phase variance is infinite for low frequencies. This problem has been reported by [17, 20, 31]. A workaround reported in [20] involves defining the covariance of the phase over a finite region with the mean phase removed, as the low frequencies are predominant in this piston mode. An alternative solution is using more sophisticated turbulence models such as the von Kármán PSD to replace the Kolmogorov PSD [12, 17, 28]. This PSD is also provided in its original and modified form by Schmidt [39]:

$$\Phi_\phi(\kappa)^{VK} = \frac{0.49r_0^{-5/3}}{(\kappa^2 + \kappa_0^2)^{11/6}}, \quad (4-5)$$

$$\Phi_\phi(\kappa)^{MVK} = 0.49r_0^{-5/3} \frac{e^{(-\kappa^2/\kappa_m^2)}}{(\kappa^2 + \kappa_0^2)^{11/6}}, \quad (4-6)$$

where l_0 and L_0 are the inner and outer scales, the average sizes of the smallest and largest turbulent eddies, respectively. The variables $\kappa_m = 5.92/l_0$ and $\kappa_0 = 2\pi/L_0$ are chosen to match the high and low frequency behaviour predicted by dimensional analysis, as put by Schmidt. The phase PSD description allows one to find the turbulent phase covariance C_ϕ by inverse Fourier transform. Obtaining phase aberration ϕ from this covariance is described by [20], where the eigenvalue decomposition of the phase is substituted into the covariance:

$$C_\phi = E[\phi\phi^T] \propto (U\Lambda U^{-1})(U\Lambda U^{-1})^{-1} = U\Lambda U^{-1}U\Lambda U^{-1} = U\Lambda^2 U^{-1}. \quad (4-7)$$

Matrices Λ and U contain the eigenvalues and eigenvectors of ϕ . By generating a random vector \vec{x} , where each element x_i is independently generated with a variance proportional to Λ_i^2 , a random phase can be obtained:

$$\phi = U\vec{x}. \quad (4-8)$$

This provides one a mathematical model to generate random Kolmogorov phase screens.

4-2 Anisoplanatic Aberrations

The illustration of the single phase screen model in Figure 4-1 shows that two different phase aberrations are observed in the pupil plane. Due to their position in the object plane, both point sources illuminate a different section of the phase plane and thus propagate a different aberration to the pupil plane. From the perspective of the WFS this means that for different points in its FOV - the object plane - different aberrations are sensed. Such aberrations that vary throughout the FOV are referred to as *anisoplanatic aberrations* [6, 33, 46]. In the phase screen model of Figure 4-1 these aberrations are both dependent on the location of the phase screen and the separation between objects.

Phase Screen Location One contribution to anisoplanatism in the pupil plane is the distance between phase plane and object plane. In illustration Figure 4-1 one can determine the *common path* between two objects by the overlap of their optical paths. As the phase screen moves closer to the object plane the common path before the phase screen decreases. This means that the difference between illuminated phase screen sections and thus the difference of the aberrations in the pupil plane increases. By introducing distance ratio RL this

contribution to anisoplanatism can be quantified:

$$RL = \frac{l_1}{l_1 + l_2}, \quad (4-9)$$

where l_1 is the distance between object plane and phase plane and l_2 is the distance between phase plane and pupil plane, as illustrated in Figure 4-1. With this ratio three cases of anisoplanatism can be identified:

- 100 % anisoplanatism, $RL = 0$: The phase screen overlaps with the object plane and there is no common path between point sources and phase screen. Each point source uniquely propagates a section of the phase screen to the WFS.
- 100 % isoplanatism, $RL = 1$: The phase screen overlaps with the pupil plane and the common path between object plane and phase screen is equal for each point source. For each point source in the FOV the sensed aberrations will be the same.
- Anisoplanatism, $0 < RL < 1$: The phase screen is located somewhere between object plane and pupil plane and the degree of anisoplanatism depends on distance ratio RL .

The second case is a special situation, where the WFS detects *isoplanatic aberrations*, which are equal for each point in the FOV, the direct opposite of anisoplanatic aberrations. In reality atmospheric turbulence is often multiple kilometres away from a WFS and the dominant turbulent layers never overlap with the sensor pupil. Nevertheless one can approximate the aberrations as isoplanatic depending on their distance to bright objects.

Separation Between Objects - Isoplanatic Angle As mentioned in Section 4-2, both phase screen location and separation between point sources affect the degree of anisoplanatism. This second contribution is also stated by [6, 13]. Chapter 2 mentioned the important complication, supported by [17, 32], that light is scarce in astronomic imaging. This means that a WFS may not be able to accurately sense the aberrations from each imaged star. Common practice in AO systems is to use a bright guide star or laser beacon to provide sufficient light to accurately sense aberrations. As this distance between guide stars and science object grows, so does the difference between their respective aberrations. This means that the AO correction degrades for larger distances between guide star and science object [6]. To this end one can define the *isoplanatic angle* θ_0 that defines the region where one can assume aberrations to be isoplanatic. As formulated by [13, 17]: it can be used as a measure of spatial correlation of the atmospheric turbulence for different viewing angles and is often in the range of $5 - 10 \mu\text{rad}$ or several arcseconds. If the viewing angle between an object and a guide star is less than θ_0 , the mean-square error between the two reconstructed phases is less than 1rad^2 . It is emphasized in [17] however, that only 0.1 % – 1 % of stars in the sky fit the criterion of the isoplanatic angle, meaning that many science objects either lack a natural guide star or fall outside the isoplanatic angle.

An alternative solution is the use of artificial guide stars, such as laser beacons. These beacons operate by projecting a laser beam into the air to create a laser guide star close to the object to be imaged thus enforcing artificial isoplanatism. Unfortunately, these systems introduced other disadvantages, reported by [17], such as significant increase to system complexity and

cost, but also conical anisoplanatism. Laser beacons create guide stars up to 100 km altitude, depending on the type of laser used, which is significantly closer to the Earth's surface than natural guide stars. This results in the back-scattered light from the laser beacon to form a conical beam as it reaches the surface, sampling a different part of the atmosphere than light coming from a distant star.

Finally, it is also reported in [17] that these systems become unstable when the telescope diameter is too large, which led to the development of Multi-Conjugate Adaptive Optics (MCAO), where multiple AO systems and laser guide stars work together to achieve AO correction. As one can imagine, this increases the complexity and costs of the AO system even more. These downsides provided a drive to further develop anisoplanatic models and the incorporation of these models in AO setups [6, 33, 46]. To this end this thesis restricts itself to the use of one WFS to solve the anisoplanatic aberration problem.

Conclusion The theory highlighted in this chapter is part of the conducted literature survey, where the description of atmospheric turbulence and corresponding models have been investigated. This chapter concludes by having presented a straightforward model of simulating single or multiple layers of dominant turbulence. This phase screen model is used to generate input aberrations for the simulations.

The phase screen model enables a clear description of anisoplanatism and how anisoplanatic aberration are implemented. Additionally, the illustration in Figure 4-1 propagates the phase screen to the pupil plane by means of Projection. Two approximations of free space propagation, one of them being the Projection method, are discussed in Appendix A-1-3. For reasons presented in that discussion the Projection method will be used to propagate light through the phase screen onto the pupil plane.

The combination of the single phase screen model with Projection defines a so-called *forward* model for phase generation. This can be used to generate phase screens and corresponding input aberrations for the WFS simulations. By reversing the directions one can define a *backwards* model, that propagates the pupil plane phases back to the phase screen. This backwards model will be used to reconstruct the phase screen in presence of anisoplanatic aberrations.

The implementation of these phase generation methods will be elaborated in Chapter 5, that collects the development of the simulation toolbox, the author's contribution in this thesis.

Chapter 5

Simulation Toolbox

In order to produce wavefront sensor images and compare the plenoptic and Shack-Hartmann (SH) sensors, simulations in MATLAB are carried out. The development and results of a simulation toolbox present the majority of the author's contribution to modern wavefront sensing. The core of this toolbox is designed for isoplanatic aberrations, which are treated as aberrations generated in the wavefront sensor (WFS) aperture. The toolbox is extended to work with anisoplanatic aberrations by adding models for free space propagation and atmospheric turbulence. The toolbox itself is a collection of MATLAB-functions written by the author that are implemented by the main code or simulation front-end *CSS.m*.

The goal of this chapter is to detail the development of the simulation toolbox and present important results. At first, the implementation of the discrete Fourier transform (DFT) and subsequent constraints will be discussed in Section 5-1. Second, Section 5-2 details the most important toolbox functions that facilitate WFS simulation. Third, the actual algorithms to simulate the SH and plenoptic WFS will be presented in Section 5-3, supported by previously stated functions. Fourth, the developed algorithms are used to produce reconstruction results for simulated aberrations, presented in Section 5-4 up to Section 5-6, that lead to the conclusion of this chapter.

As mentioned in Section 3-1 the SH and plenoptic WFS are considered for further study. The plenoptic sensor can be configured as the plenoptic 3F (P3F) sensor by placing the microlens array (MLA) inside the objective lens back focal plane. The plenoptic 4F (P4F) sensor positions the MLA further behind the objective lens, such that objective lens back focal plane and MLA front focal plane overlap. This configuration significantly simplifies simulations, as will be explained further in Section 5-3. Additionally, each lenslet in the MLA of the P4F sensor forms a mini-Keplerian telescope with the objective lens, fixing the P4F sub-image locations beforehand. With this in mind, only the P4F sensor is considered of both configurations.

The presented algorithms make extensive use of the intensity patterns stored in the sensor images and sub-images. Throughout this chapter these terms will be used interchangeably, as the only function of the images is to provide the necessary intensity-values to the algorithm. Thus, if algorithms refer to sub-images or sensor images, they refer to the intensity patterns.

5-1 Discrete Modelling of Waves and Wave Propagation

Before elaborating the framework of the developed simulation toolbox, the treatment of discretely-sampled waves and their propagation through lenses is presented. The discrete-space analysis of wave propagation allows one to define the *simulation constraints* that ensure correct propagation of discretely-sampled waves through lenses. This section will discuss the equation for discrete wave propagation, the implementation of pupil functions and the derivation of multiple constraints for the simulation framework. Throughout this thesis the objective lens of the plenoptic sensor is circular, whereas the lenslets in the MLA are considered square.

Light can be represented as a bundle of light rays or as a complex wave. In order to adequately simulate the propagation of light through lenses diffraction must be accounted for. This means that wave optics should be the model of propagating light through optical systems. For more background information on representations of light, the reader is referred to Appendix A-1. At any distance z along its optical path, light is described by the two-dimensional field $U(x, y) = A(x, y)e^{j\phi(x, y)}$, where x and y are the spatial coordinates in the transverse plane and j is the imaginary unit. This field has amplitude $A(x, y)$ and phase $\phi(x, y)$.

One of the main goals of the simulation toolbox is obtaining a correct sensor image, not simulating the complex field as a three-dimensional volume. Additionally, wave propagation can be simplified to a Fourier transform operation if one propagates it from the front focal plane to the back focal plane of a lens, as elaborated in Appendix A-1-2. With this in mind, the propagation of light is represented by two-dimensional complex fields located at these focal planes. The propagation between these planes through an optical system is described by the following equation:

$$U_2(x, y) = \frac{1}{j\lambda f} \mathcal{F}\{P_l(\xi, \eta)U_1(\xi, \eta)\}, \quad (5-1)$$

where $U_1(\xi, \eta)$ and $U_2(x, y)$ are the complex fields at the front and back focal planes respectively, represented by the transverse spatial coordinate pairs (ξ, η) and (x, y) . The wavelength of light is represented in [m] by λ , the lens focal distance by f and the lens pupil function by $P_l(\xi, \eta)$. This pupil function will be properly defined and detailed below. The Fourier transform is implemented by operator $\mathcal{F}\{\}$.

For the purpose of simulations the complex fields are represented by discretely-sampled matrices, such that $U = A \odot e^{j\phi} \in \mathbb{C}^{N \times N}$, where \odot represents the element-wise Hadamard product. Here N is the number of matrix elements along one dimension, which will be referred to as the *grid size* of the matrix - assuming square matrices. In the same argument (5-1) is described by its discretized counterpart as well:

$$U_2 = \frac{1}{j\lambda f} \mathcal{F}_d\{P_l \odot U_1\}, \quad (5-2)$$

where $\mathcal{F}_d\{\}$ implements the DFT. One should be extra careful with the implementation of (5-2) since the Fourier transform expresses the output matrix in the frequency coordinates of the input matrix. To obtain matrix U_2 in its spatial domain a coordinate transform is necessary, which will be elaborated in the second boundary equation later in this section. For more details on the discretized effects of the DFT the reader is referred to Appendix A-2.

Pupil function The lens pupil function P_l shown in (5-1) and (5-2) enforces the physical dimensions of the lens onto the field. Its definition is given by (A-20) and repeated here:

$$P_l(x, y) = \begin{cases} 1 & \text{if } (x^2 + y^2) \leq r^2 \\ 0 & \text{otherwise} \end{cases} \quad (\text{Circular lens}),$$

$$P_l(x, y) = \begin{cases} 1 & \text{if } |x| \leq r \text{ AND } |y| \leq r \\ 0 & \text{otherwise} \end{cases} \quad (\text{Square lens}),$$
(5-3)

where r is the radius or half-width of the lens. In the case of discrete simulations the pupil function can be implemented as a two-dimensional *pupil mask*. This is a binary matrix with a corresponding x - and y -grid that imposes the physical dimensions. The mask is uniformly 0 except for x - and y -coordinates that fall inside the lens dimensions, where its value is 1 as defined in (5-3). An example of four masks is shown in Figure 5-1, where the masks are generated with two different centres and two different shapes, since both square and circular lenses are used throughout the simulations.

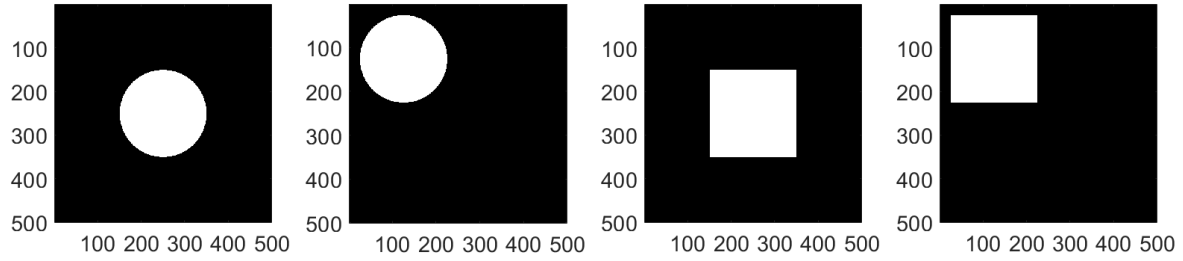


Figure 5-1: Examples of pupil masks generated by MATLAB. The size of the mask is set to 1, each lens has a radius of 0.2. The mask grid size is set to $N = 500$. From left to right: A circular lens on the optical axis with origin $(0, 0)$, a circular lens with origin $(-h/4, h/4)$, a square lens on the optical axis and finally, a square lens with origin $(-h/4, h/4)$.

Constraint - Phase difference The complex exponent in $U = A \odot e^{j\phi} \in \mathbb{C}^{N \times N}$ limits the maximum phase difference that allows unique representation of the phase. The exponent itself is periodic with 2π such that $e^{j(b+2\pi)} = e^{jb}$, a phenomenon often referred to as *phase wrapping*. For discretely-sampled phases this wrapping may or may not affect the propagation through a wavefront sensor. If the absolute phase difference $|\Delta\phi|$ between sampled points, referred to as the *phase jump*, does not exceed 2π the propagation is not affected.

An example of the phase wrapping problem is shown in Figure 5-2 for waffle aberration $\phi = -2\pi \cos(x - y)$. The spatial coordinates x and y are defined from $-7\pi/3$ to $+7\pi/3$. For each aberration ϕ a complex field $U = e^{-j\phi}$ is defined, where amplitude A is set to 1. The wrapped phase is retrieved by taking the natural logarithm of each field U . The left pair of images represents the waffle aberration represented on a 1000×1000 discretely-sampled grid, such that the phase jumps are smaller than 2π . Despite the higher phase values being wrapped back the phase gradients remain intact and one can still distinguish the waffle shape from the retrieved phase. The second pair of images represent a severely undersampled waffle aberration, represented on a 10×10 grid. Due to the low grid size this aberration consists of the piece-wise piston terms valued at -2π , 0 and 2π , such that both phase values and phase

jumps are wrapped back to 0. This is reflected by the retrieved phase aberration that is now uniformly 0.

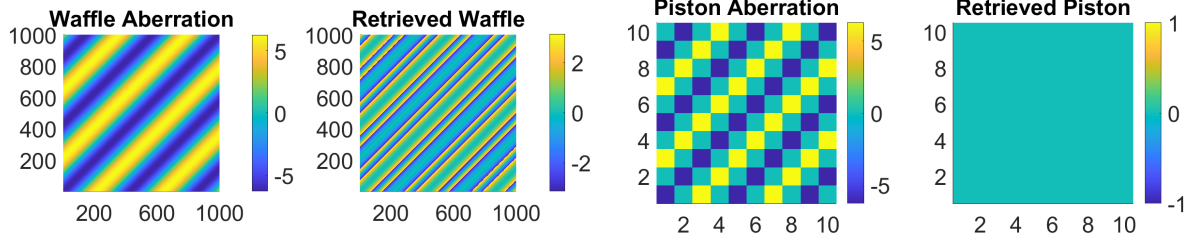


Figure 5-2: An example of the phase wrapping problem. The left pair of images show the input aberration ϕ and the retrieved phase, which is obtained by taking the natural logarithm of $e^{j\phi}$. In both examples phase wrapping affects the retrieved phase but in the latter, where the absolute phase differences are equal to 2π , also the overall phase gradient is affected. This significantly changes the shape of the retrieved phase.

To illustrate the effect on the imaging process both aberrations are propagated through a lens using the DFT. The pupil mask in both propagations are 0 in the pixels on the outer edges and 1 elsewhere. Furthermore, the images shown are not realistically recorded WFS images, but the absolute values of the Fourier transformed fields normalized to a maximum value of 1. A realistic approach of obtaining the digital sensor image is explained in Section 5-2-1. The images clearly show that undersampling can be detrimental in the phase description and propagation

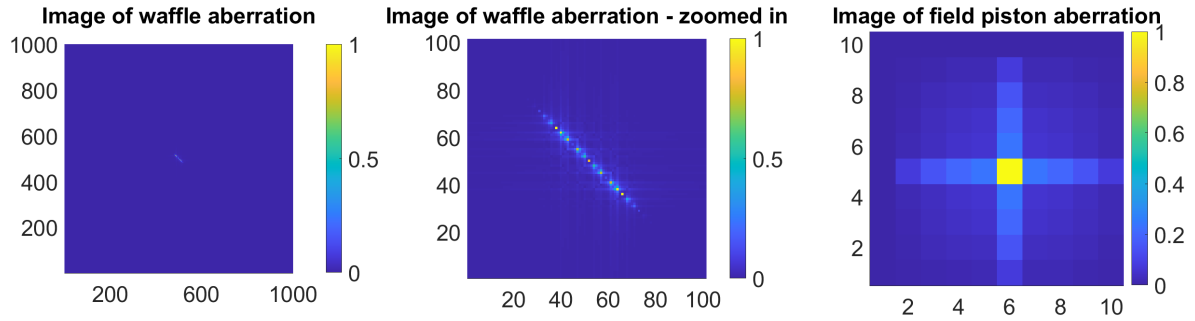


Figure 5-3: The images of the example aberrations in Figure 5-2. For each aberration a complex field is defined and propagated through a lens with the DFT. The a image represents the normalized absolute value of the Fourier transformed field.

The above example clearly shows that undersampling can be detrimental in the complex field description and propagation. It is important to keep the absolute phase differences below 2π , such that the phase gradients of the field remain intact. To this end the following constraint is defined for two-dimensional phase matrices $\phi(i, j) \in \mathbb{R}^{N \times N}$:

$$\begin{aligned} |\phi(i, j) - \phi(i + 1, j)| &< 2\pi, \\ |\phi(i, j) - \phi(i, j + 1)| &< 2\pi, \\ |\phi(i, j) - \phi(i + 1, j + 1)| &< 2\pi \end{aligned} \tag{5-4}$$

Constraint - Sizes Input and Output Planes As explained in the beginning of this section the DFT dominates the propagation of light through lenses. Unfortunately the DFT, implemented by the Fast Fourier Transform (FFT) algorithm in MATLAB, does not preserve the physical representations of the in- and output fields, as elaborated in Appendix A-2. If the light at the input plane with size h_1 is represented by field $U_1 = A_1 \odot e^{j\phi_1} \in \mathbb{C}^{N \times N}$, the physical size h_2 of the output plane, where output field $U_2 \in \mathbb{C}^{N \times N}$ is placed though (5-2) is described by:

$$h_2 = \frac{\lambda f N}{h_1}, \quad (5-5)$$

where wavelength λ , h_1 and h_2 are expressed in [m] and both U_1 and U_2 in transverse spatial coordinates. Parameter N is the grid size of both input and output planes.

Constraint - Minimum Grid Size Lenses The output of the DFT depends on the range of frequencies available and is thus susceptible to *aliasing*. High frequency content that falls outside the frequency range of the DFT is 'folded back' into this range, translating higher frequencies to lower ones and causing aliasing. For the propagation of light these frequencies are traced back to the gradients of phase aberration ϕ . The phenomenon of aliasing has been discussed in Appendix A-2 and the best solution is to simply increase the range of frequencies represented in the DFT, thus increasing grid size N . As is further elaborated in Appendix A-2, this is an ill-posed problem since the to-be-sensed aberration ϕ is unknown. One possible solution is to assume that the lens focussing is significantly stronger than the phase aberration. This assumption results in the phase directly behind the lens to be approximately equal to the known lens phase delay (A-19), neglecting all unknown phase terms. Then the Nyquist-Shannon sampling theorem can be applied for the phase differential to result in a minimum grid size:

$$N > \frac{4R^2}{\lambda f} + 1, \quad (5-6)$$

where R and f are the lens radius and focal distance expressed in [m].

Constraint - Fixed Pixel Size The size of the image sensor pixels can be taken into account before propagation, given this parameter is known. The elaboration of this approach in Appendix A-2 will be outlined here. As opposed to (5-6), where a minimum grid size was derived to ensure a certain frequency range, one could decrease the frequency step size, resulting in a finer grid in the image and ensuring a required pixel size. If one increases both physical size h_1 and grid size N linearly, keeping ratio N/h_1 constant, the resulting output plane retains its original size h_2 but is sampled much more densely. This process is referred to as *padding*, where zeroes are added along the edges of matrices, in this case complex fields U and pupil masks. For a fixed pixel size p one can find the required image plane grid size by:

$$M = \frac{\lambda f N}{2Rp}, \quad (5-7)$$

where N is the grid size inside the lens, fixed for example by minimum (5-6). This means that an $N \times N$ pupil mask can be defined for the physical dimensions of the lens and subsequently padded with $(M - N)/2$ zeroes on all sides. The same holds for complex fields $U = A \odot e^{j\phi}$ that are propagated through said lens.

The four constraints listed above establish the foundation of simulating light propagation through lenses and thus wavefront sensors. Through out this work they will be referred to as the *keystone equations*.

5-2 Supporting Toolbox Functions

The equations derived in Section 5-1 establish the foundation for simulating light propagation through lenses and wavefront sensors, but do not simulate the sensor image itself. As explained in the introduction of this chapter, the developed simulation toolbox is a collection of functions perform specific aspects of the WFS simulations. This section will elaborate on important features of the simulation toolbox, either by implemented concepts or specific functions. These elaborations enable a compact presentation of the WFS algorithms in Section 5-3, by a cascade of individual functions. The distinction between function and implemented concept lies in the title of the subsection. Specific functions are accompanied by names of corresponding m-files, implemented concepts are a part of functions or algorithms and thus have no dedicated m-file.

This section does not cover all toolbox features, as some are irrelevant to the chapter's narrative. In the same argument only the important elements of each function are highlighted, omitting some insignificant input and output arguments. If necessary the functions are presented alongside pseudocode. Furthermore, it should be noted that the functions are designed to work in the order as implemented by front-end code *CSS.m* and the algorithms for the SH and P4F sensor. As such, multiple input arguments reoccur throughout various functions. To improve readability the input arguments of several such functions are collected in Table 5-1, where the algorithms will refer to. Some of these listed parameters have been detailed in prior sections of chapters, others will be addressed by their respective functions.

Parameter description	SH sensor	P4F sensor	checkprelim	getpupil.m	planeprop.m	getPG.m	getscreen
Lens Parameters							
Focal distance MLA lenslets f_{obj}	X	X	X	-	X	X	-
Focal distance objective lens f_{obj}	-	X	X	-	X	X	-
Number of MLA lenslets M ($M \times M$ MLA)	X	X	X	-	X	X	-
Radius MLA lenslets r_{mla}	X	X	X	-	X	X	-
Radius objective lens r_{obj}	-	X	X	-	X	X	-
Sensor Size Parameters							
Aperture plane size P4F h_{ap}^{P4F}	-	X	X	X	X	-	X
Aperture plane size SH h_{ap}^{SH}	X	-	X	X	X	-	X
Grid size aperture plane P4F N^{P4F}	-	-	-	X	-	-	-
Grid size aperture plane SH N^{SH}	-	-	-	X	-	-	-
Image size P4F h_{im}^{P4F}	-	X	-	-	X	X	-

Table 5-1: List of parameters necessary to simulate SH sensor, P4F sensor and input arguments for important toolbox functions

Image size SH h_{im}^{SH}	X	-	-	-	X	X	-
Pixel size P4F d_{pix}^{P4F}	-	X	X	-	X	-	-
Pixel size SH d_{pix}^{SH}	X	-	X	-	X	-	-
Pupil plane size h_{pupil}	-	-	-	-	-	-	X
WFS type 'sensor_type'	-	-	-	-	X	X	X
Fourier Transform Parameters							
Propagation image size (for "sub-propagation") P4F h_{sub}^{P4F}	X	X	X	-	X	-	-
Propagation image size (for "sub-propagation") SH h_{sub}^{SH}	X	X	X	-	X	-	-
Light and Phase Screen Parameters							
Aberration type 'abtype'	X	X	-	X	-	-	-
Aperture plane phase aberration ϕ_{ap}	-	-	-	-	-	-	X
Aperture plane phase gradients $\partial\phi_{ap}/\partial u$	-	-	-	-	-	-	X
Fried parameter r_0	X	X	-	X	-	-	-
Input complex field U_{in}	-	-	-	-	X	-	-
Distance ratio phase screen RL	X	X	-	X	-	-	X
Spatial xyz locations point sources $O \in \mathbb{R}^{no \times 3}$	X	X	-	X	-	X	X
Sensor image P4F I^{P4F}	-	-	-	-	-	X	-
Sensor image SH I^{SH}	-	-	-	-	-	X	-
Wavelength λ	X	X	X	X	X	X	X
Zernike mode(s) z	X	X	-	X	-	-	-
Zernike mode strength(s) S_z	X	X	-	X	-	-	-
Reconstruction Parameters							
Slope model P4F sensor 'P4Fslopes'	-	X	-	-	-	-	-
Reconstruction type 'rec_method'	X	X	-	-	-	-	X
Retrieval type 'screen_method'	X	X	-	-	-	-	X
Other Parameters							
Paraxial approximation tolerance (%) 'tol'	X	X	X	-	-	-	-

Table 5-1: List of parameters necessary to simulate SH sensor, P4F sensor and input arguments for important toolbox functions (continued)

5-2-1 Digitizing Sensor Image

The intensity distribution of a complex field $U(x,y) = A(x,y)e^{j\phi(x,y)}$ is given by the absolute value squared, as shown in Appendix A-1-2:

$$I_c = |U(x,y)|^2 = |A(x,y)|^2, \quad (5-8)$$

where I_c represents the continuous intensity pattern, where the recorded intensity values are continuous variables. Image sensors, such as CCD or CMOS sensors, record this intensity pattern and produce digital integer pixel-values proportional to the *pixel bit depth*. This physical recording is simulated by taking the absolute value squared, normalizing it and multiplying it by the maximum pixel value. This recording is rounded to integer values with operator $\mathcal{R}\{\}$ such that the digitized image is found by:

$$I_d = \mathcal{R}\left\{2^b \frac{I_c}{\max(I_c)}\right\}, \quad (5-9)$$

where the pixel bit depth is given by b , such that an 8-bit pixel records $2^8 = 256$ pixel values. I_d is the digitized image, where the recorded values are discrete variables between 0 and 256. In this work sensor images and propagated complex fields are presented in this *digitized format*. Complex fields introduced in the sensor aperture plane are defined with a uniform amplitude $A(x, y) = 1$. It is thus more informative to present the aperture plane fields by their aberrated phases.

5-2-2 Field Shifting - *manipmat.m*

The implementation of the DFT in MATLAB shifts all zero-frequency components to the centre of the image, as explained in Appendix A-2. For lenses on the optical axis this does not pose a problem as zero-frequencies end up in the focal point. It becomes problematic for off-axis lenslets such as an MLA, where each lens images to its own sub-image in a specific region in the image plane.

The problem is shown for the SH sensor in Figure 5-4, where the MLA contains a 5×5 lenslet grid and the propagated fields are shown for lenslet 1 and 23. This selection is arbitrary since all 25 lenslets show the same behaviour: the image contains pixels close to the centre and fall within the centremost sub-image, indicated by the red outline. The same problem is shown for the P4F sensor in Figure 5-5. The generated aberration was a trefoil aberration (Zernike Z_3^{-3} in double-index ordering or Z_{10} in single-index Noll's ordering).

In Figure 5-4 each lenslet is indicated by its pupil mask, shown on the first and third image. For the P4F sensor each lenslet receives the light focussed by the objective lens, so the complex field behind objective lens is shown instead. The lenslets themselves are indicated by the red outlines.

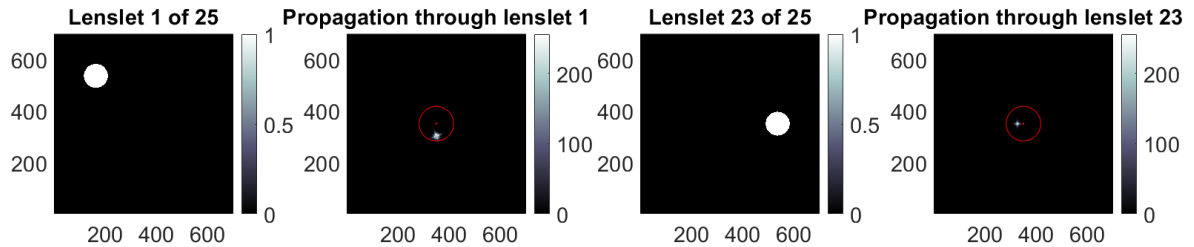


Figure 5-4: Propagation of a complex field through two different lenslets in a 5×5 MLA. From left to right: the pupil mask of lenslet 1, the corresponding lenslet image, the pupil mask of lenslet 23 and the digitized imaged field. The red circles in the propagated fields represent the image area corresponding to the centremost lenslet, lenslet 13. The simulated phase is a trefoil aberration, the grid size is set to $N = 700$.

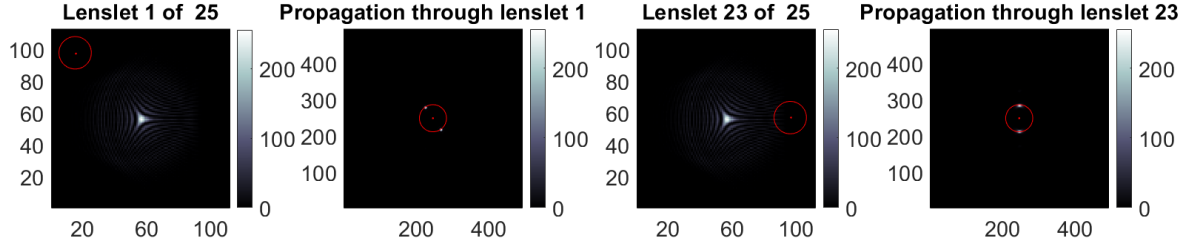


Figure 5-5: Propagation of a complex field through two different lenslets in the P4F sensor for a 5×5 MLA. From left to right: the objective lens image with the pupil function of lenslet 1 indicated in red, the corresponding image of lenslet 1 with the centremost sub-image indicated in red and the same for lenslet 23 and its image. The simulated aberration is a trefoil aberration.

Figure 5-4 and Figure 5-5 illustrate that the DFT aligns all propagated light to the centre of the image, as all light is imaged by the centremost lenslet. To obtain the correct sub-image each propagated field is shifted to the correct sub-image location. In the toolbox this is done for the propagation of each lenslet (i, j) , where the centre of propagated field $U_{out} \in \mathbb{C}^{N \times N}$ is aligned with the geometric centre of corresponding sub-image (i, j) . This shifting is illustrated for the SH sensor in Figure 5-6 and for the P4F sensor in Figure 5-7.

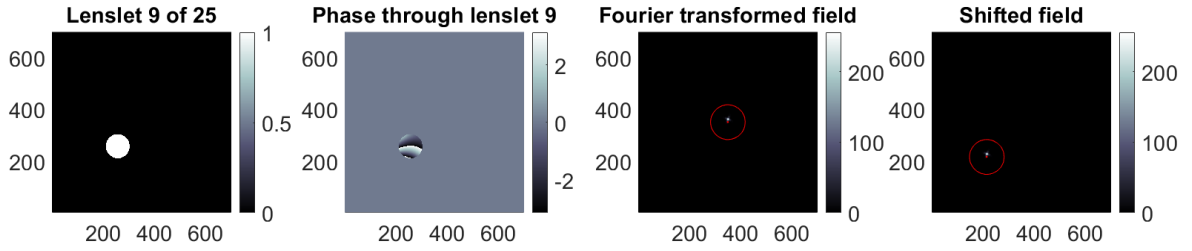


Figure 5-6: Propagation of a complex field through one lenslet in a 5×5 MLA, including the shift of the image plane field to its correct image area. From left to right: the pupil mask of the lenslet, the (wrapped) phase of the pupil-limited incident field and the images of both unshifted and shifted propagated field. The red circle in the unshifted image is to the centremost lenslet (lenslet 13) sub-image, while the red circle in the shifted field, the rightmost image, corresponds to the correct sub-image of lenslet 9. The simulated phase is a trefoil aberration, the grid size is set to $N = 700$.

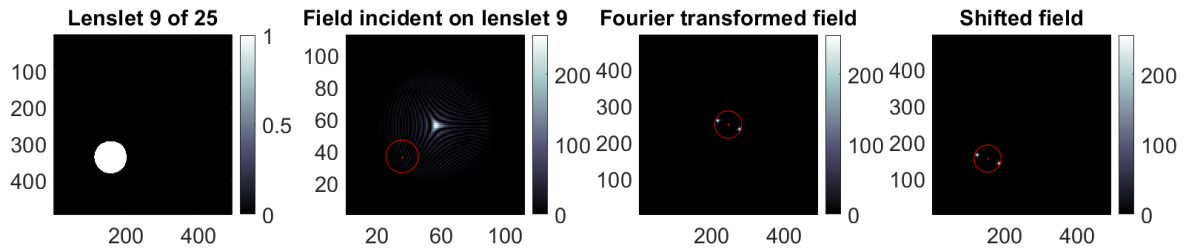


Figure 5-7: Propagation of a complex field through one lenslet in the P4F sensor for a 5×5 MLA. From left to right: the pupil function of lenslet 9, the image of the field incident on lenslet 9, which is indicated by the red outline, and the images of both unshifted and shifted propagated fields. In the unshifted image the red outline shows the centremost sub-image, whereas the outline indicated the correct sub-image of lenslet 9 in the shifted (rightmost) image. The simulated aberration is a trefoil aberration.

5-2-3 Preliminary Conditions - *checkprelim.m*

Before simulations start, one should check that all simulation parameters enable correct propagation and imaging. These checks include for example the plenoptic equality, minimum grid sizes, validity of the paraxial and minimum image sensor sizes. Each performed check will be individually detailed here.

Enforcing Plenoptic Equality The design of the P4F sensor is severely constrained by the *plenoptic equality*, as explained in Section 3-3-1. The equality for the P4F sensor is defined by substituting $L = f_{obj} + f_{mla}$ in (3-11):

$$\frac{d_{obj}}{f_{obj} + f_{mla}} = \frac{d_{mla}}{f_{mla}}, \quad (5-10)$$

where d_{obj} and f_{obj} are the size and focal distance of the objective lens, d_{mla} and f_{mla} the same for the MLA lenslets. If this equality is not upheld it will be enforced by changing the objective lens focal distance:

$$f_{obj} = \frac{d_{obj} f_{mla}}{d_{mla} - d_{obj}}.$$

Validity Paraxial Approximation Throughout this work many equations rely on the paraxial approximation. The validity of this approximation allows one to propagate light using Fresnel diffraction, or to write plenoptic equality (5-10) and angular spectrum (3-10) as linear equations. If the relative error between a trigonometric function and its argument is less than 1 % one can approximate the sine-, cosine- and tangent-functions by:

$$\sin(\theta) \approx \theta, \quad \cos(\theta) \approx 1, \quad \tan(\theta) \approx \theta. \quad (5-11)$$

In order to check the validity for all lenses one computes the ratio of lens radius r by its focal distance f and compares it to the tangent thereof. The paraxial approximation can be used in the simulations if:

$$\frac{|\theta - \tan(\theta)|}{\tan(\theta)} < 0.01, \quad (5-12)$$

where $\theta = r/f$. This criterion is checked for each lens in the optical system.

Determine Padding Factor In order to enforce the numerical aperture of the lens during propagation padding is required. As explained in Section 5-1 and Appendix A-2, the required padded grid size M is found by fixing the image plane pixel size p . By dividing M by the grid size inside the lens N , one can obtain the *padding factor*:

$$a_{pad} = \frac{\lambda f}{pd}, \quad (5-13)$$

where λ is the wavelength of light, d the size or pitch of the lens and f its focal distance. In the simulation toolbox this padding is applied before propagation on individual lenses without influencing how these lenses are spatially defined. This padding and propagation are properly addressed in Section 5-2-5.

Determine Aperture and Image Plane Sizes As explained in Section 5-1, the propagation through a WFS is performed by Fourier transforms between the focal planes of lenses. The physical dimensions of the input plane is imposed by the lens pupil mask.

In the SH sensor light is propagated through each lenslet of the MLA. The aperture plane is defined at the microlens front focal plane, whereas the image is formed in the array back focal plane. Its design, detailed in Section 3-2, lead to the conclusion that both aperture and image planes should be at least as large as the size spanned by the MLA lenslets $2Mr_{mla}$:

$$\begin{aligned} h_{ap}^{SH} &\geq 2Mr_{mla}, \\ h_{im}^{SH} &\geq 2.2Mr_{mla}, \end{aligned} \quad (5-14)$$

where the MLA consists of an array of $M \times M$ lenslets, each with radius or half-width r_{mla} . A 10 % tolerance is added in the image plane size to ensure sufficient space on the image for the algorithm. It is assumed that the MLA is packed without space between lenslets, such that its size is equal to $2Mr_{mla}$. If physical aperture and image sizes are provided, condition (5-14) will be checked for both. If either condition fails the provided parameter is updated with the minimum value.

The P4F design is significantly more complex and involves a cascade of two Fourier transform operations. The aperture plane is defined in the front focal plane of the objective lens, whereas the image forms in the back focal plane of the microlens array. There is an intermediary plane where objective lens back focal plane and array front focal plane overlap. Here the image of the objective lens is carried over to the MLA. From its design, detailed in Section 3-3 and illustrated in Figure 3-6, the minimum sizes of these planes can be determined. The aperture plane should be at least as large as the objective lens, whereas the image of the objective lens should be large enough to fill the MLA. Finally, the minimum size for the P4F image is found through geometry and given by (3-20). The following minimum sizes are listed as follows:

$$\begin{aligned} h_{ap}^{P4F} &\geq 2r_{obj}, \\ h_{mla}^{P4F} &\geq 2.2Mr_{mla}, \\ h_{im}^{P4F} &\geq 2.2r_{mla} \left(M + \frac{f_{mla}(M-1)}{f_{mla} + f_{obj}} \right), \end{aligned} \quad (5-15)$$

where r_{obj} and r_{mla} are the radii or half-widths of the objective lens and MLA lenslets, f_{obj} and f_{mla} their focal distances and $M \times M$ the array lenslet grid. Again, 10 % tolerances have been added to the objective lens propagated field and the P4F image, to ensure sufficient space for in the algorithms.

Minimum Grid Size SH Sensor With the lens dimensions and focal plane sizes determined one can express the minimum required grid sizes for the SH, to ensure a properly-sampled image. The SH image is formed by propagating the aperture plane field through the MLA for each individual lenslet, where separate pupil masks are defined for each lenslet. It is assumed that each lenslet images to the whole image, such that (5-5) can be written as:

$$h_{im}^{SH} = \frac{\lambda f_{mla} N}{2r_{mla}}.$$

If image size h_{im}^{SH} is fixed, either by minimum (5-14) or provided, the above equation can be rewritten to solve for grid size N . Additionally, the minimum grid size for each MLA lenslet is described by (5-6), by substituting $R = r_{mla}$ and $f = f_{mla}$. The minimum grid size for the aperture plane found by appropriate scaling of the larger of these two, expressed as:

$$N_{min}^{SH} = \frac{h_{ap}^{SH}}{2r_{mla}} \cdot \max \left[\left(\frac{4r_{mla}^2}{\lambda f_{mla}} + 1 \right), \left(\frac{h_{im}^{SH} \cdot 2r_{mla}}{\lambda f_{mla}} \right) \right]. \quad (5-16)$$

In the formation of the output argument, this grid size is rounded up to the nearest even integer, as is done for every grid size throughout the toolbox functions.

Minimum Grid Size Plenoptic Sensor For the P4F sensor a similar analysis can be performed, but due to the cascaded Fourier transform it is not feasible to retain one grid size throughout the entire sensor. The physical size of the field focussed by the objective lens is found by invoking (5-5):

$$h_{mla}^{P4F} = \frac{\lambda f_{obj} N}{h_{ap}^{P4F}}$$

and the physical size of the image, in the back focal plane of the MLA, is described by:

$$h_{im}^{P4F} = \frac{\lambda f_{mla} N}{h_{mla}^{P4F}} = \frac{\lambda f_{mla} N}{\lambda f_{obj} N} h_{ap}^{P4F} = \frac{f_{mla}}{f_{obj}} h_{ap}^{P4F} < h_{ap}^{P4F}.$$

From (5-15) it can be seen that the image should be larger than the MLA size, which means in most practical cases that it should also be larger than the aperture plane size. The size of the image can be increased by either resampling the field before propagation through the MLA, or implement a stricter *padding*. The latter will be elaborated here.

It is assumed that each MLA lenslet images to the whole image, such that the minimum grid size for the microlens array is given by:

$$N_{min,mla}^{P4F} = 1.1M \cdot \max \left[\left(\frac{4r_{mla}^2}{\lambda f_{mla}} + 1 \right), \left(\frac{h_{im}^{P4F} \cdot 2r_{mla}}{\lambda f_{mla}} \right) \right],$$

where the $1.1M$ prefactor ensures the grid size is defined for a physical size of $2.2Mr_{mla}$, the size of the MLA with 10 % tolerance. The aperture plane should be represented by the same grid size at minimum, to fit all sub-images on the P4F image. Additionally the physical size of the field propagated through the objective lens should fit on the MLA, resulting in a minimum grid size inside the objective lens:

$$N_{min,obj}^{P4F} = \max \left[\left(\frac{4r_{obj}^2}{\lambda f_{obj}} + 1 \right), \left(\frac{4.4Mr_{obj}r_{mla}}{\lambda f_{obj}} \right) \right].$$

This means that the grid size inside the objective lens should be equal to $N_{min,obj}^{P4F}$ at minimum. The aperture plane is found by padding the objective lens mask up to a grid size of $N_{min,mla}^{P4F}$. As such, the latter represents the minimum grid size to correctly simulate the P4F image:

$$N_{min}^{P4F} = N_{min,mla}^{P4F} = 1.1M \cdot \max \left[\left(\frac{4r_{mla}^2}{\lambda f_{mla}} + 1 \right), \left(\frac{h_{im}^{P4F} \cdot 2r_{mla}}{\lambda f_{mla}} \right) \right]. \quad (5-17)$$

The physical size of the aperture plane is equal to the physical size of the padded objective lens:

$$h_{ap}^{P4F} = \left(\frac{N_{min,mla}^{P4F}}{N_{min,obj}^{P4F}} \right) \cdot 2r_{obj}. \quad (5-18)$$

In a similar fashion to the minimum SH grid size, N_{min}^{P4F} is rounded up to the nearest integer in the formation of the output arguments of *checkprelim.m*.

Minimum Grid Size Sub-propagation The minimum grid sizes ensure correct light propagation but also establish a bottleneck in the toolbox. For larger MLAs - read $M > 11$ - the minimum grid size requirements for the MLA can effortlessly surpass 1000. The introduced bottleneck is an exponential growth of computation time, where the MLA image is simulated through M^2 Fourier transforms of $N \times N$ fields. Clearly, increasing M increases both the minimum grid size N in the Fourier transforms as the total number of transforms performed, resulting in an exponentially increasing simulation time.

In the expressions for the minimum grid sizes (5-16) and (5-17) the second term is often the larger of the two. This term can be reduced and thus the bottleneck circumvented, by introducing "sub-propagation". Instead of imaging to the whole image plane, each lenslet in the MLA propagates the input field to a smaller region on the image. By substituting the physical size of this region h_{sub} , specified by user input, in (5-5) one can significantly reduce the minimum grid size required for propagation.

If one simulates weak aberrations in the SH WFS, it is realistic to assume that each focal spot stays in its corresponding sub-image. This means that at minimum each lenslet only needs to image to a region of size $h_{sub}^{SH} = 2r_{mla}$. The P4F sensor geometry ensures that each lenslet images only to its corresponding sub-image, unaffected by aberration strength and such $h_{sub}^{P4F} = 2r_{mla}$ can be used as well.

The toolbox allows the user to specify the physical size of sub-propagation for each sensor, but the default values are set to $h_{sub}^{SH} = 10r_{mla}$ and $h_{sub}^{P4F} = 6r_{mla}$. This means that each lenslet in the SH MLA images to a 5×5 sub-image grid on the image, with the corresponding lenslet image in the centre of this grid. Since the P4F sub-images are not placed side-by-side each lenslet in the P4F MLA simply images to a region three times the size of one sub-image.

5-2-4 Generating Input Phase Aberrations - *getpupil.m*

The simulation framework allows two methods of generating phase aberrations. These will be represented by square $N \times N$ discretely-sampled phase screens. It was detailed in Section 4-2, that anisoplanatism is affected by the phase screen location and distance between individual point sources. In this thesis the isoplanatic angle is not considered and thus anisoplanatism is only dependent on the location of the phase screen, represented by distance ratio RL (4-9). If the phase screen is located inside the pupil plane, such that $RL = 1$, the toolbox performs *isoplanatic reconstruction*. For any value of RL between 0 and 1 *anisoplanatic reconstruction* is performed, the case of 100 % anisoplanatism with $RL = 0$ is not considered in this thesis.

Aberration Representations One method is the common representation by Zernike polynomials, which are ordered by Noll's index and adjusted with user-defined aberration strengths. The mathematical model and some examples of Zernike polynomials are detailed in Appendix A-4. One can generate determinate screens by selecting specific modes and strengths, but pseudo-random screen can be generated as well. This has been implemented by drawing aberration strengths from a pseudo-random normal distribution. By specifying the first k number of modes to consider, the mean strength and scale, one can modify the randomized distribution as desired.

Another representation is by Kolmogorov statistics, as explained in Section 4-1. By specifying the coherence length of light or Fried parameter r_0 , one can generate random two-dimensional phase screens that adhere to Kolmogorov theory.

Pupil Plane versus Aperture Plane Throughout this work both pupil and aperture planes have been used to indicate the same location - the entry of the WFS - but a distinction must be made between them. The entry of the WFS is referred to as the pupil plane when free space propagation is discussed, where it represents the endpoint of free space. When internal propagation through a WFS is considered its entry is referred to as the aperture plane and is the start of sensor propagation. Besides nomenclature there is also a difference in physical size. The aperture plane sizes of each sensor are determined either by the preliminary conditions Section 5-2-3 or given parameters. The pupil plane size is set equal for both SH and P4F sensor such that the generated phase screens are represented on the same physical plane but only differ in required grid size. The size of the pupil plane is set to the maximum of the aperture plane sizes:

$$h_{pupil} = \max(h_{ap}^{SH}, h_{ap}^{P4F}). \quad (5-19)$$

Phase Screen Model and Free Space Propagation After the phase screen is defined by representation of either Zernike modes or Kolmogorov statistics, it will be placed in three-dimensional free space, as illustrated by Figure 5-8. This illustration is a three-dimensional representation of the single phase screen model Figure 4-1. The distance ratio RL (4-9) determines where the phase screen is placed between object plane and pupil plane.

The illustration of the phase screen model in Figure 4-1 propagated light to the pupil plane by projection. This method, detailed further in Appendix A-1-3, used in this thesis, as it guarantees fixed grid size N in the pupil plane. This parameter is crucial in the simulations as it ensures correct propagation of light through the Fourier transform. It should be noted that the validity of this approximation decreases as the aberration strength or distance between phase screen and pupil plane increases.

To obtain the projected aberrations in the pupil plane one needs four parameters: object locations O , distance ratio RL , pupil plane physical size h_{pupil} its grid size N_{pupil} . Each object is defined as a point source that fully illuminates the pupil plane. By knowing the pupil plane size h_{pupil} and distance ratio RL one can find the size of the *phase screen cut-outs*, the sections of phase screen that each point source illuminates. These cut-outs are represented by the dashed coloured squares in Figure 5-8. Their sizes are found through geometry:

$$\frac{h_{pupil}}{L1 + L2} = \frac{h_c}{L1} \rightarrow h_c = RL \cdot h_{pupil}. \quad (5-20)$$

The Projection method enables one to find the cut-out centres, by tracing a line from each point through phase screen to the centre of the pupil plane. Where this line intersects with the phase screen the cut-out centre is found, shown in Figure 5-8 by the coloured **X**. Each point source is defined by spatial x - and y -coordinates, collected by matrix $O = \begin{bmatrix} O_x & O_y \end{bmatrix}$ for all sources. Cut-out centres d_c are found through geometry:

$$\begin{bmatrix} d_{c,x} & d_{c,y} \end{bmatrix} = (1 - RL) \begin{bmatrix} O_x & O_y \end{bmatrix}, \quad (5-21)$$

where $d_{c,x}$ and $d_{c,y}$ are the x - and y -coordinates of the cut-out centres respectively. The size of the phase screen can be determined in a number of ways. In this work the furthest point in all cut-outs is selected, based on its x - or y -coordinate with a 10 % tolerance:

$$h_{screen} = 1.1 \left(\max \left(\begin{bmatrix} |d_{c,x}| & |d_{c,y}| \end{bmatrix} \right) + \frac{h_c}{2} \right). \quad (5-22)$$

By separating d_c in its absolute x - and y -components the furthest distance in either x or y is found. This results in a square phase screen that encloses all phase cut-outs.

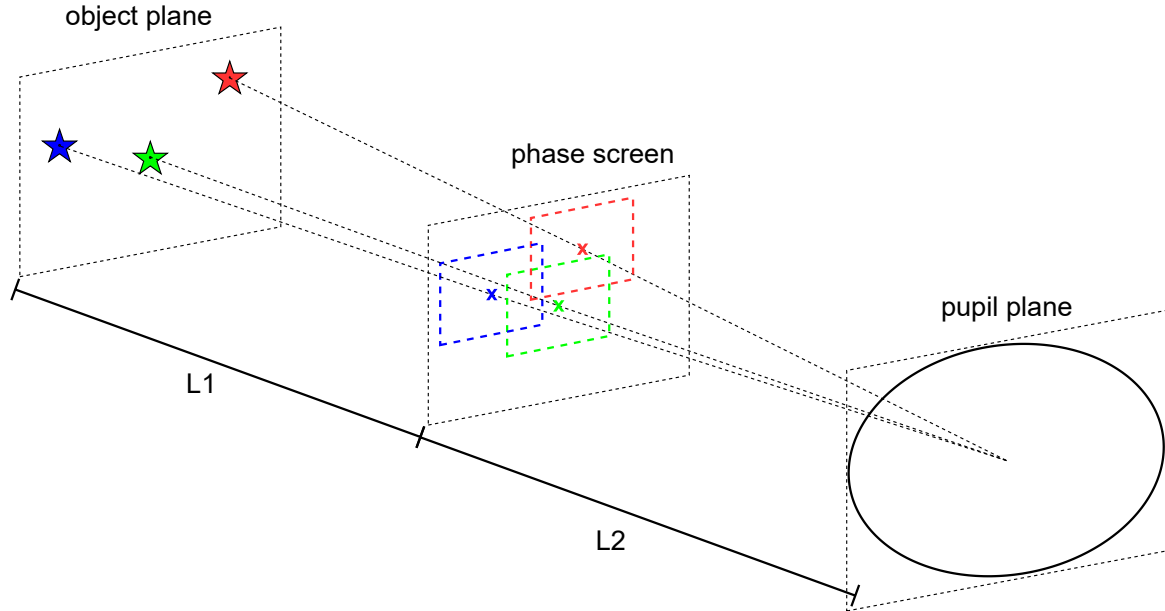


Figure 5-8: Three-dimensional visualisation of the phase screen model to represent atmospheric turbulence. One layer of turbulence is considered, which is represented by the Phase Screen at a distance $L2$ from the pupil plane. The phase screen cut-outs illuminated by each point source are found in the intersection of the Phase Screen with each objects FOV. These parts are shown by the coloured dashed squares lines in the Phase Screen.

Finally, the true benefit of the Projection method can be demonstrated. As explained in Appendix A-1-3, a fixed grid size N - one of the most crucial simulation parameters - is guaranteed. The preliminary conditions define the minimum grid size N in an aperture plane of size h_{ap} , such that the WFS image can be simulated. In the other direction, one can find the grid size N_{pupil} in the pupil plane:

$$N_{pupil} = \mathcal{R}^E \left\{ \frac{h_{pupil}}{h_{ap}} N \right\},$$

where operator $\mathcal{R}^E\{\}$ rounds its argument up to the nearest even number. The Projection method places phase screen cut-outs in the pupil plane, which means that each cut-out should be represented on a $N_{pupil} \times N_{pupil}$ grid. This allows one to find the phase screen grid size by linear scaling and rounding up to the nearest even integer:

$$N_{screen} = \mathcal{R}^E \left\{ \frac{h_{screen}}{h_c} N_{pupil} \right\}, \quad (5-23)$$

such that the phase screen can be generated as a square $N_{screen} \times N_{screen}$ matrix, via Zernike modes or Kolmogorov statistics.

Object-based Tilt In the Projection model described above the coordinates O of a simulated point source do not influence the imaging process. Take for example a point source at angular coordinates $\begin{bmatrix} \theta_x & \theta_y \end{bmatrix} = \frac{1}{O_z} \begin{bmatrix} O_x & O_y \end{bmatrix} = \begin{bmatrix} \frac{r_{mla}}{2f_{mla}} & \frac{r_{mla}}{2f_{mla}} \end{bmatrix}$ imaged by the SH sensor without atmospheric turbulence. In geometrical optics light rays emitted from this point source enter the aperture plane at angles θ_x and θ_y . This means that the focal spots are imaged with displacement $\begin{bmatrix} -\frac{r_{mla}}{2} & -\frac{r_{mla}}{2} \end{bmatrix}$ with respect to the sub-image centre.

With the Projections method the phase screen contains a flat phase which is projection onto the pupil plane. This means that each MLA lenslet images a complex field with a constant phase term, resulting in focal spots in the sub-image centres.

In order to obtain the correct spot locations as dictated by geometrical optics, object-based tilt (OBT) needs to be added. This is an additional phase term derived from a tilt aberration (Zernike modes Z_1^1 and Z_1^{-1}) defined on the unit square. The OBT is added in the pupil plane with size h_{pupil} , which means that an additional factor $h_{pupil}/2$ is added to ensure the right phase gradients are obtained. The OBT for simulated point source i is written as:

$$\phi_{OBT,i} = \frac{\pi h_{pupil}}{\lambda} (\theta_{x,i} \odot X + \theta_{y,i} \odot Y), \quad (5-24)$$

where X and Y are spatial coordinates on the unit square and (A-6) is invoked to obtain the phase aberration from the tilt wavefront.

For each point source i the pupil plane phase is obtained by adding the projected phase screen cut-out $\phi_{cut,i}$ to the OBT:

$$\phi_{p,i} = \phi_{cut,i} + \phi_{OBT,i}. \quad (5-25)$$

In the example above, the Projection method now results in a tilted phase in the aperture, such that the focal spots on the SH sensor image are found at displacement $\begin{bmatrix} -\frac{r_{mla}}{2} & -\frac{r_{mla}}{2} \end{bmatrix}$ with respect to the sub-image centre.

5-2-5 Focal Plane Propagation - *planeprop.m*

The discrete-space propagation through the focal planes of a lens (5-2) is implemented by toolbox function *planeprop.m*. The function is called for each propagation step and for each sensor, which means one call for the SH sensor and two for the P4F sensor. Beyond simply performing the DFT on an input field, it also checks that all input arguments enable correct propagation. An overview of *planeprop.m* is given as pseudocode in Algorithm 1. To improve

readability of the pseudocode, rounding up operator $\mathcal{R}^E\{\}$ is omitted when grid sizes are determined, but is implemented in *planeprop.m*.

First, the function checks that the current parameters satisfy the minimum size requirements (5-14) and (5-15). Next it generates the spatial locations of the lenslets and sub-images, referred to as *MlaLoc* and *CellLoc*. Finally it checks if padding or resampling is needed and adjusts the parameters where necessary. With all these preparations carried out the field can be propagated, which is done for each lenslet in the current step with an individual pupil mask. This means that the P4F objective lens is propagated through with 1 DFT operation, while the $M \times M$ MLA requires M^2 DFT operations - one for each lenslet. After propagation the dimensions of the output field are compared to the minimum sizes and either trimmed or padded if necessary.

In order to reduce nested *for*-loops the simulation toolbox associates each lenslet in the $M \times M$ microlens array with index $i = 1, \dots, \dots M^2$, instead of index-pair (i, j) used in earlier chapters. This double-index notation was adopted on purpose such that the working principles of the SH and P4F could be clearly presented and discussed. The switch from (i, j) to single-index notation i simplifies the toolbox functions, but does require one to store the spatial MLA information separately. This is done in matrices *MlaLoc* $\in \mathbb{R}^{M^2 \times 5}$ and *CellLoc* $\in \mathbb{R}^{M^2 \times 5}$, storing the information for the MLA and sub-images, respectively. The first three columns of *MlaLoc* contain the spatial locations of the M^2 lenslet centres, whereas the fourth and fifth columns store the index-displacements of the corresponding centre. For example, lenslet $i = 1$ in a 5×5 MLA of the SH sensor corresponds to $MlaLoc(i) = [-4r_{mla} \quad +4r_{mla} \quad 0 \quad -2 \quad +2]$, where the y -coordinate is defined positive in upward direction and the MLA is set at distance $z = 0$. The same structure is used for the sub-images and stored in matrix *CellLoc*. Each sub-image pixels remains associated with (s, t) .

Function *planeprop.m* implements padding around each lenslet propagated through in order to enforce the numerical aperture during propagation and ensure a specified image pixel size. This padding is only applied during propagation and does not influence the spatial locations of lenslets or their sub-images. As explained in Section 5-2-3 the user can choose to image to a sub-region on the image, decreasing the minimum grid size requirements and speeding up simulation times. If the user specifies this sub-region size as input parameter h_{sub} as "sub" DFT propagation is selected that implements this imaging. If the user request propagation to the entire image, using the "full" DFT propagation, one needs to specify $h_{sub} = 0$.

After propagation through each individual lenslet the output field is shifted to the correct spatial location on the image, dictated by the information stored in *CellLoc*. The shifted fields are added together to form the 'total' output field.

Algorithm 1 Focal Plane Fresnel Propagation**Input:** See Table 5-1**Output:** Output field U_{out} , its physical size h_{out} , sub-image outlines and locations $Cell$ and $CellLoc$, and pupil mask M_{tot}

```

1: function PLANEPROP.M
2:   Determine the input grid size  $N$  from input field  $U_{in} \in \mathbb{C}^{N \times N}$ 
3:   Determine minimum sizes of the input and output planes,  $h_{min,in}$  and  $h_{min,out}$ 
4:   Depending on 'sensor_type' set focal distance  $f$ , distance between objective lens and
   MLA  $L$  and lens radius  $r$ 
5:   Generate lenslet and sub-image spatial locations  $MlaLoc$  and  $CellLoc$ 
6:   Generate lenslet and sub-image outlines  $Mla$  and  $Cell$  for plotting purposes
7:   Determine physical quantities of the Fourier transform operation depending on input
   parameters:
8:   if  $h_{sub} = 0$  then
9:     % The Fourier transform propagates to the whole image %
10:    Set the minimum propagation size equal to specified image size  $D = h_{im}$ 
11:  else
12:    % The Fourier transform propagates to an image region % with size  $h_{sub}$ 
13:    Set the minimum propagation size equal to this size  $D = h_{sub}$ 
14:  end
15:  if  $d_{pix} = 0$  then
16:    Set Fourier propagation  $fprop = \text{"full"}$ 
17:    Determine size of image  $h_{out} = \frac{\lambda f N}{h_{in}}$ 
18:  else
19:    Set Fourier propagation  $fprop = \text{"sub"}$ 
20:    Determine the grid size in each individual lens  $N_{lens} = \frac{N \cdot h_{min,in}}{M \cdot h_{in}}$ 
21:    Determine the padding factor to ensure pixel size  $d_{pix}$  in the image:  $a_{pad} = \frac{\lambda f}{2rd_{pix}}$ 
22:  end
23:  Determine the physical size of the Fourier transform output  $h_{out} = \frac{\lambda f N}{h_{in}}$ 
24:  if  $h_{out} < D$  then
25:    % The size of the propagated field is not enough to fill the required size  $D$  %
26:    Determine required grid size  $N_{new} = \frac{h_{in} D}{\lambda f}$ 
27:    Resample input field  $U_{in}$  and determine new output size  $h_{out} = \frac{\lambda f N_{new}}{h_{in}}$ 
28:    Update the grid size  $N = N_{new}$ 
29:  end
30:  Determine the required grid sizes for each Fourier transform:
31:  if  $h_{sub} = 0$  then
32:    if  $fprop = \text{"full"}$  then
33:      % The Fourier transform takes the whole input plane and propagates to the
      whole image %
34:      Set the grid size of the image  $N_{image} = N$ 
35:    else if  $fprop = \text{"sub"}$  then
36:      % The Fourier transform takes individual lenses with padding and propagates
      to the whole image %
37:      Set the image grid size by adding lens padding:  $N_{image} = N_{lens} \cdot a_{pad}$ 

```

Algorithm 1 Focal Plane Fresnel Propagation (continued)

```

38:     end
39:   else
40:     if  $fprop = \text{"full"}$  then
41:       % The Fourier transform takes the whole input plane and propagates to a region
         in the image %
42:       Set the image grid size by linearly scaling  $N$ :  $N_{image} = \frac{h_{min,out}}{h_{out}} N$ 
43:       Set the image size to its minimum:  $h_{image} = h_{min,out}$ 
44:     else if  $fprop = \text{"sub"}$  then
45:       % The Fourier transform takes individual lenses with padding and propagates
         to a region in the image %
46:       Set the grid size of this region by adding lens padding:  $N_{im,sub} = N_{lens} \cdot a_{pad}$ 
47:       Determine the image grid size:  $N_{image} = \frac{h_{min,out}}{h_{out}} N_{im,sub}$ 
48:       Determine the physical size of the image:  $h_{image} = \frac{N_{image}}{N_{im,sub}} h_{out}$ 
49:     end
50:   end
51:   Define zero-matrices  $U_{out} \in \mathbb{C}^{N_{image} \times N_{image}}$  and  $M_{tot} \in \mathbb{R}^{N \times N}$ 
52:   Perform the lens propagation:
53:   for  $i = 1 : M^2$  do
54:     Generate the pupil mask for lens  $i$ 
55:     if  $fprop = \text{"full"}$  then
56:       Generate  $N \times N$  lens mask  $M_{lens,i}$  with the pupil function for lens  $i$ 
57:       Add it to the total pupil mask:  $M_{tot} = M_{tot} + M_{lens,i}$ 
58:       Pupil-limit the input field:  $U_{lim,i} = U_{in} \odot M_{lens,i}$ 
59:     else if  $fprop = \text{"sub"}$  then
60:       Using spatial locations  $MlaLoc$  and grid size  $N_{lens}$  extract the field incident on
         lenslet  $i$ ,  $U_{lens,i}$  from input field  $U_{in}$ 
61:       Set the values of the same pixels in  $M_{tot}$  to 1
62:       Pad it with zeros to obtain  $U_{lim,i} \in \mathbb{R}^{N_{image} \times N_{image}}$ 
63:     end
64:     Propagate the field with (5-2):  $U_{im,i} = \frac{1}{j\lambda f} \mathcal{F}_d\{U_{lim,i}\}$ 
65:     if ' $sensor\_type$ ' = 'plenoptic' then
66:       % To ensure each plenoptic sub-image stays within the bounds of the projected
         aperture, the propagated field is filtered with a Gaussian Mask %
67:       Generate a spatial coordinate grid spanning the physical dimensions of  $U_{im,i}$ 
68:       Define a circular aperture mask on these coordinates with diameter equal to
         the size of each plenoptic sub-image ( $2r_{mla}$ )
69:       Filter  $U_{im,i}$  with Toolbox function gaussmask.m
70:     end
71:     if  $h_{sub} = 0$  then
72:       Shift the propagated (and possibly filtered) field to its correct sub-image loca-
         tion with Toolbox function manipmat.m
73:       Generate an empty matrix  $U_{shift,i} \in \mathbb{C}^{N_{image} \times N_{image}}$ 
74:       Place the propagated (and possibly filtered) field in the correct pixels in  $U_{shift,i}$ 
75:     end Add shifted field  $U_{shift,i}$  to the output field  $U_{out} = U_{out} + U_{shift,i}$ 
76:   end

```

Algorithm 1 Focal Plane Fresnel Propagation (continued)

```

77:   Compare the physical image size  $h_{image}$  to the used specified parameter  $h_{im}$ 
78:   if  $h_{image} < h_{im}$  then
79:       Pad the output field with zeroes to obtain requested size  $h_{im}$ 
80:   else if  $h_{image} > h_{im} \& h_{im} > h_{min,out}$  then
81:       % The requested output size is larger than the minimum size so excessive pixels
       on the edges are trimmed away %
82:       Remove excess pixels from edges of  $U_{out}$  to obtain the requested size  $h_{im}$ 
83:   end
84:   Update the physical size of the output field
85: end

```

5-2-6 Retrieving Phase Gradients - *getPG.m*

The first goal of the toolbox is simulating the propagation of light through a WFS and obtaining the its image. Retrieving measurements from this image and performing phase reconstruction is the second goal. Toolbox function *getPG.m* reads the digitized sensor image and extracts the phase gradients depending on the selected WFS using approximation (A-6).

The specific method of retrieving the gradients depends on two input parameters; the sensor type - either SH or P4F - and the number of simulated point sources np . If only one point source is simulated the retrieval is straightforward and performed using the models discussed in Chapter 3. Unfortunately simulating only one point source is not realistic. It is nearly impossible to isolate light coming from a single star so the toolbox allows the simulation of multiple point sources. This does require additional image post-processing before phase gradient retrieval can be performed. This section will detail the retrieval process for both the SH and P4F sensor, differentiating between single and multiple point source imaging, abbreviated by (SPI and (MPI) respectively.

SH gradient retrieval - SPI If only one point-source is simulated for the retrieval of gradients from a SH sensor image, one can directly use the *image first moment* calculation (3-1) on each sub-image, as shown in Figure 5-9. The image is represented by matrix $I^{SH} \in \mathbb{R}^{N^{SH} \times N^{SH}}$, containing M^2 focal spots for each lenslet in the $M \times M$ microlens array. In Section 3-2-1 each lenslet and corresponding sub-image was indicated by index-pair (i, j) . As explained in Section 5-2-5, each lenslet is associated with index i and its corresponding information saved in $MlaLoc(i)$ and $CellLoc(i)$.

Each sub-image $I_i(s, t)$ is extracted from I^{SH} by converting the geometric locations of lenslet i - the values in the first two columns of $MlaLoc(i)$ - into a set of pixels on the image, as shown for lenslet 1, 38 and 97 in Figure 5-9. Each square sub-image consists of $P \times P$ pixels, where P is found by:

$$P = \mathcal{R}^E \left\{ \frac{2r_{mla}}{h_{im}^{SH}} N^{SH} \right\}$$

where h_{im}^{SH} and $2r_{mla}$ are the physical sizes of the SH image and sub-image, respectively. Operator $\mathcal{R}^E\{\}$ rounds its argument up to the nearest even number. The local coordinates

(\tilde{x}, \tilde{y}) are expressed in pixels, such that $\tilde{x}, \tilde{y} = -P/2, \dots, P/2$. The adjusted image first moment calculation is thus written as:

$$\Delta^p x(i) = \frac{\sum_{s=1}^P \sum_{t=1}^P \tilde{x}(s, t) \cdot I_i(s, t)}{\sum_{s=1}^P \sum_{t=1}^Q I_i(s, t)},$$

$$\Delta^p y(i) = \frac{\sum_{s=1}^P \sum_{t=1}^P \tilde{y}(s, t) \cdot I_i(s, t)}{\sum_{s=1}^P \sum_{t=1}^Q I_i(s, t)},$$
(5-26)

where centroids $(\Delta^p x, \Delta^p y)$ are the computed centroids expressed in pixels. The Hartmannogram in Figure 5-9 is obtained with an 11×11 MLA, where the physical size of each sub-image on the image is indicated by the red squares. The red dots show the geometric centre of each sub-image. The sub-images of lenslets 1, 38 and 97 are magnified to show the locations of both sub-image centre and computed centroid, marked as an enlarged dot and cross respectively.

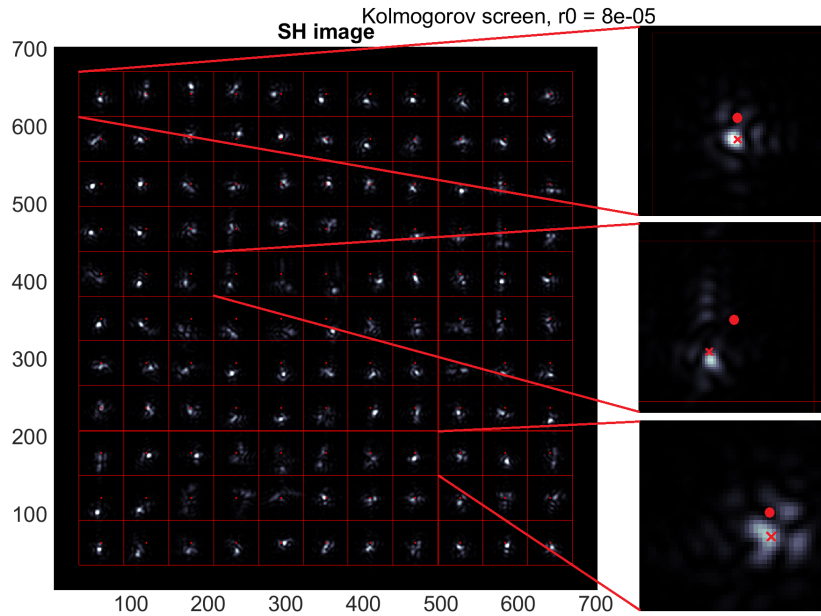


Figure 5-9: Demonstration of SPI gradient retrieval in the Hartmannogram. The simulated aberration is a random Kolmogorov screen with $r_0 = 8 \cdot 10^{-5}$ incident on an 11×11 MLA. Each sub-image is indicated by red squares, their geometric centres by red dots. The sub-images for lenslets 1, 38 and 97 are magnified to show to contents more clearly. In these sub-images both sub-image centre (dot) and centroid (X) are enlarged to be clear to the reader. The total image is digitized such that the maximum intensity is equal to 256.

The obtained pixel displacements are converted back to metric displacement by geometry:

$$(\Delta x, \Delta y) = \frac{h_{im}^{SH}}{N^{SH}} (-\Delta^p y, \Delta^p x)$$

In this conversion pixel displacements $\Delta^p x$ and $\Delta^p y$ are expressed in index-format, meaning that positive $\Delta^p x$ corresponds to negative Δy . As was explained in Section 3-2-1 one can obtain the slopes by division of the metric displacements by the lenslet focal length f_{mla} .

Approximation (A-6) allows one to find the phase gradients from these slopes:

$$\begin{aligned}\frac{\partial\phi(i)}{\partial x} &= \frac{-2\pi\Delta x}{\lambda f_{mla}}, \\ \frac{\partial\phi(i)}{\partial y} &= \frac{-2\pi\Delta y}{\lambda f_{mla}},\end{aligned}\tag{5-27}$$

where $\left(\frac{\partial\phi(i)}{\partial x}, \frac{\partial\phi(i)}{\partial y}\right)$ represents the average gradient in both directions of phase $\phi(x, y)$ incident on lenslet i . The incident phase is defined as the phase in the *aperture plane*. The minus sign is added to account for the direction reversion due to the lens focussing. A positive phase gradient in either direction results in a negative displacement or centroid in the image.

P4F gradient retrieval - SPI As explained in Section 3-3-1, each sub-image is assigned one slope with the P4F quantized angular spectrum (3-10):

$$(\alpha, \beta) = \frac{2r_{mla}}{f_{mla} + f_{obj}}(m, n),\tag{5-28}$$

where α and β are the slopes in x - and y -direction and pair (m, n) the index-displacements of the lenslet centre with respect to the optical axis. This means that for the centremost lenslet $(m, n) = (0, 0)$ such that it corresponds to slopes of value 0. For each sub-image i this information is stored in $CellLoc(i, 4:5)$.

Each pixel in the plenoptic image is assigned such a slope depending on the sub-image, which is the same in both the CAFADIS and Ray Tracing methods. The CAFADIS model does distance itself from the Ray Tracing model in averaging over all sub-images, instead of retaining all pixels as they are and building a geometrical light field. Function *getPG.m* is designed to retrieve the gradients and place them directly in the aperture plane. The Ray Tracing model assigns gradients and propagates the light back through the sensor, making it incompatible with the design of *getPG.m*. For this reason the CAFADIS model is used here and the Ray Tracing model is implemented by toolbox function *plenopticRT*.

The CAFADIS model assigns slopes to each pixel depending on the sub-image, using a modified image first moment calculation that averages over sub-images, instead of pixels as in (5-26). Using the index-displacements stored in $CellLoc$ and invoking (A-6) the phase gradients are found by:

$$\begin{aligned}\frac{\phi(s, t)}{\partial x} &= \frac{-4\pi r_{mla}}{\lambda f_{obj} + f_{mla}} \frac{\sum_{i=1}^{M^2} I_i(s, t) \cdot CellLoc(i, 4)}{\sum_{i=1}^{M^2} I_i(s, t)}, \\ \frac{\phi(s, t)}{\partial y} &= \frac{-4\pi r_{mla}}{\lambda f_{obj} + f_{mla}} \frac{\sum_{i=1}^{M^2} I_i(s, t) \cdot CellLoc(i, 5)}{\sum_{i=1}^{M^2} I_i(s, t)}.\end{aligned}\tag{5-29}$$

The minus sign is included with the same argumentation as in the SH retrieval, where positive phase gradients result in negative displacements.

SH gradient retrieval - MPI If multiple point sources are simulated, the gradients are extracted from the SH image with the same equations (5-26). Before the gradient can be

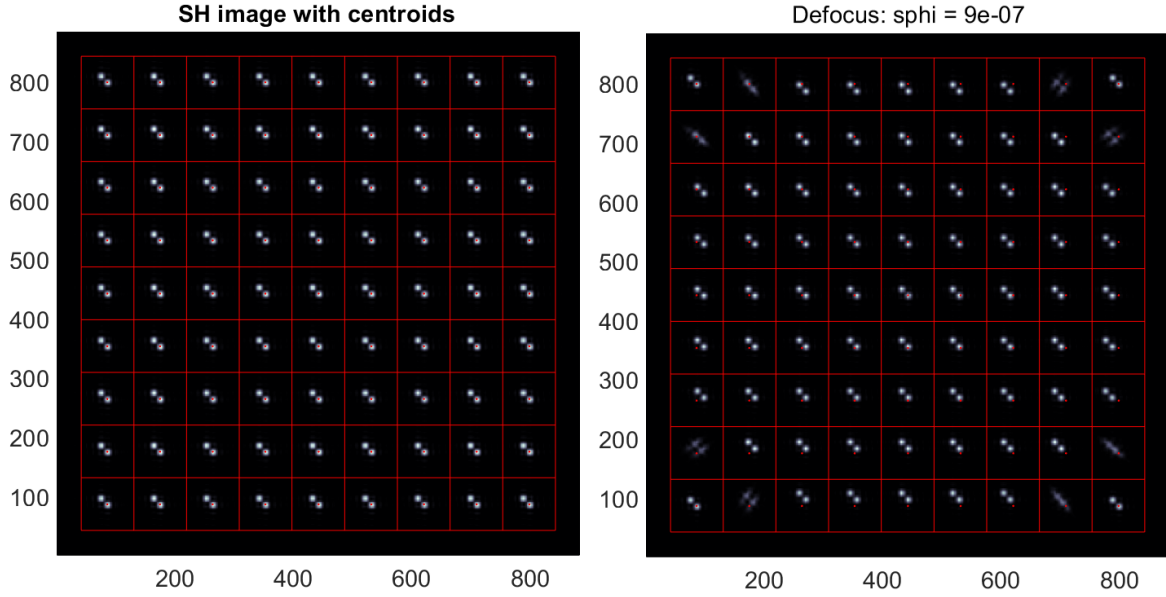


Figure 5-10: The SH sensor images for two objects located at $(x_1 = 0, y_1 = 0, z_1 = 1)$ [m] and $(x_2 = 0.0019, y_2 = -0.0019, z_2 = 1)$ [m]. The first scenario (left) is aberration-free imaging, whereas a defocus aberration is defined in the aperture plane of the second scenario (right).

retrieved however, image post-processing is necessary to separate the multiple focal spots. The problem is illustrated in Figure 5-10. On the left the aberration-free image of two point sources can be identified. The rightmost images shows the same, where a defocus aberration (Zernike mode Z_2^0 or Z_4) with strength $9 \cdot 10^{-7}$ [m] defined in the aperture plane.

The image first moment algorithm (5-26) computes one centroid over the whole sub-image without knowledge on which pixel belongs to which spot. In the case of Figure 5-10 this results in one *averaged centroid* for each sub-image, as opposed to one centroid per focal spot. As noted in Section 3-2-3 one can isolate the focal spots using *cluster detection* as long as they are sufficiently separated. This cluster detection is a common technique in computer vision and image processing, with a plethora of algorithms freely available. The algorithm used in this thesis is developed by own hand, such that time is not spent on finding an existing algorithm with desired behaviour and rewriting it in order to work with *getPG.m*.

The developed algorithm *getcluster.m* detects separate groups of clustered pixels and computes each cluster's centroid. The image is associated with a spatial grid defined on a unit square. This means that the output of *getcluster.m* must be scaled to the correct sub-image dimensions $2r_{mla}$ before phase gradients are obtained with (5-27). The algorithm itself is called for each sub-image $I_i(s, t)$ and requires four input parameters: the i -th SH sub-image, number of objects np , threshold θ and distance tolerance $dtol$. Threshold θ filters the image by setting pixel-values lower than θ to zero. Multiple thresholds can be specified by $\theta \in \mathbb{R}^{nt \times 1}$ such that *getcluster.m* performs detection on nt -filtered sub-images. From these nt -iterations the best match is selected and returned as output. Input $dtol$ controls the minimum distance between the centres-of-mass of detected clusters and merges clusters that are too close. Its default value is set to the *Rayleigh resolution criterion* (A-34).

The cluster detection of the aberration-free image with threshold $\theta \in \mathbb{R}^{5 \times 1}$ is shown in Figure 5-11. Based on the maximum intensity in the i -th sub-image, shown by 'SH image'

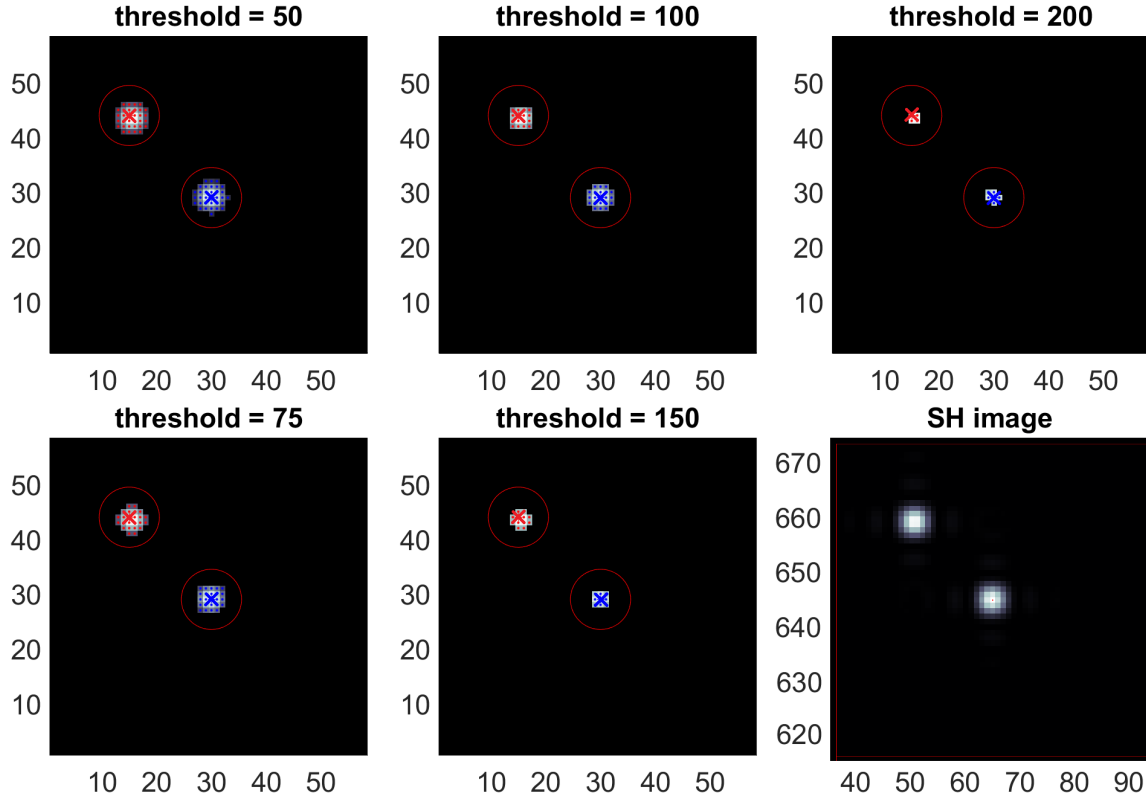


Figure 5-11: An example of cluster detection in the aberration-free Hartmannogram of Figure 5-11. The i -th sub-image is provided as input, here shown as 'SH image'. The maximum intensity in this image determines the number of filtered images to process, where each threshold value is shown in the image titles. The results of the algorithm are detected clusters and their centroids, distinguishable by their colour. The red circle outlines the expected size of the original focal spots.

in Figure 5-11, the number of thresholds θ are determined. This sub-image lead to 5 thresholds, whose values such that $I_i (I_i < \theta(i)) = 0$, are shown above each detection result. Each detected cluster is given its own colour, such that clustered pixels can be distinguished from each other. The coloured crosses, here enlarged for emphasis, indicate each cluster's centroid. The red circles indicate the focal spot's expected size, predicted by the Rayleigh resolution criterion. In Figure 5-11 each filtered image returns the same clustering, in which case *getcluster.m* selects the clusters with the highest pixel count. If different filtered images return different results, *getcluster.m* returns the best match, based on the number of clustered pixels, their pixel-values and the cluster separation. This selection procedure prioritises detections that result in the same number of clusters as simulated point sources. After the best match is found, the cluster's centroids are recomputed and returned by the algorithm.

The classification of focal spots, the knowledge which spot belongs to which imaged point source, is not performed by *getcluster.m*. This challenge can be solved many ways, but without explicit knowledge on the aberrations it is difficult to find an optimal solution. In this thesis the classification problem is tackled in the multiple point source imaging (MPI)-routine of *getPG.m*, where each focal spot is matched to the closest expected centroid. These centroids are determined from the aberration-free image of the same point sources, which means one only needs to source's angular coordinates to find them. Each detected cluster is

then matched to the point source whose aberration-free centroid is closest.

This matching means that the classification accuracy depends on the aberration strengths. Each expected centroid is found by the aberration-free image of the corresponding source, which means that they depend on the object-based tilt phase.. As explained in Section 5-2-4, this phase term is added to the pupil plane phase of each point source, depending on its angular coordinates. The OBT phase addition ensures that source's position is reflected in the aberration-free image. If this OBT is significantly stronger than the defined aberrations, which is the case in the rightmost image of Figure 5-10, the centroid of each focal spot is only slightly displaced with respect to the expected centroid. In this case it is very likely that each centroid is correctly classified to its imaged source.

If the aberrations are stronger than the object-based tilt, the aberrated centroids may be significantly displaced from their expected centroids. This means that too strong aberrations may cause the centroids and thus gradients to be assigned to a different point source than their own, resulting in an ill-posed phase reconstruction. This is an unfortunate side effect of the MPI phase gradient retrieval and leaves room for future improvement.

Algorithm 2 Image Cluster Detection

Input: Image I , number of objects np , threshold θ , distance tolerance d_{tol}

Output: Clustered pixels C , pixel colour-codings C_colour , cluster centroids C_com

```

1: function GETCLUSTER.M
2:   % For each given threshold the input image is filtered, all non-zero pixels are collected
   % and sorted based on proximity to existing clusters %
3:   Determine grid size  $N$  from image  $I \in \mathbb{R}^{N \times N}$  and define  $x, y$  on unit square
4:   Determine the number of thresholds supplied  $nt$ :  $\theta \in \mathbb{R}^{nt \times 1}$ 
5:   for  $i = 1 : nt$  do
6:     Set all pixel-values below threshold  $\theta(i)$  to zero:  $I_f = I$ ,  $I_f(I_f < \theta(i)) = 0$ 
7:     Collect all non-zero pixels and their indices in  $DATA \in \mathbb{R}^{K \times 3}$  with row-indices in
       the first column, column-indices in the second and pixel-values in the last.
8:     Sort DATA on the first column and save it as matrix  $P$ 
9:     Define cell-array  $C_{sub}$  to store clustered pixels and set iteration count  $wcount = 0$ 
10:    while  $P$  is non-empty do
11:      Check if the stopping criterion is reached (maximum iteration count)
12:      Determine the number of currently detected clusters  $nc$ 
13:      Select the first pixel from  $P$  as  $p\_current = P(i, :)$ 
14:      Determine the distance between  $p\_current$  and all pixels in each cluster (stored
        in  $C_{sub}$ ) in both  $x$ - and  $y$ -directions
15:      If current pixel  $p\_current$  is at most 1 pixel removed from any pixel in any
        cluster (directly neighbouring) assign a match to that cluster
16:      if  $\# \text{ matches} = 0$  then
17:        No neighbouring clusters found, make a new cluster for  $p\_current$ 
18:      else if  $\# \text{ matches} = 1$  then
19:        One neighbouring cluster found, assign  $p\_current$  to this cluster
20:      else if  $\# \text{ matches} > 1$  then

```

Algorithm 2 Image Cluster Detection (continued)

```

21:         Multiple matches found, assign  $p\_current$  to cluster with lowest pixel count.
           Assign randomly is multiple clusters are possible
22:     end
23:     Set cluster colour-coding of  $p\_current$  and save it in  $ColSub(i,:)$ 
24:     Remove  $p\_current$  from  $P$ , update cluster count  $nc$  and iteration count  $wcount$ 
25: end while
26: Find each cluster's centre-of-mass image first moments and coordinates  $x, y$ 
27: Merge clusters that have centres-of-mass less than  $dtol$  separated
28: Update the clustered pixels stored in  $Csub$  and their centres-of-mass
29: end
30: Collect all clustered pixels in cell-array  $C \in \mathbb{R}^{1 \times nt}$ . Do the same for the centres-of-mass
   and colour-codings
31: Give each collection  $C\{i\}$  a score by multiplying the total number of clustered pixels
   with the number of detected clusters. Normalize the score.
32: Keep collections  $C\{i\}$  with a score higher than 0.5 and more than one detected cluster.
33: Select best collection  $C\{i\}$  based on the detected pixels per individual cluster. Prior-
   itize collections with detected clusters equal to the simulated object count  $np$ 
34: Return the selected clusters, colour-codings and centres-of-mass
35: end

```

P4F gradient retrieval - MPI In the SH WFS each sub-image contains the focal spots of the incident wavefronts, sampled in the spatial domain. The P4F sensor samples wavefronts angularly, such that each sub-image represents the aperture plane filtered for a specific angle. Cluster detection described for the SH will not work here. This is also concluded from Figure 5-12, showing the aberration-free P4F sensor image (left) and the addition of a defocus in the aperture plane (right). In the SH sensor each sub-image contained separable focal spots, but the P4F spreads the pixels over the sub-images. In order to separate the pixels analogously to the SH MPI retrieval, one can reshuffle the pixels in the P4F image.

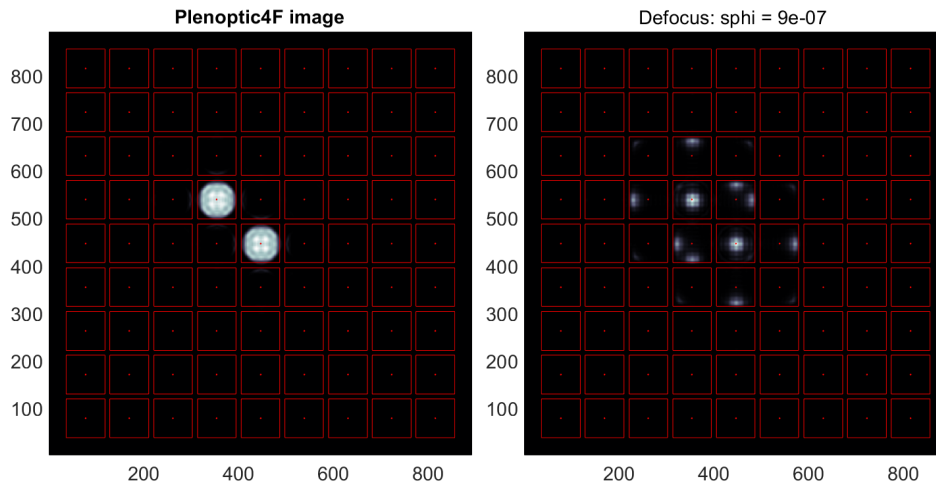


Figure 5-12: The P4F sensor images for aberration-free imaging (left) and the addition of a defocus aberration with strength $9 \cdot 10^{-7}$ [m] in the aperture plane (right).

This *plenoptic pixel reshuffling* treats the SH and plenoptic sensor as equivalent in the sense of geometrical optics, only differing in how the pixels are arranged. Take for example a square $M \times M$ MLA and assume both SH and P4F sub-images contain $P \times P$ pixels. For clarity the notation of Chapter 3 is used, associating each MLA lenslet and sub-image with index-pair (i, j) . In each sub-image the pixels are associated with index-pair (s, t) , such that the pixel intensity in each sensor image is given by $I^{SH}(i, j, s, t)$ and $I^{P4F}(i, j, s, t)$. In Chapter 3 it was explained that these indices encode the type of information stored in the pixels, where the SH WFS stores spatial information in indices (i, j) and angular information in (s, t) . In the P4F sensor these encodings are switched, which means that one could reshuffle the P4F image to obtain an SH-like image by reversing the type of information encoded in the pixels.

In this new Plenoptic 4F Shack-Hartmann hybrid (P4FSH) image each sub-image represents the spatial information (s, t) , angular information (i, j) is represented by the pixels inside the sub-image. Note that each P4F sub-image contains $P \times P$ pixels and as such the P4FSH image will show $P \times P$ sub-images. This means that as long as $M = P$ the SH and the P4FSH sensors will use the same MLA.

The plenoptic reshuffling is performed for each lenslet (i, j) , where pixel (s, t) is placed in pixel (i, j) of lenslet (s, t) in the new P4FSH microlens array. An example of this reshuffling is shown in Figure 5-13 in the aberration-free imaging of three point sources. The SH sensor image in this situation would contain three distinct focal spots in each of its sub-images, exactly what is produced by the P4FSH image. In this example it is straightforward to distinguish the imaging of each point source. The next example defines a defocus aberration in the aperture plane and images two point sources, shown in Figure 5-14. The pixels in the aberrated image are spread over the sub-images, complicating the classification of which pixel belongs to which source. The reshuffled image on the middle demonstrates the advantage of the reshuffling process, where a selection of sub-images is magnified to the right showing the reshuffled structure. The reshuffling algorithm, implemented by the MPI-routine of *getPG.m*, is shown in Algorithm 3.

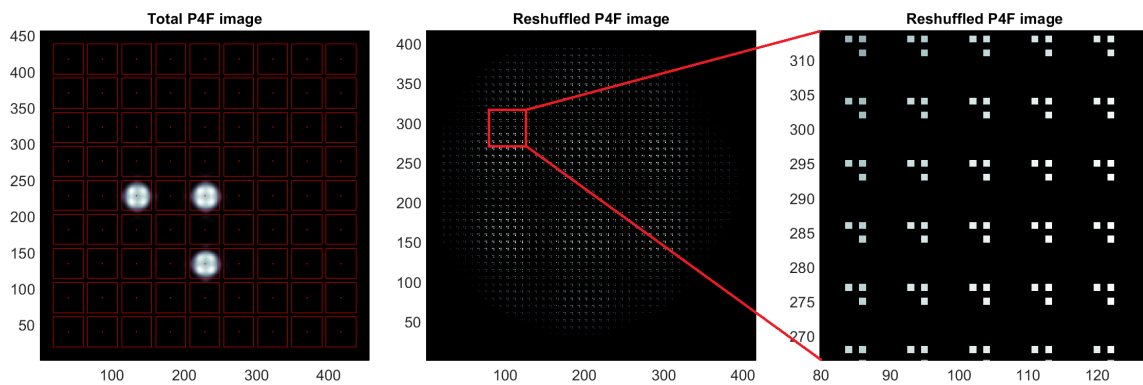


Figure 5-13: An example of reshuffling the plenoptic 4F sensor image to obtain the P4FSH image. Three objects sufficiently separated have been simulated without aberrations. From left to right: The original P4F sensor image, the reshuffled P4FSH image and a closer look at the P4FSH sub-images.

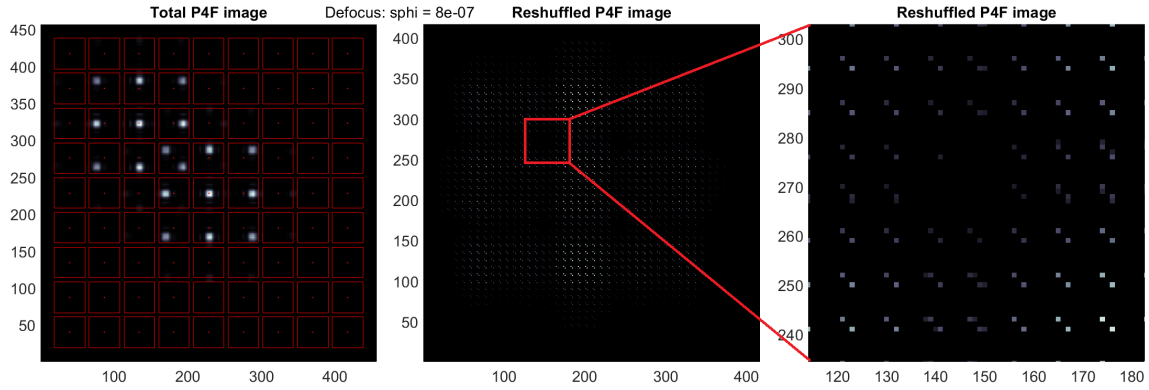


Figure 5-14: An example of reshuffling the plenoptic 4F sensor image to obtain the P4FSH image. Two objects sufficiently separated have been simulated with the addition of a defocus aberration of magnitude $8 \cdot 10^{-7}$ [m] in the aperture plane. From left to right: The original P4F sensor image, the reshuffled P4FSH image and a closer look at the P4FSH sub-images.

Algorithm 3 Plenoptic Pixel Reshuffling

Input: Plenoptic image I , image size h_{im} , lenslet count M and their radius r in the $M \times M$ MLA and pixel locations of the sub-image centre $CellLoc$

Output: Reshuffled image I_{rs} , pixel locations of shuffled sub-image centres $CellLoc_{rs}$

- 1: Obtain image grid size N from the plenoptic image $I \in \mathbb{R}^{N \times N}$
 - 2: Determine the number of pixels in each $P \times P$ sub-image: $P = \mathcal{R}^E\{\frac{2r}{h_{im}}N\}$
 - 3: Generate a $P \times P$ reshuffled MLA with each sub-image $M \times M$ pixels large
 - 4: **for** $i = 1 : M^2$ **do**
 - 5: Select i -th sub-image $I_{sub,i} \in \mathbb{R}^{P \times P}$ from image I using $CellLoc(i, 1:2)$
 - 6: Select index displacement of i -th lenslet from $CellLoc(i, 4:5)$
 - 7: Assign each pixel in $I_{sub,i}$ to the corresponding lenslet in the $P \times P$ reshuffled MLA
 - 8: Place each pixel in its reshuffled sub-image displaced $CellLoc(i, 4:5)$ pixels from sub-image centre
 - 9: **end**
-

The main difference between SH and P4FSH images, is that the reshuffled sub-images contains individual pixels instead of focal spots. The plenoptic pixels encode spatial information, whereas the SH sensor encodes it in entire sub-images. This is consistent with the inclusion of diffraction effects that, such that the plenoptic sensor performs angular sampling. One could interpret this sampling in an ordinary SH sensor with a $P \times P$ MLA by filtering the Hartmannogram, retaining only the most intense pixel per spot in the SH sub-images.

An obvious benefit of the P4FSH image is that no complicated cluster detection algorithm is required to find the centroids as these are already visible. What is necessary however is the classification of which pixel belongs to the image of which point source. This problem is solved in the same manner as described for the SH cluster detection, by matching the pixels to the expected aberration-free locations. Just as with cluster detection this method may incorrectly classify pixels when the phase aberrations are stronger than the object-based tilts.

Overview of Phase Gradient Retrieval With all routines in the phase gradient retrieval algorithm presented, pseudocode of *getPG.m* is provided to the algorithm's structure

Algorithm 4 Retrieve Phase Gradients from Sensor Image

Input: See Table 5-1

Output: Retrieved phase gradients PG

```

1: function GETPG.M
2:   % Select retrieval routine based on sensor_type and np %
3:   Determine number of simulated point sources  $np$  from  $O \in \mathbb{R}^{np \times 3}$  and image grid size
    $N$  from  $I \in \mathbb{R}^{N \times N}$ 
4:   if sensor_type = "SH" then
5:     Set the gradient grid size equal to  $M$ , gradients are expressed on  $M \times M$  grid
6:     if  $np = 1$  then
7:       Compute centroids using (5-26) and determine phase gradients  $PG \in \mathbb{R}^{M^2 \times 2}$ 
8:     else if  $np > 1$  then
9:       For each SH sub-image  $I_i^{SH}$  obtain clusters with getcluster.m
10:      Determine distances  $d$  between cluster centroids and point source expected
      centroids
11:      Assign clusters to source with smallest distance  $d$ 
12:      Recompute the centroids of classified clusters and compute phase gradients
       $PG_i \in \mathbb{R}^{M^2 \times 2}$  for each source
13:    end
14:  else if sensor_type = "P4F" then
15:    Determine the number of pixels  $P$  in each  $P \times P$  sub-image by linear scaling and
    rounding up to nearest even integer:  $P = \mathcal{R}^E \left\{ \frac{2r_{mla}}{h_{im}} N \right\}$ 
16:    if  $np = 1$  then
17:      Compute phase gradients  $PG \in \mathbb{R}^{P^2 \times 2}$  using CAFADIS model (5-29)
18:    else if  $np > 1$  then
19:      Reshuffle the P4F image and compute the expected pixel locations
20:      For each P4FSH sub-image  $I_i^{P4FSH}$ , match intensity pattern to expected loca-
      tions, assign sources to each pixel and save the classified pixel-displacements
21:      For each point source compute the phase gradients  $PG_i \in \mathbb{R}^{P^2 \times 2}$  with the
      classified pixel-displacements
22:    end
23:  end
24: end

```

5-2-7 Reconstruct Phase - *getphase.m*

As explained in Section 5-2-6 the obtained gradients from *getPG.m* result either from the image first moment model for the SH sensor, or from the CAFADIS model for the P4F sensor. As such the gradients are placed in the *aperture plane*. The resulting phase reconstruction is placed there as well. This thesis performs phase reconstruction using either the basic modal or zonal models, elaborated in Appendix A-5 and Appendix A-6. The reconstruction is

implemented by toolbox function *getphase.m*, whose pseudocode is presented in Algorithm 5.

Algorithm 5 Reconstructing Aperture Plane Phase

Input: Reconstruction method *rec_method*, reconstruction data *DAT* (2×1 cell-array), aperture plane mask padding factor *h_{pad}*, shape of aperture lens *aplens*

Output: Reconstructed phase ϕ

```

1: function GETPHASE.M
2:   if rec_method = "modal" then
3:     Get parameters  $D = DAT\{1\} \in \mathbb{R}^{3 \times 1}$ : aperture size  $h_{ap} = D(1)$ , wavelength  $\lambda = D(2)$  and output grid size  $N_{out} = D(3)$ . Get gradients  $PG = DAT\{2\} \in \mathbb{R}^{N^2 \times 2}$ 
4:     Limit output grid size with predefined maximum size:  $N_{out} = \min(N_{out}, maxN)$ 
5:     Assuming slopes are represented on square  $N \times N$  grid, retrieve grid size  $N$ 
6:     Determine the number of Zernike modes to consider:  $M = \min(200, \frac{1}{2}N^2)$ 
7:     Find grid size of  $\phi$  inside aperture:  $N_{def} = N_{out}/h_{pad}$ 
8:     Generate basis  $Z_{def}$  on grid size  $N_{def}$  and gradients  $G_X, G_Y$  on input grid size  $N$ 
9:     Stack gradients alternating between  $x$ - and  $y$ , resulting in matrix  $G_{in} \in \mathbb{R}^{2N^2 \times M}$ 
10:    Orthonormalize the basis and gradients, resulting in matrices  $Z_{\perp, def} \in \mathbb{R}^{N_{def}^2 \times M}$  and  $G_{\perp, in} \in \mathbb{R}^{2N^2 \times M}$  and conversion matrices  $\{\beta_Z, \beta_G\} \subset \mathbb{R}^{M \times M}$ 
11:    Check that  $Z_{\perp, def}$  and  $G_{\perp, in}$  are orthonormal:  $Z_{\perp, def}^T Z_{\perp, def} = I$  &  $G_{\perp, in}^T G_{\perp, in} = I$ 
12:    Scale gradients  $PG$  to wavefront slopes defined on unit square with aperture plane size:  $fs = \frac{\lambda h_{ap}}{2\pi(N-1)} PG$ 
13:    Stack the slopes alternating between  $x$ - and  $y$ -slopes, resulting in  $S \in \mathbb{R}^{2N^2 \times 1}$ 
14:    Estimate orthonormalized gradient coefficients  $\gamma$  with the MVUE:  $\tilde{\gamma} = (G_{\perp, in}^T G_{\perp, in})^{-1} G_{\perp, in}^T S$ 
15:    Convert  $\tilde{\gamma}$  to orthonormal Zernike coefficients:  $\tilde{\alpha} = \beta_Z \beta_G^{-1} \tilde{\gamma}$ 
16:    Reconstruct  $\phi$  from the orthonormalized Zernike basis  $\tilde{\phi} = \frac{2\pi}{\lambda} Z_{\perp, def} \tilde{\alpha} \in \mathbb{R}^{N_{def}^2 \times 1}$ 
17:    Reshape  $\tilde{\phi}$  into  $N_{def} \times N_{def}$  array and pad with zeros until  $N_{out} \times N_{out}$ 
18:  else if rec_method = "zonal" then
19:    Get cell size  $D = DAT\{1\}$  and phase gradients  $PG = DAT\{2\} \in \mathbb{R}^{N^2 \times 2}$ 
20:    Assuming slopes are represented on square  $N \times N$  grid, retrieve grid size  $N$ 
21:    Stack gradients in slope vector  $S \in \mathbb{R}^{2N^2 \times 1}$  alternating between  $x$ - and  $y$ -slopes
22:    Generate Southwell geometry matrices  $G_{grad} \in \mathbb{R}^{2N^2 \times 2N^2}$  and  $G_S \in \mathbb{R}^{2N^2 \times N^2}$  alternating between  $x$ - and  $y$ -directions
23:    Select all rows from  $G_{grad}$  that contain a single 1 and zeros elsewhere and remove these rows from both  $G_{grad}$  and  $G_S$ 
24:    Using the trimmed matrices determine averaged slopes  $S_a = \frac{1}{2} G_{grad} S \in \mathbb{R}^{2N^2 \times 1}$  and cell geometry  $G = \frac{1}{D} G_S \in \mathbb{R}^{2N^2 \times N^2}$ 
25:    Estimate  $\phi$  with the MVUE:  $\tilde{\phi} = (G^T G)^{-1} G^T S_a$  and reshape into  $\tilde{\phi} \in \mathbb{R}^{N \times N}$ 
26:  end
27:  Generate pupil mask in output dimensions with shape aplens and pupil-limit  $\tilde{\phi}$ . Return reconstructed phase  $\phi$ 
28: end

```

Modal Reconstruction - Orthonormality and Mode Limit The basic modal reconstruction algorithm, elaborated in `axrefax:BasicModal`, discusses the limited to the number of modes M that can be used for reconstruction. Using circular sub-images one should consider less modes than measurements available, resulting in the criterion $M < N^2$. Here it is assumed that the slope measurements are represented on a $N \times N$ discretely-sampled grid. In the simulations toolbox there is no guarantee that this criterion holds true, as the sub-images generated are square. Additionally, the representation and reconstruction of the phase rests on orthonormalized matrices.

In the following the orthonormality of the reconstruction matrices is checked for an increasing grid size. If orthonormality can be guaranteed, the criterion $M < N^2$ can be studied further. The reconstruction matrices are the Zernike basis functions $Z \in \mathbb{R}^{N^2 \times M}$ and gradients $G \in \mathbb{R}^{2N^2 \times M}$, where M Zernike modes and their slopes are represented on $N \times N$ discretely-sampled grids. The gradients are stacked, alternating between x - and y -gradients, consistent with the stacking in the zonal algorithm.

Matrices Z and G are orthonormalized using a modified Gram-Schmidt algorithm, that produces orthonormalized matrices $Z_{\perp} \in \mathbb{R}^{N^2 \times M}$ and $G_{\perp} \in \mathbb{R}^{2N^2 \times M}$ and conversion matrices $\{\beta_Z, \beta_G\} \subset \mathbb{R}^{M \times M}$, such that the original matrices can be retrieved by:

$$\begin{aligned} Z &= Z_{\perp} \beta_Z, \\ G &= G_{\perp} \beta_G. \end{aligned}$$

For any orthonormal matrix A holds $C = A^T A = I$, where I is the identity matrix. For matrix C one can state that each diagonal element should be equal to one, such that $c_{ij} = 1$, $i = j$. To track orthonormality of matrix $A \in \mathbb{R}^{N^2 \times N}$ for increasing grid sizes N the following parameters are defined:

$$\begin{aligned} c &= b \cdot \prod_{i=1}^N (A_i^T A_i), \\ b &= \begin{cases} 1, & \text{if } A^T A \text{ is diagonal} \\ 0, & \text{otherwise} \end{cases} \end{aligned} \quad (5-30)$$

where A_i is the i -th row of matrix A . This means that if $c = 1$ matrix A is orthonormal, if $c > 1$ it is orthogonal and if $c = 0$ matrix A is neither.

The orthonormality is checked for both matrices Z_{\perp} and G_{\perp} using different grid sizes $N = [5 \ 15 \ 25 \ 35 \ 45 \ 55]$. The selected sizes are values encountered throughout reconstruction processes in this thesis. The considered modes M are upper-bounded by $M < N^2$, such both Z and G contain the same Zernike modes. The considered modes range from 1 to N^2 , an exponentially growing range affecting computation times significantly. This is prevented by taking a maximum of 45 values for M from this range, as long as $N^2 > 45$, otherwise $M = 1, \dots, N^2$ can be used.

From the results shown in Figure 5-15 it is readily observed that condition $M < N^2$ is not met in any occasion, a result that does not come as a surprise. What is remarkable on the other hand is that the orthonormality is broken around either $M = \frac{1}{2}N^2$ or at $M = 200$, the dotted and dashed vertical lines respectively. Upon closer inspection of the figures it becomes

clear that orthonormality can be guaranteed as long as:

$$M < \min\left(\frac{1}{2}N^2, 200\right) \quad (5-31)$$

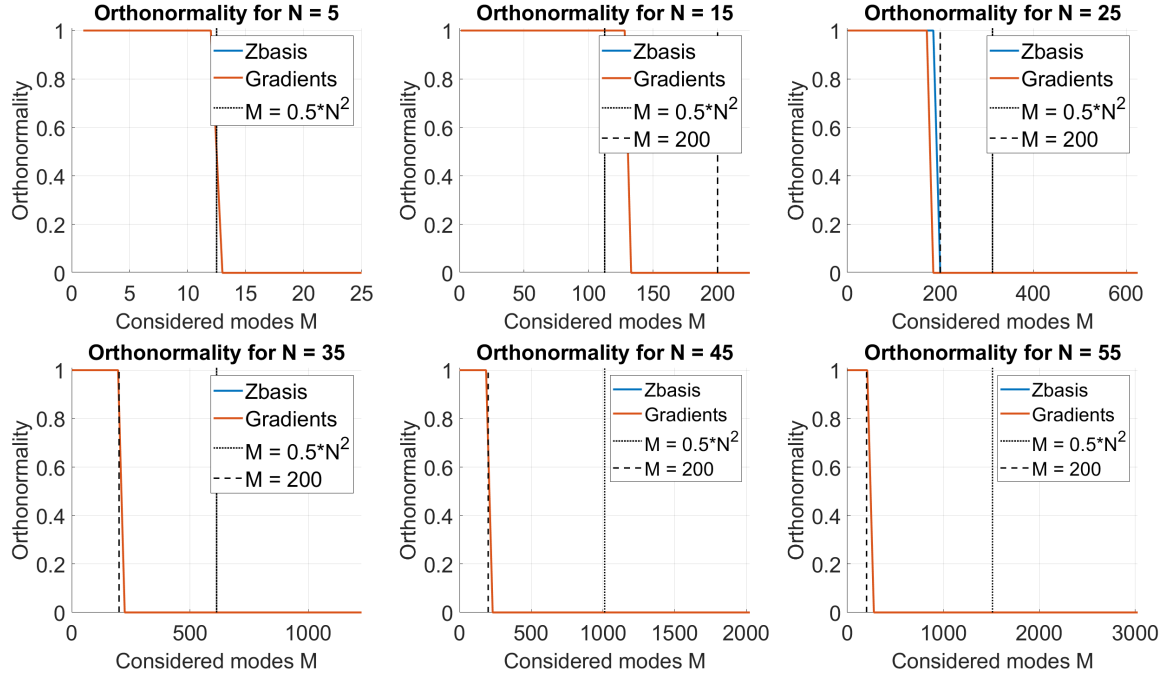


Figure 5-15: The orthonormality is checked for increasing grid size N and number of modes M . If the value on the vertical axis equals 1, the matrix is orthonormal. If the value is larger than 1 the matrix is orthogonal and the matrix is neither if the value is equal to 0. The condition is not checked for each $M = 1, \dots, N^2$ but rather 45 values for M are taken from the range of 1 and N^2 . This means that the orthonormality breaking point is not always aligned with either the $M = 0.5 * N^2$ (dotted) or $M = 200$ (dashed) vertical lines.

In the context of the simulation framework, $N = 11$ and $N = 50$ are reasonable and tractable grid sizes for the SH and P4F slope measurements respectively. These result in $M < 60$ for the SH and $M < 200$ for the P4F modal reconstruction.

Why precisely the orthonormality is broken beyond (5-31) is beyond the purpose of this thesis, as it may be a programming error in the used algorithms or an intrinsic property of orthonormalizing a large number of columns. The important conclusion is the knowledge for which values of M one can guarantee orthonormality. The upper-bound detailed in Appendix A-5 does not need to be accounted for. For reasonable sample densities orthonormality breaks long before the $M = N^2$ upper-limit is reached.

5-2-8 Retrieve Phase Screen - *getscreen.m*

In the case of anisoplanatic aberrations with $RL \in \langle 0, 1 \rangle$, reconstruction of the aperture plane phase will not be enough for phase correction. Function *getscreen.m* restores screen at distance $1 - RL$ away from the aperture plane, which can be done by two retrieval methods.

Both are elaborated here and included in pseudocode Algorithm 6. Both retrieval methods use the Projection model for approximating free space propagation. As such, *getscreen.m* is the inverse of the phase generation function *getpupil.m* elaborated in Section 5-2-4.

Knowing locations $O \in \mathbb{R}^{np \times 3}$ of the np simulated point sources and distance ratio RL , the intersections of each source's field of view (FOV) with the phase screen can be determined. With the WFS pupil plane grid size N , one can find the minimum grid size of the phase screen. Combining both concepts enables one to determine the exact pixels of the phase screen that each point source illuminates, the *phase screen cut-outs* $\phi_{cut} \in \mathbb{R}^{N \times N \times np}$.

Object-based Tip and Tilt Forward model *getpupil.m* determines each phase screen cut-out and adds object-based tilt, such that the point source location is accounted for in the image. This OBT is proportional to the angular coordinates of each i -th point source:

$$\phi_{OBT,i} = \frac{2\pi}{\lambda} \left(\frac{O_{x,i}}{O_{z,i}} X + \frac{O_{y,i}}{O_{z,i}} Y \right), \quad (5-32)$$

where $O = \begin{bmatrix} O_x & O_y & O_z \end{bmatrix} \in \mathbb{R}^{np \times 3}$ and $\{X, Y\} \subset \mathbb{R}^{N \times N}$ the x - and y -coordinates. These spatial coordinates are defined on the unit square since phase ϕ_{OBT} is derived from Zernike tip and tilt aberrations (Z_1^1 and Z_1^{-1} respectively). In *getpupil.m* the OBT is added in the pupil plane, which means that ϕ_{OBT} is represented on a plane with size h_{pupil} (5-19) and appropriate scaling is required. The pupil plane phase for point source i is expressed as:

$$\phi_{p,i} = \phi_{cut,i} + \frac{\pi h_{pupil}}{\lambda} \left(\frac{O_{x,i}}{O_{z,i}} X + \frac{O_{y,i}}{O_{z,i}} Y \right). \quad (5-33)$$

To enforce the physical dimensions of the pupil plane on the phase gradients, one needs to apply additional scaling:

$$\begin{aligned} \frac{\partial \phi_{OBT,i}}{\partial X} &= \frac{\pi h_{pupil}}{\lambda} \frac{2}{h_{pupil}} \frac{O_{y,i}}{O_{z,i}} = \frac{2\pi}{\lambda} \theta_{x,i}, \\ \frac{\partial \phi_{OBT,i}}{\partial Y} &= \frac{\pi h_{pupil}}{\lambda} \frac{2}{h_{pupil}} \frac{O_{x,i}}{O_{z,i}} = \frac{2\pi}{\lambda} \theta_{y,i}, \end{aligned} \quad (5-34)$$

where $\theta = \begin{bmatrix} \theta_x & \theta_y \end{bmatrix} = \begin{bmatrix} \frac{O_x}{O_z} & \frac{O_y}{O_z} \end{bmatrix}$ are the angular coordinates of the simulated point sources, with respect to the pupil and aperture plane.

The correct phase screen cut-outs are found by removing the object-based tilt ϕ_{OBT} from each aperture plane reconstruction:

$$\phi_{cut,i} = \phi_{ap,i} - \frac{\pi h_{pupil}}{\lambda} (\theta_{x,i} X + \theta_{y,i} Y), \quad (5-35)$$

where $\phi_{ap} \in \mathbb{R}^{M \times M \times np}$ are the reconstructed aperture plane phases for each point source, obtained from function *getphase.m*. This means that spatial coordinates X and Y should be represented by $M \times M$ matrices as well, as opposed to the $N \times N$ grid size in the forward model of *getpupil.m*. One finds the gradients of the phase screen cut-outs in the same manner by removing the OBT gradients:

$$\begin{aligned} \frac{\partial \phi_{cut,i}}{\partial X} &= \frac{\partial \phi_{ap,i}}{\partial X} - \frac{2\pi}{\lambda} \theta_{x,i}, \\ \frac{\partial \phi_{cut,i}}{\partial Y} &= \frac{\partial \phi_{ap,i}}{\partial Y} - \frac{2\pi}{\lambda} \theta_{y,i}. \end{aligned} \quad (5-36)$$

In this case gradients $\partial\phi_{ap,i}/\partial X$ and $\partial\phi_{ap,i}/\partial Y$ are the aperture plane phase gradients obtained from function *getPG.m*.

Phase Screen Retrieval 1 - Phase Back-propagation The most straightforward method of retrieving the phase screen is propagating the reconstructed aperture plane phases back to the phase screen location. With the pupil plane grid size fixed by the reconstruction grid size, one can determine the exact pixels in the phase screen that correspond to each cut-out. Back-propagation with projection is then reduced to simply placing each pupil plane phase in the corresponding pixels in the phase screen.

An unfortunate side effect of this method is illustrated in Figure 5-16. A tilt aberration is generated in the phase screen and illuminated by 9 equidistant point sources, resulting in a 3×3 grid of cut-outs. The phase screen is retrieved by placing the pupil plane reconstructions in their corresponding pixels, assuming the cut-outs is correctly reconstructed. Since the piston mode - which represents an averaged constant phase term - cannot be detected each phase reconstruction is *zero-mean*. As such, the retrieved phase screen as becomes *piece-wise zero-mean* as in Figure 5-16. The retrieved sawtooth-pattern is clearly an incorrect retrieval of the original phase screen.

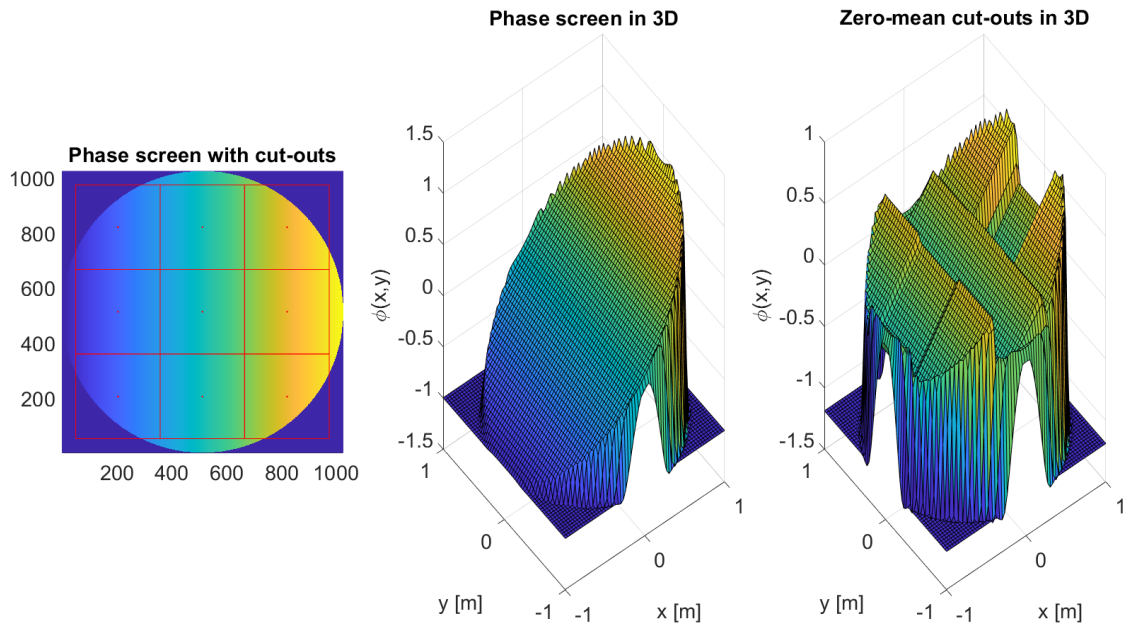


Figure 5-16: The hidden piston influence on phase screen reconstruction. From left to right: The phase screen and the corresponding cut-outs indicated by red squares, the phase screen profile and the profile where each cut-out has its mean removed. All values that fall outside the unit circle aperture are set to the minimum value of the original phase screen for clarity.

To present the problem as clear as possible the piece-wise zero-mean (PWZM) phase screen on the right of Figure 5-16 was retrieved by extracting each cut-out, removing the mean value and placing it back. No WFS sensing or reconstruction has been simulated as noisy reconstruction could obscure the illustration of the problem.

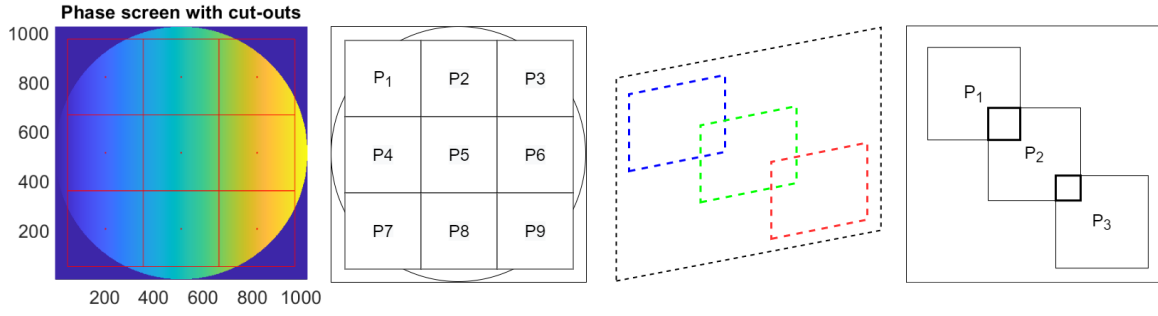


Figure 5-17: Two examples of parametrizing the piece-wise piston modes. The first two figures show the phase screen from Figure 5-16 and the corresponding square regions for each piston mode P_i . Outside of its corresponding region piston term P_i is zero. The last two figures show the same for a phase screen with three cut-outs, similar to the Phase Screen in Figure 5-8. The overlapping piston regions are indicated by the two thicker boxes. In the top-left box the total piston contribution is equal to $P_1 + P_2$, whereas it is equal to $P_2 + P_3$ for the bottom-right box.

One can describe the problem by *hidden piston modes* in each of the phase screen cut-outs that need to be reconstructed. The problem can be solved by introducing a piece-wise step function P for these modes, where each constant piston term P_i is present in the pixels of cut-out i . The piece-wise function is illustrated in Figure 5-17 for the phase screen in Figure 5-16 and in Figure 5-8. Then the actual phase screen can be represented by a total sum of each phase screen cut-out and the corresponding piston term. For the following derivation np point sources are simulated and each of their aperture plane reconstructions is represented by an $N \times N$ discretely-sampled grid. Through geometry one finds that the phase screen should be represented by a $N_{screen} \times N_{screen}$ grid. Each cut-out is defined over $N \times N$ regions in this screen and zero outside this region, such that $\phi_{cut} \in \mathbb{R}^{N_{screen} \times N_{screen} \times np}$. The phase screen retrieval is then expressed as:

$$\phi_{screen} = \sum_{i=1}^{np} \phi_{cut,i} + P_i \odot M_i = \phi_{screen,d} + P, \quad (5-37)$$

where $\phi_{screen,d}$ is the previously-determined PWZM phase screen obtained by simply adding all cut-outs together. Function P is now represented by piston mask $P \in \mathbb{R}^{N_{screen} \times N_{screen}}$, obtained by summing together individual piston terms P_i multiplied with binary masks M_i . These masks are 1 inside the i -th cut-out and 0 outside. In the case of overlapping pixels one should multiply ϕ_{screen} with an *averaging mask*, explained further below.

Since the actual phase screen is unknown one can solve the hidden piston problem by minimizing the variance of ϕ_{screen} in both x - and y -directions. Minimum variance results in a phase screen that is as smooth as reconstruction measurements allow. The phase screen variance is described by vectorizing the matrices in (5-37) twice, once row-wise in positive x -direction and once column-wise in negative y -direction. the element-wise variance is computed by taking the last $N_{screen}^2 - 1$ terms of each vectorized matrix and subtracting them from the first $N_{screen}^2 - 1$, similar to a single shift-register operation. This is performed by operator $\mathcal{V}\{\}$, resulting in the following variances:

$$\begin{aligned} V_x &= \mathcal{V}\{\phi_{screen,d}^{vx}\} + \mathcal{V}\{P^{vx}\}, \\ V_y &= \mathcal{V}\{\phi_{screen,d}^{vy}\} + \mathcal{V}\{P^{vy}\}, \end{aligned} \quad (5-38)$$

where superscripts vx and vy refer to vectorization in x - and y -direction. The obtained variances $\{V_x, V_y\} \subset \mathbb{R}^{(N_{screen}^2-1) \times 1}$ are stacked in one large vector $V \in \mathbb{R}^{(2N_{screen}^2-2) \times 1}$.

Before moving on, two considerations should be accounted for. First, there is no use in minimizing the variance over pixels with the same piston modes acting on them, which is identified by zeros in $\mathcal{V}\{P^{vx}\}$ and $\mathcal{V}\{P^{vy}\}$. If two pixels have the same piston term P_i acting on them, the variance of these pixels will be uninfluenced by P_i it does not contribute to a better estimation of P_i . Second, the minimization should result in an expression for piston modes P_i , which means that they should be extracted from the variance equations. With these considerations the stacked variances V can be written as a system of equations:

$$V = \begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \mathcal{V}\{\phi_{screen,d}^{vx}\} \\ \mathcal{V}\{\phi_{screen,d}^{vy}\} \end{bmatrix} + \begin{bmatrix} \mathcal{V}\{P^{vx}\} \\ \mathcal{V}\{P^{vy}\} \end{bmatrix} = b + Ap, \quad (5-39)$$

where the determined PWZM phase screen variances are collected by vector b , matrix $A \in \mathbb{R}^{(2N_{screen}^2-2) \times np}$ collects the coefficients of the individual piston modes, obtained by decomposition of the stacked piston variances and $p \in \mathbb{R}^{np \times 1}$ contains the parametrized piston modes. Finally, by treating the total variance V as an error the argumentation of the minimum-variance unbiased estimate (MVUE) can be applied to estimate the hidden piston modes:

$$\tilde{p} = (A^T A)^{-1} A^T b. \quad (5-40)$$

With the piston terms determined one can generate an averaging mask $M_{avg} \in \mathbb{R}^{N_{screen} \times N_{screen}}$ for the phase screen. Each pixel in this mask is the inverse of the number of independent cut-outs defined on that pixel. Take for example a 2×2 phase screen, where 3 cut-outs overlap in the top-left pixel, one cut-out is present in the bottom-right pixel and the remaining pixels are not illuminated. The averaging mask then becomes:

$$M_{avg} = \begin{bmatrix} \frac{1}{3} & 1 \\ 1 & 1 \end{bmatrix}, \quad (5-41)$$

where the default value for non-illuminated pixels is set to 1, such that pixels with either one cut-out or none are not averaged but retain their value. The retrieved phase screen is then expressed as a function of M_{avg} , phase screen cut-outs and reconstructed piston modes P_i :

$$\phi_{screen} = M_{avg} \odot \left(\sum_{i=1}^{np} \phi_{cut,i} + M_i \odot P_i \right). \quad (5-42)$$

Phase Screen Retrieval 2 - Local Reconstruction The first retrieval method described above relies on a double reconstruction, once in the aperture plane and once for the hidden piston terms. Such a reconstruction cascade is sensitive to measurement noise, as errors in the obtained gradients can lead to errors in the piston estimation. An alternative method circumvents this cascade by placing the sensed gradients in the phase screen instead. This method is based on the Ray Tracing model presented for the plenoptic WFS in Section 3-3-1, but switching the geometrical optics for the Projection method to describe free space propagation.

In a similar fashion to the method above one needs the grid size in the pupil plane and spatial parameters to define the phase screen cut-outs. Then, by placing each of the obtained phase

gradients in their cut-outs and averaging over the overlapping pixels one obtains the phase gradients in the phase screen. It should be noted that gradients outside the cut-outs are set to 0 by default, as they are unknown to the reconstruction algorithm and undetected by the WFS. The phase screen gradients are then reconstructed by function *getphase.m*, using either the modal or zonal algorithm available. These algorithms are elaborated in Appendix A-5 and Appendix A-6 for the interested reader.

Pseudocode for Retrieving Phase Screens The two methods discussed are implemented by function *getscreen.m*, which is presented by pseudocode here. In the pseudocode a distinction is made between the required grid size, depending on which retrieval method is selected. Both pupil plane grid sizes are indicated by N , but the number of phase gradients obtained from *getPG.m* may not be the same as the number of phase points reconstructed by *getphase.m*. Each grid size is rounded by to nearest even integer by operator $\mathcal{R}^E\{\}$. For readability this operator is emitted from Algorithm 6 but is implemented by the code.

Algorithm 6 Retrieving Phase Screen

Input: See Table 5-1

Output: Phase screen $\phi_{screen} \in \mathbb{R}^{N_{screen} \times N_{screen}}$

```

1: function GETSCREEN.M
2:   Determine spatial centres of phase screen cut-outs  $[d_{c,x} \ d_{c,y}] = (1 - RL) [O_x \ O_y]$ 
3:   Determine physical sizes of cut-outs  $\phi_{cut}$  and phase screen  $\phi_{screen}$ :  $h_c = RL \cdot h_{pupil}$ 
   and  $h_{screen} = 1.1 \left( \max \left( [|d_{c,x}| \ |d_{c,y}|] \right) + \frac{h_c}{2} \right)$ 
4:   % Based on parameter 'screen_type' select retrieval method for the phase screen %
5:   if screen_type = "phaseprop" then
6:     % Retrieve phase screen by back-propagating pupil plane phase and estimating
     hidden piston mode %
7:     Determine pupil plane grid size  $N$  from pupil plane phases  $\phi_p \in \mathbb{R}^{N \times N \times np}$ 
8:     Set the phase screen grid size by linear scaling:  $N_{screen} = \frac{h_{screen}}{h_c} N$ 
9:     Define  $\{\phi_{screen}, M_{avg}\} \subset \mathbb{R}^{N_{screen} \times N_{screen}}$  of zeroes
10:    for  $i = 1 : np$  do
11:      Determine pixel selection where cut-out  $\phi_{cut,i}$  is defined in phase screen and
      according binary piston mask  $M_i \in \mathbb{R}^{N_{screen} \times N_{screen}}$ 
12:      Place pupil plane reconstruction  $\phi_{p,i}$  in cut-out  $\phi_{cut,i} \in \mathbb{R}^{N_{screen} \times N_{screen}}$ 
13:      Add 1 to the pixel selection in  $M_{avg}$ :  $M_{avg}(M_i = 1) = M_{avg}(M_i = 1) + 1$ 
14:    end
15:    Overwrite all zeroes in  $M_{avg}$  with ones and invert it element-wise:
     $M_{avg}(M_{avg} = 0) = 1$  and  $M_{avg} = M_{avg}^{\circ-1}$ 
16:    Compute averaged PWZM phase screen  $M_{avg} \odot \phi_{screen,d}$  and piston mask  $P$  (5-37)
    and vectorize them in both  $x$ - and  $y$ -directions
17:    Stack variances of  $M_{avg} \odot \phi_{screen,d}$  to obtain  $b \in \mathbb{R}^{2N_{screen} \times 1}$  (5-39). Do the same
    for  $P$  and obtain variance coefficients  $A \in \mathbb{R}^{(2N_{screen}-2) \times np}$ 
18:    Reconstruct piston terms  $p = (A^T A)^{-1} A^T b \in \mathbb{R}^{np \times 1}$ 
19:    Retrieve phase screen  $\phi_{screen} = M_{avg} \odot (\sum_{i=1}^{np} \phi_{cut,i} + p_i \odot M_i)$ 

```

Algorithm 6 Retrieving Phase Screen (continued)

```

20:  else if screen_type = "gradprop" then
21:      % Place obtained gradients in phase screen and reconstruct the screen %
22:      Set the phase screen grid size by linear scaling:  $N_{screen} = \frac{h_{screen}}{h_c} N$ 
23:      Determine pupil plane grid size  $N$  from phase gradients  $PG \in \mathbb{R}^{N^2 \times 2 \times np}$ 
24:      Define zero-matrices  $PG_{screen} \in \mathbb{R}^{N_{screen}^2 \times 2}$  and  $M_{avg} \in \mathbb{R}^{N_{screen} \times N_{screen}}$ 
25:      for  $i = 1 : np$  do
26:          Determine pixel selection where  $\phi_{cut,i}$  is defined in phase screen and according
            binary piston mask  $M_i \in \mathbb{R}^{N_{screen}^2 \times 1}$ 
27:          Add  $i$ -th gradients  $PG(:, :, i) \in \mathbb{R}^{N^2 \times 2}$  to index-selection of  $PG_{screen}$  and add
            1 to these pixels in  $M_{avg}$ 
28:      end
29:      Average the phase screen gradients  $M_{avg} \odot PG_{screen}$  and reconstruct phase screen
            with getphase.m, modal or zonal reconstruction specified by rec_method
30:  end
31: end

```

5-2-9 Correct Aberration - *correctphase.m*

As explained in Chapter 2, extensive analysis of the deformable mirror (DM) is beyond the scope of this thesis and an idealized DM is used instead. The considered assumptions are:

- The DM has the same number of pistons as pixels in the reference phase screen
- Each DM piston has a completely independent response and has a continuously-valued unlimited range
- The DM has 0 % hysteresis

These assumptions allow one to reduce the DM simulation to resampling the reconstructed phase to the reference grid size and subtracting it from the reference phase. This is implemented by function *correctphase.m*, which is detailed in pseudocode below.

Algorithm 7 Correcting Reference Phase Aberrations

Input: Reference phase $\phi_{ref} \in \mathbb{R}^{N_{ref} \times N_{ref}}$, reconstructed phase $\phi \in \mathbb{R}^{N \times N}$, aperture mask $M \in \mathbb{R}^{N_{ref} \times N_{ref}}$

Output: Residual phase $\phi_{res} \in \mathbb{R}^{N_{ref} \times N_{ref}}$, root-mean-square error $RMSE$, phase profile on DM $\phi_{corr} \in \mathbb{R}^{N_{ref} \times N_{ref}}$

- 1: **function** CORRECTPHASE.M
- 2: Determine grid sizes N and N_{ref} from $\phi \in \mathbb{R}^{N \times N}$ and $\phi_{ref} \in \mathbb{R}^{N_{ref} \times N_{ref}}$ respectively
- 3: Aperture-limit the reference phase with mask M : $\phi_{ref} = \phi_{ref} \odot M$
- 4: Resample the reconstructed phase ϕ to the reference grid size (implemented by own function *manipmat.m* or MATLAB function *im_resize.m*) to obtain DM correction profile $\phi_{corr} \in \mathbb{R}^{N_{ref} \times N_{ref}}$
- 5: Aperture-limit the correction phase with mask M : $\phi_{corr} = \phi_{corr} \odot M$
- 6: Compute the residual phase $\phi_{res} = \phi_{ref} - \phi_{corr}$
- 7: Remove the piston term from residual phase $\phi_{res} = \phi_{res} - \text{mean}(\phi_{res})$
- 8: Extract all pixels of ϕ_{res} inside aperture mask M : $p_{res} = \phi_{res}(M = 1)$
- 9: Compute the residual RMSE from the extracted pixels: $RMSE = \frac{\sqrt{p_{res}^T p_{res}}}{n}$, where $p_{res} \in \mathbb{R}^{n \times 1}$ such that $RMSE \in \mathbb{R}$
- 10: Return output arguments ϕ_{res} , $RMSE$ and ϕ_{corr}
- 11: **end**

5-2-10 Plenoptic Ray Tracing - *plenopticRT.m*

This thesis is focussed on a fair comparison between the Shack-Hartmann and plenoptic 4F sensor. The CAFADIS model is based on image first moments and is very similar to the gradient retrieval of the SH sensor. As such, much effort was put into the development of this model, resulting in comparisons that highlight the benefits of the *plenoptic design*.

Compared to the development of the CAFADIS model, implemented by functions *getPG.m*, *getphase.m* and *getscreen.m*, less emphasis is placed on the Ray Tracing model described by Ko, Davis and Wu. This method is described with its own free space propagation and phase screen reconstruction, that differ from the framework used to develop the toolbox.

To support the fair comparison between the SH and P4F sensors, the CAFADIS model is used to reconstruct the P4F phases throughout this thesis. The Ray Tracing model is described in *plenopticRT.m* and will be briefly introduced here.

The Ray Tracing model approximates light by a bundle of geometrical vectors, the *light field*. The propagation of these vectors through lenses and free space is detailed in Appendix A-1-1, but is summarized by the following equations in 2D:

$$\begin{aligned}
 h_2 &= h_1 + (z_2 - z_1) \alpha_2, \\
 \alpha_2 &= \alpha_1 - \frac{\Delta h_2}{f} \quad \text{In lenses,} \\
 \alpha_2 &= \alpha_1 \quad \text{In free space.}
 \end{aligned} \tag{5-43}$$

At each propagation stage a light ray is represented by the coordinate along the optical axis z -axis z_i , the distance from the z -axis h_i and the forward facing angle α_i , which is positive

in direction away from the optical axis. This model is readily extended to three-dimensional propagation.

In *plenopticRT.m* each pixel in the recorded plenoptic image is converted into a pencil of light rays, where the pencil thickness is determined by the pixel intensity. The image plane slope of the pencil is determined by the sub-image and angular spectrum (5-28), whereas its location is fixed by the spatial pixel location. The collective of pencil rays form the *light field*, that is stepwise propagated back to the phase screen by inversely applying (5-43). This light field consists of N pencil rays, propagated through M stages and stored in light field matrix $LF \in \mathbb{R}^{N \times M \times 4}$, where spatial coordinates h are stored in $LF(:, :, 1)$ and $LF(:, :, 2)$ and angles α in $LF(:, :, 3)$ and $LF(:, :, 4)$.

An example of the geometrical back-propagation performed by *plenopticRT.m* is shown Figure 5-18. Here $RL = 0.8$ and an one point source is simulated on the optical axis. The phase screen is generated using an astigmatism (Zernike mode Z_2^2) with magnitude $5 \cdot 10^{-5}$ [m]. The number of slopes forced to be equal to the maximum number of lenslets, represented on a 7×7 discretely-sampled grid.

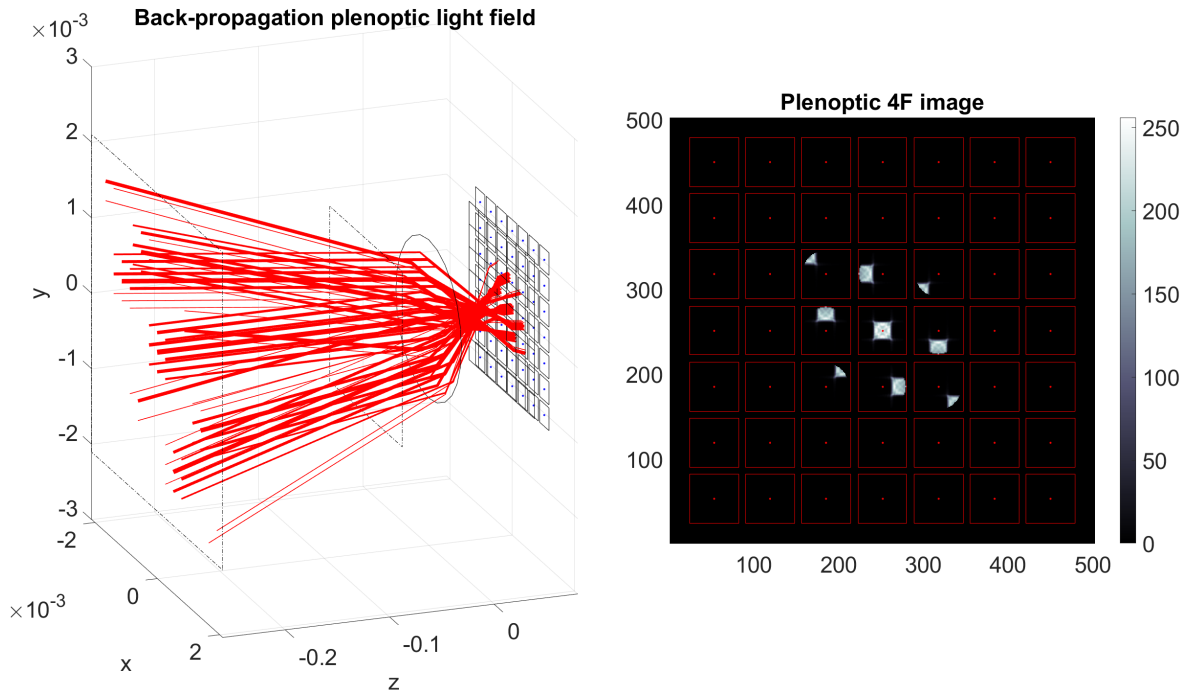


Figure 5-18: Illustration of the Ray Tracing method for the P4F sensor with $RL = 0.8$ and one point source simulated on the optical axis. The phase screen is generated with an astigmatism aberration (Zernike mode Z_2^2) with magnitude $5 \cdot 10^{-5}$ [m]. The leftmost image shows the back-propagated light field, where the phase screen and aperture planes are indicated by the dash-dotted squares. The geometric centres of the sub-images are indicated by the blue dots. The rightmost image shows the corresponding P4F sensor image, obtained using Algorithm 10.

Algorithm 8 Plenoptic Phase Screen Reconstruction using Ray Tracing

Input: Plenoptic sensor image $I^{P4F} \in \mathbb{R}^{N \times N}$ and its physical size h_{im} , physical lens parameters r_{obj} , r_{mla} , f_{obj} , f_{mla} , number of lenslets M in $(M \times M)$ MLA

Output: Reconstructed phase ϕ

```

1: function PLENOPTICRT
2:   Read the plenoptic sensor image  $I_{P4F} \in \mathbb{R}^{N \times N}$  and obtain grid size  $N$ 
3:   Determine the number of pixels  $P$  in the  $P \times P$  sub-images and round it up to the
      nearest integer:  $P = \mathcal{R}^E \left\{ \frac{2r_{mla}}{h_{im}} N \right\}$ 
4:   Define the propagation stages: phase screen location, the P4F aperture plane, the
      objective lens, the MLA and the image plane.
5:   Set the coordinates of the stages along the optical axis and store in matrix  $Z \in \mathbb{R}^{N^2 \times 5}$ 
6:   Generate the spatial locations of the lenslet and sub-image centres, stored in matrices
       $MlaLoc \in \mathbb{R}^{M^2 \times 5}$  and  $CellLoc \in \mathbb{R}^{M^2 \times 5}$ 
7:   % The sensor image is converted into  $N^2$  light rays whose spatial and angular coordi-
      nates are stored in  $LF$  %
8:   Allocate matrices for the light field  $LF \in \mathbb{R}^{N^2 \times 5 \times 4}$  and pixel intensities  $I_{rt} \in \mathbb{R}^{N^2 \times 1}$ 
9:   for  $i = 1 : M^2$  do
10:    Extract sub-image  $I_i^{P4F} \in \mathbb{R}^{P \times P}$ 
11:    Determine ray pencils  $iL$  corresponding to pixel locations of pixels in  $I_i^{P4F}$ .
12:    Assign each pixel its slope based on the angular spectrum and index-displacements
      of sub-image  $i$ , stored in  $CellLoc(i, 4:5)$ 
13:    Store slopes in  $LF(iL, 5, 3)$  and  $LF(iL, 5, 4)$  for the  $x$ - and  $y$ -directions.
14:    Store the geometric positions of the pixels in  $LF(iL, 5, 1)$  and  $LF(iL, 5, 2)$  for the
       $x$ - and  $y$ -positions
15:    Store the intensity values of the pixels in  $I_{rt}(iL)$ 
16:  end
17:  Remove pencils with an intensity value smaller than 10 (or adjusted by user) and
      determine number of remaining pencil rays  $n$ 
18:  Propagate light field  $LF$  from image plane to the aperture plane using geometrical
      optics model (5-43)
19:  Define a spatial grid for the aperture plane and extract angular coordinates  $\{LF(:, 2, 3), LF(:, 2, 4)\} \subset \mathbb{R}^{n \times 1}$ , removing pencils outside physical range of aperture plane
20:  Propagate the light field back to the location of the phase screen and remove pencil
      rays not illuminating phase screen
21:  Define the phase screen spatial grid and extract the angular coordinates  $\{LF(:, 1, 3), LF(:, 1, 4)\} \subset \mathbb{R}^{n \times 1}$ 
22:  Average the obtained angular coordinates from the phase screen and aperture plane,
      resulting in slope matrices  $\{S_x, S_y\} \subset \mathbb{R}^{M \times M}$ 
23:  Determine the phase gradients, reconstruct the phase and place it in the screen location
24: end

```

5-3 Modelling the Shack-Hartmann and Plenoptic Sensors

As explained in the introduction of this chapter, only the plenoptic 4F sensor is considered throughout this thesis. The 4F configuration of the plenoptic sensor allows for simpler propagation equations and predetermined sub-image locations. The Shack-Hartmann sensor is simulated as well and compared to the performance of the P4F sensor.

The complete simulation algorithms for both sensor will be presented in this section. The extensive demonstration of the supporting toolbox functions in Section 5-2 allows one to reduce the WFS algorithms to a cascade of these functions, greatly improving pseudocode readability. The input parameters required for both algorithms are listed in Table 5-1. Results from the simulation toolbox will be presented in the next sections.

5-3-1 Shack-Hartmann sensor

Simulating the SH WFS is reduced to a propagation from the aperture plane - the MLA front focal plane - to the MLA back focal plane where the digital image is formed. The preliminary conditions of *checkprelim.m*, detailed in Section 5-2-3, perform the necessary checks and provide the aperture plane size h_{ap}^{SH} and grid size N^{SH} .

Function *getpupil.m* then generates aperture plane phase $\phi \in \mathbb{R}^{N^{SH} \times N^{SH}}$ that can be used to define the input field $U_1 = A \odot e^{j\phi} = e^{j\phi} \in \mathbb{R}^{N^{SH} \times N^{SH}}$. Magnitude A is equal to one such that a square collimated beam illuminates the WFS. This complex input field is propagated through an $M \times M$ microlens array by isolating each lenslet, pupil-limiting field U_1 accordingly and propagating it to the image plane with the DFT (5-2).

The propagation itself is handled by the toolbox function *planeprop.m* that employs the DFT to propagate from the front to the back focal plane of a lens. The function checks input arguments to ensure correct propagation, adds padding and generates pupil masks before propagating field U_1 to the image plane. Output field U_2 is trimmed or padded to fit image size requirements and finally converted to digital image I^{SH} .

After the digital image is recorded, function *getPG.m* reads it and retrieves the phase gradients using image first moments (5-26). These gradients are used to reconstruct the phase with toolbox function *getphase.m*, employing either the modal or zonal reconstruction algorithms. The implementation of these algorithms is detailed in Appendix A-5 and Appendix A-6.

The combination of functions *getPG.m* and *getphase.m* reconstruct the phase aberration in the aperture plane. For isoplanatic aberrations this reconstruction is directly used for phase correction. If the aberrations are anisoplanatic, the aperture plane reconstruction is presented to function *getscreen.m* that back-propagates it to the location of the phase screen and reconstructs said screen. Then the simulated aberration is compensated for using the reconstructed phase screen. The correction step for both iso- and anisoplanatic aberrations is performed by function *correctphase.m*.

An overview of the algorithm to simulate the SH image and reconstruct the aberrations is presented in Algorithm 9.

SH image examples To illustrate the workings of Algorithm 9 two examples of the simulated Hartmannogram are shown. The reconstruction and correction is left for the reader to explore in Section 5-4, where they are presented for both the SH and P4F WFS. Important design parameters are listed in Table 5-2. The first example in Figure 5-19 shows the sensor image for a focal distance $f_{mla} = 5\text{mm}$, whereas Figure 5-20 shows it for a focal distance $f_{mla} = 10\text{mm}$.

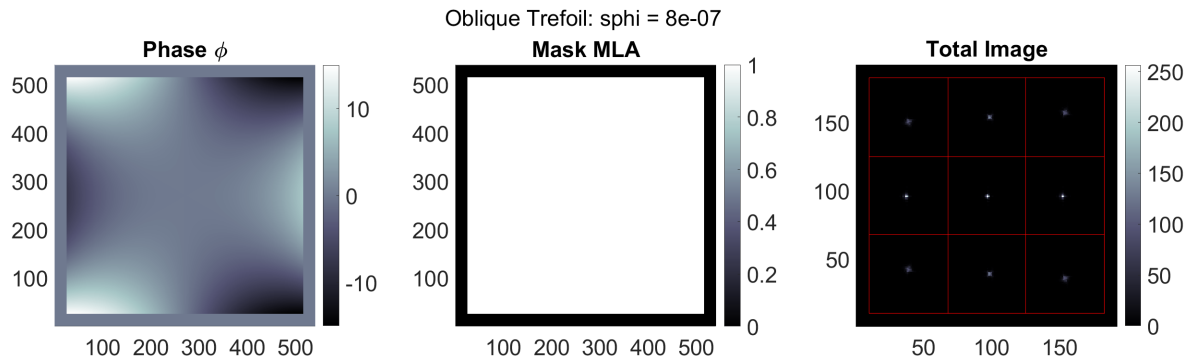


Figure 5-19: An example of simulating the SH sensor image with the *CSS.m* code and Algorithm 9. From left to right: The phase aberration in the aperture plane pupil-limited with the MLA pupil mask, the mask of the MLA itself and the final sensor image. The red outlines in the sensor image show the sub-images. The generated phase screen is a trefoil aberration (Zernike mode Z_3^3 or Z_9) with magnitude $8 \cdot 10^{-7}$ [m]. The design parameters are listed in Table 5-2 under 'Example SH₁'.

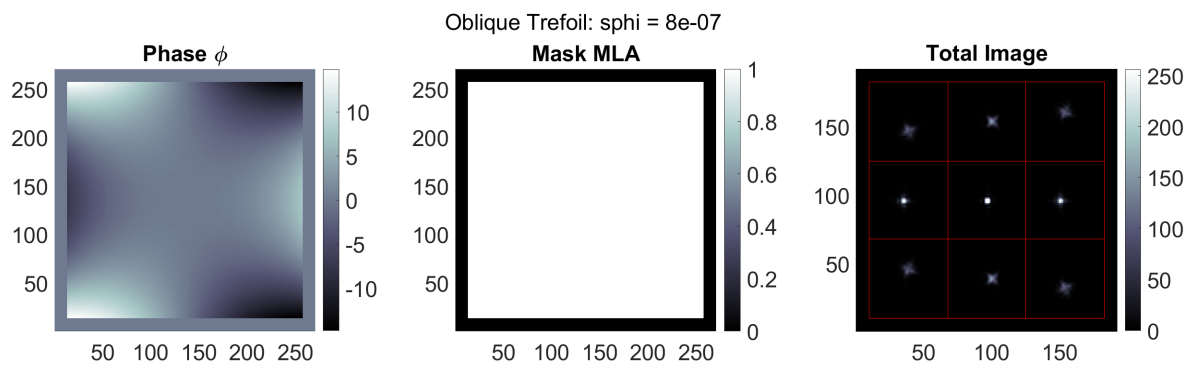


Figure 5-20: An example of simulating the SH sensor image with the *CSS.m* code and Algorithm 9. From left to right: The phase aberration in the aperture plane pupil-limited with the MLA pupil mask, the mask of the MLA itself and the final sensor image. The red outlines in the sensor image show the sub-images. The generated phase screen is a trefoil aberration (Zernike mode Z_3^3 or Z_9) with magnitude $8 \cdot 10^{-7}$ [m]. The design parameters are listed in Table 5-2 under 'Example SH₂'.

Algorithm 9 Simulating SH Sensor Image and Phase Reconstruction**Input:** See Table 5-1

- 1: Parse all parameters and assign variables
- 2: Check the prerequisite conditions - *checkprelim.m*
- 3: Important parameters: aperture plane grid size N^{SH} and physical size h_{ap} , number of simulated point sources np , MLA parameters r_{mla} , f_{mla} and M
- 4: Display sensor performance metrics from Section 3-2-4
- 5: Generate the pupil plane phases $\phi_p \in \mathbb{R}^{N^{SH} \times N^{SH} \times np}$ - *getpupil.m*
- 6: % Simulate the SH sensor image for each point source $O(i, :)$ %
- 7: **for** $i = 1 : np$ **do**
- 8: Determine phase in aperture plane by selecting i -th 2D matrix from ϕ_p : $\phi_{ap} = \phi_p(:, :, i) \in \mathbb{R}^{N^{SH} \times N^{SH}}$
- 9: Determine the MLA lenslet outlines Mla and centre locations $MlaLoc$
- 10: Generate MLA input field: $U_{in} = e^{-j\phi_{ap}} \in \mathbb{C}^{N^{SH} \times N^{SH}}$
- 11: Propagate the input field through the MLA, resulting in image plane field U_{out} , image size h_{im} , sub-image outlines $Cell$ and centre locations $CellLoc$ - *planeprop.m*
- 12: Record the sensor image $I_{c,i} = |U_{out}|^2$ of point source i and digitize it (5-9): $I_i^{SH} = 2^b \frac{I_{c,i}}{\max(I_{c,i})}$
- 13: **end**
- 14: Obtain the total sensor image by summing over all I_i^{SH} : $I^{SH} = \text{sum}(I_i^{SH})$
- 15: Retrieve phase gradients $PG^{SH} \in \mathbb{R}^{M^2 \times 2 \times np}$ for each simulated point source - *getPG.m*
(If $np = \text{using direct image first moments}$, if $np > 1$ using image first moments after cluster detection with *getcluster.m*)
- 16: **for** $i = 1 : np$ **do**
- 17: Select gradients from i -th point source $PG_i \in \mathbb{R}^{M^2 \times 2}$ from PG^{SH}
- 18: Reconstruct the aperture plane phase with either the modal or zonal algorithm specified by parameter '*rec_method*' - *getphase.m*
- 19: Store the reconstruction in 3D matrix $\phi_{ap}^{SH}(:, :, i) \in \mathbb{R}^{M \times M \times np}$
- 20: **end**
- 21: Restore the phase screen. If $RL = 1$ object-based tip/tilt is removed from ϕ_{ap}^{SH} and the phase screen is the average of the corrected ϕ_{ap}^{SH} . If $RL \in \langle 0, 1 \rangle$ the phase screen is found by back propagation and retrieval - *getscreen.m*
- 22: Correct for the phase screen - *correctphase.m*
- 23: Return relevant variables and plot relevant figures

Parameter description	Example SH ₁	Example SH ₂
MLA lenslet pitch $2r_{mla}$	0.3mm	0.3mm
MLA lenslet focal distance f_{mla}	5mm	10mm
Number of lenslets M ($M \times M$ MLA)	3	3
Image sensor pixel size d_{pix}	$5.2\mu\text{m}$	$5.2\mu\text{m}$

Table 5-2: Design parameters for simulating two SH sensor images. Only relevant parameters for the SH design are listed, other light and simulation parameters are left out.

5-3-2 Plenoptic 4F sensor

By setting the plenoptic sensor in 4F configuration the propagation from aperture plane to image can be performed by a direct cascade of DFTs (5-2). The first propagation is performed from aperture plane through the objective lens, the second from MLA front focal plane through the MLA onto the image plane.

The series of functions that lead to a correction of the phase aberration is the same as described for the SH sensor shown in Algorithm 9, with the main difference that two calls to function *planeprop.m* are made. The input field is defined as $U_1 = Ae^{j\phi} = e^{j\phi} \in \mathbb{R}^{N^{P4F} \times N^{P4F}}$, where ϕ is the simulated aperture plane phase obtained from function *getpupil.m*. Just as described in Section 5-3-1 for the SH sensor magnitude A is set to 1 and grid size N^{P4F} is determined by the preliminary checks in *checkprelim.m*.

The first call to *planeprop.m* propagates input field U_1 through the objective lens. Field U_2 in the objective lens back focal plane is then propagated through the MLA in the second call to *planeprop.m*, resulting in complex field U_3 in the image. In both calls *planeprop.m* checks the input parameters for correct propagation, adds individual padding and generates pupil masks.

The algorithm that simulates the P4F sensor image and phase reconstruction is presented in Algorithm 10 as an overview in pseudocode.

P4F image examples Similar to Section 5-3-1 two examples of the simulated P4F image are shown, leaving the reconstruction and correction steps for Section 5-4. Important design parameters are listed in Table 5-3. Both examples differ from the SH images in Section 5-3-1 as the aberration defined is stronger and more lenslets are considered in the MLA such that a clearer image is obtained. The obtained images are shown in Figure 5-21 for an MLA lenslet focal distance $f_{mla} = 5\text{mm}$ and for $f_{mla} = 10\text{mm}$ in Figure 5-22. Note that the pupil mask aperture plane - defined with the size of the objective lens- lets only the central part of the trefoil aberration through.

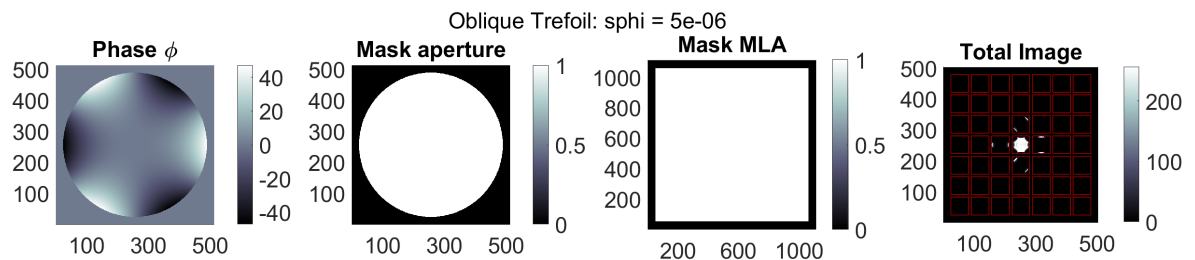


Figure 5-21: An example of simulating the P4F sensor image with the *CSS.m* code and Algorithm 10. From left to right: The phase aberration in the aperture plane pupil-limited by the objective lens pupil mask, the mask of the objective lens itself, the MLA pupil mask and the final sensor image. The red outlines in the sensor image show the sub-images. The generated phase screen is a trefoil aberration (Zernike mode Z_3^3 or Z_9) with magnitude $2 \cdot 10^{-6}$ [m]. The design parameters are listed in Table 5-3 under 'Example P4F₁'.

Algorithm 10 Simulating P4F Sensor Image and Phase Reconstruction**Input:** See Table 5-1

```

1: Parse all parameters, assign variables and check prerequisite conditions - checkprelim.m
2: Important parameters: aperture plane grid size  $N^{P4F}$ , number of simulated point sources
    $np$ , objective lens and MLA parameters  $r_{obj}$ ,  $f_{mla}$ ,  $r_{mla}$ ,  $f_{mla}$  and  $M$ 
3: Display sensor performance metrics from Section 3-3-4
4: Generate the pupil plane phases  $\phi_p \in \mathbb{R}^{N^{P4F} \times N^{P4F} \times np}$  - getpupil.m
5: % Simulate the P4F sensor image for each point source  $O(i, :)$  %
6: for  $i = 1 : np$  do
7:   Determine phase in aperture plane by selecting  $i$ -th 2D matrix from  $\phi_p$ :  $\phi_{ap} =$ 
      $\phi_p(:, :, i) \in \mathbb{R}^{N^{P4F} \times N^{P4F}}$ 
8:   Determine the location of the objective lens and define its input field in the front focal
     plane  $U_{in} = e^{-j\phi_{ap}} \in \mathbb{C}^{N^{P4F} \times N^{P4F}}$ 
9:   Propagate  $U_{in}$  through the objective lens onto the MLA front focal plane - planeprop.m
10:  Determine the MLA lenslet outlines  $Mla$  and centre locations  $MlaLoc$ 
11:  Set the objective lens output field as input field for the MLA and propagate it through
     the MLA, resulting in image plane field  $U_{out}$ , image size  $h_{im}$ , sub-image outlines  $Cell$ 
     and centre locations  $CellLoc$  - planeprop.m
12:  Record the sensor image  $I_{c,i} = |U_{out}|^2$  of point source  $i$  and digitize it (5-9):  $I_i^{SH} =$ 
      $2^b \frac{I_{c,i}}{\max(I_{c,i})}$ 
13: end
14: Obtain the total sensor image by summing over all  $I_i^{P4F}$ :  $I^{P4F} = \sum (I_i^{P4F})$ 
15: % Retrieve phase gradients and reconstruct the phase screen for each point source %
16: if  $P4Fslopes = \text{"CAFADIS"}$  then
17:   % Aperture plane reconstruction with CAFADIS model and back-propagation %
18:   Retrieve phase gradients  $PG^{P4F} \in \mathbb{R}^{P^2 \times 2 \times np}$  for each simulated point source, with
      $P^2$  the number of pixels in each P4F sub-image - getPG.m (If  $np = 1$  using CAFADIS
     slopes, if  $np > 1$  using CAFADIS slopes after plenoptic reshuffling with Algorithm 3)
19:   for  $i = 1 : np$  do
20:     Select gradients from  $i$ -th point source  $PG_i \in \mathbb{R}^{P^2 \times 2}$  from  $PG^{P4F}$ 
21:     Reconstruct the aperture plane phase with either the modal or zonal algorithm
     specified by parameter 'rec_method' - getphase.m
22:     Store the reconstruction in 3D matrix  $\phi_{ap}^{P4F}(:, :, i) \in \mathbb{R}^{P \times P \times np}$ 
23:   end
24:   Restore the phase screen. If  $RL = 1$  object-based tip/tilt is removed from  $\phi_{ap}^{P4F}$  and
     the phase screen is the average of the corrected  $\phi_{ap}^{P4F}$ . If  $RL \in (0, 1)$  the phase screen
     is found by back propagation and retrieval - getscreen.m
25: else if  $P4Fslopes = \text{"RayTracing"}$  then
26:   % Assign light rays to all illuminated plenoptic pixels and propagate them back to the
     phase screen location and reconstruct the screen locally %
27:   Reconstruct the phase screen by propagating light rays back through the P4F sensor
     using geometrical optics - plenopticRT.m
28: end
29: Correct for the phase screen - correctphase.m
30: Return relevant variables and plot relevant figures

```

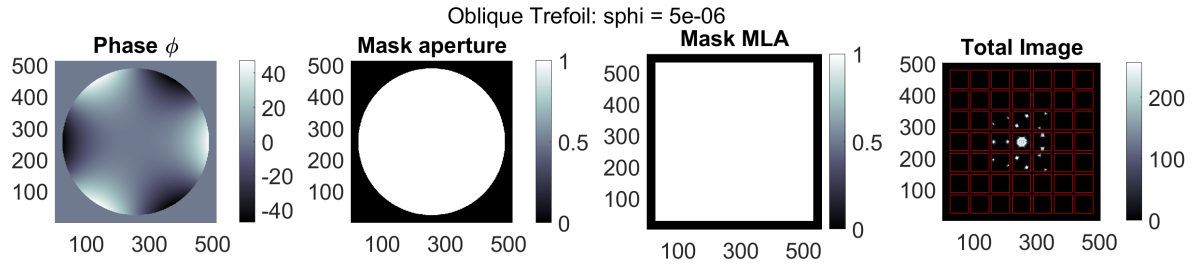


Figure 5-22: An example of simulating the P4F sensor image with the *CSS.m* code and Algorithm 10. From left to right: The phase aberration in the aperture plane pupil-limited by the objective lens pupil mask, the mask of the objective lens itself, the MLA pupil mask and the final sensor image. The red outlines in the sensor image show the sub-images. The generated phase screen is a trefoil aberration (Zernike mode Z_3^3 or Z_9) with magnitude $2 \cdot 10^{-6}$ [m]. The design parameters are listed in Table 5-3 under 'Example P4F₂'.

Parameter description	Example P4F ₁	Example P4F ₂
Objective lens pitch $2r_{obj}$	2.1mm	2.1mm
Objective lens focal distance f_{obj}	30mm	60mm
MLA lenslet pitch $2r_{mla}$	0.3mm	0.3mm
MLA lenslet focal distance f_{mla}	5mm	10mm
Number of lenslets M ($M \times M$ MLA)	7	7
Image sensor pixel size d_{pix}	$5.2\mu\text{m}$	$5.2\mu\text{m}$

Table 5-3: Design parameters for simulating two P4F sensor images. Only relevant parameters for the P4F design are listed, other light and simulation parameters are left out.

5-4 Isoplanatic Simulation Results

The simulation of the SH and P4F WFS is performed for a number of situations and phase screens, of which the most prominent results are presented here. This section considers the results for isoplanatic aberrations, which means that the phase screen is generated inside the pupil plane and $RL = 1$. At first Zernike modes are used to describe this phase screen, followed by random Kolmogorov screens. The aberrations in this section are reconstructed using both modal and zonal algorithms. Anisoplanatic aberrations will be discussed in Section 5-6.

In these simulations the aperture planes of both Shack-Hartmann and Plenoptic 4F sensor are kept equal, such that both have the same aberration defined in their aperture plane.

5-4-1 Isoplanatic Zernike Modes

To briefly show the phase reconstruction results for Zernike modes two modes are generated in the aperture planes of each sensor and reconstructed. Each of the presented results consists of four images: the phase aberration in the pupil plane, the WFS reconstruction, the phase profile of the DM and the residual phase after DM correction. It should be noted that the

DM phase profile is simply the WFS reconstruction resampled at the grid size of the pupil plane, as the correction algorithm explained in Section 5-2-9 assumes an ideal DM. The design parameters of these simulations are listed in Table 5-4, together with the number of modes considered in the modal reconstruction.

At first the SH sensor is simulated for an astigmatism aberration - Zernike mode Z_2^2 - with magnitude $3 \cdot 10^{-6}$ [m] defined in the aperture plane. Figure 5-23 shows the results for the modal algorithm, whereas Figure 5-24 shows the zonal reconstructed results. The same approach is taken for a defocus aberration - Zernike mode Z_2^0 - with the same magnitude. The corresponding reconstruction results are presented in Figure 5-25 and Figure 5-26.

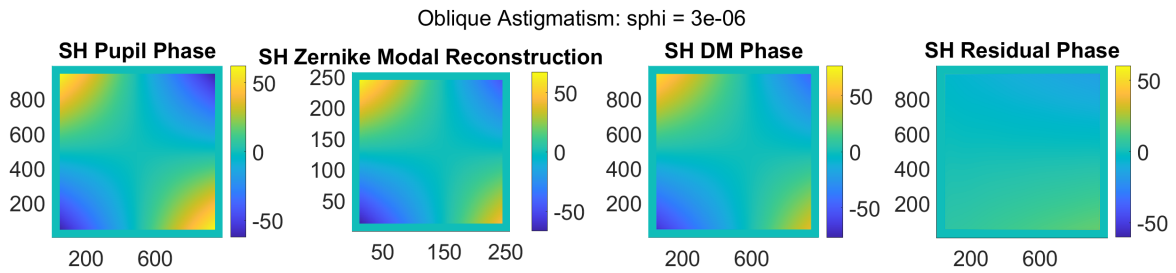


Figure 5-23: Modal reconstruction of simulated SH image with Z_2^2 and magnitude $3 \cdot 10^{-6}$ [m].

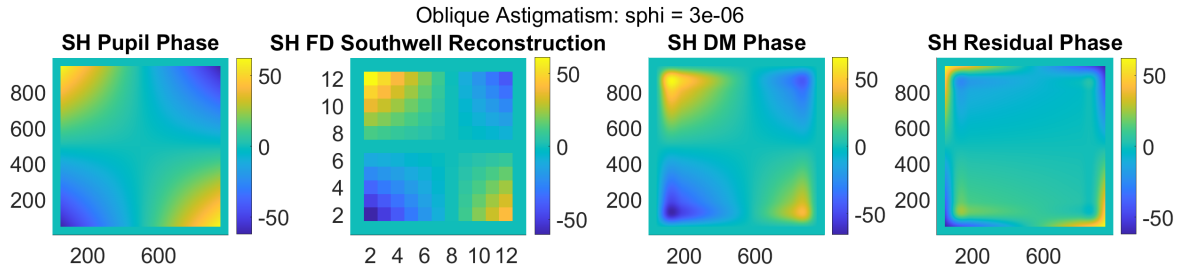


Figure 5-24: Zonal reconstruction of simulated SH image with Z_2^2 and magnitude $3 \cdot 10^{-6}$ [m].

A clear difference and benefit of the modal reconstruction algorithm is the grid size of the WFS reconstructed phase. In the zonal algorithm the number of reconstructed phase points depends on the number of measurements available. This means that the DM profile will be drastically resampled, for example with factor ~ 80 in Figure 5-24. The modal algorithm does not suffer nearly as much from this drawback, as the Zernike modes can be generated at any grid size requested. In the Toolbox this generation is limited to a maximum grid of 256×256 , which means that a resampling with factor ~ 3 would be required to obtain the DM profile in Figure 5-23.

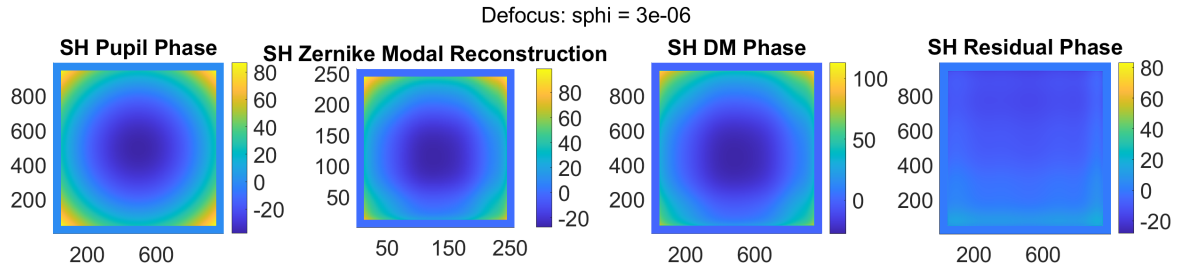


Figure 5-25: Modal reconstruction of simulated SH image with Z_2^0 and magnitude $3 \cdot 10^{-6}$ [m].

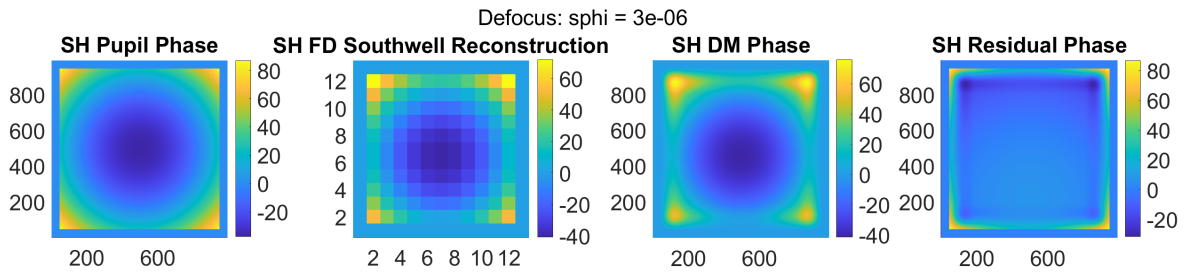


Figure 5-26: Zonal reconstruction of simulated SH image with Z_2^0 and magnitude $3 \cdot 10^{-6}$ [m].

The same behaviour is observed in Figure 5-25 and Figure 5-26. The decrease in reconstruction grid size clearly affects the correction and thus the phase residual - the modal phase residual shows a much better modal reconstruction. Additionally, the decrease in grid size density undersamples the corners such that the zonal phase reconstruction is clearly weaker than the modal counterpart.

The same aberrations are defined for the P4F WFS, resulting in the reconstruction results presented in Figure 5-27 up to Figure 5-30. The number of pixels in each P4F sub-image is fixed to a maximum of 50 pixels such that reconstruction is achieved at reasonable speed. This means that the isoplanatic phase reconstruction of the P4F sensor - using the CAFADIS slope model (3-12) - uses much more measurements than the SH sensor. This is reflected in the phase profile of the DM, which requires resampling with a lower factor. Additionally, the objective lens requires a lower grid size for correct propagation and thus the aperture plane grid size is lower as opposed to the SH sensor.

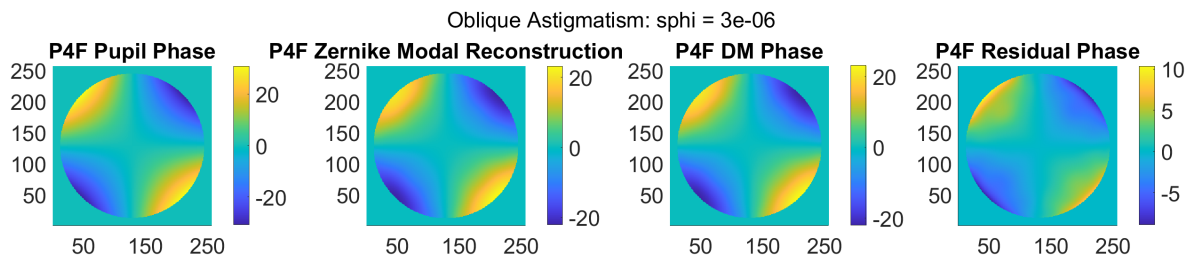


Figure 5-27: Modal reconstruction of simulated P4F image with Z_2^2 and magnitude $3 \cdot 10^{-6}$ [m].

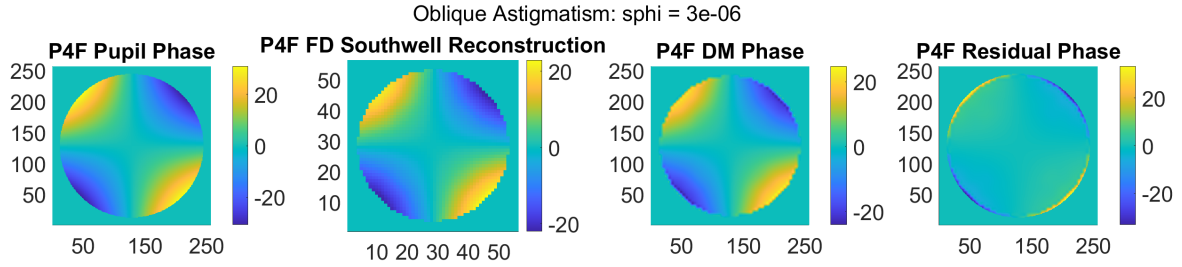


Figure 5-28: Zonal reconstruction of simulated P4F image with Z_2^2 and magnitude $3 \cdot 10^{-6}$ [m].

A primary conclusion that can be drawn from these results is that both zonal and modal reconstruction algorithms work. The phase reconstruction is not as expected, but both algorithms reconstruct the shape of the aberration very well. A second observation is that, despite the significant increase in measurement count, the modal reconstruction outperforms the zonal algorithm. Based on these two arguments, a smaller upsampling required for phase correction and an overall better phase reconstruction, leads to the conclusion that the modal algorithm outperforms zonal reconstruction in correcting for isolated Zernike modes.

The next simulation investigates the same for randomly generated Kolmogorov screens.

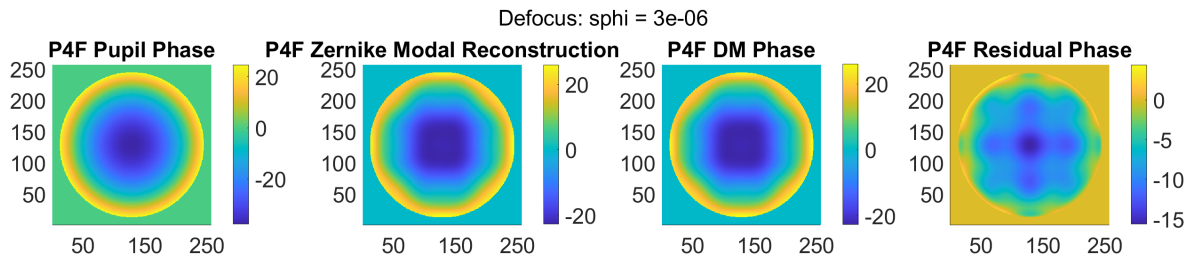


Figure 5-29: Modal reconstruction of simulated P4F image with Z_2^0 and magnitude $3 \cdot 10^{-6}$ [m].

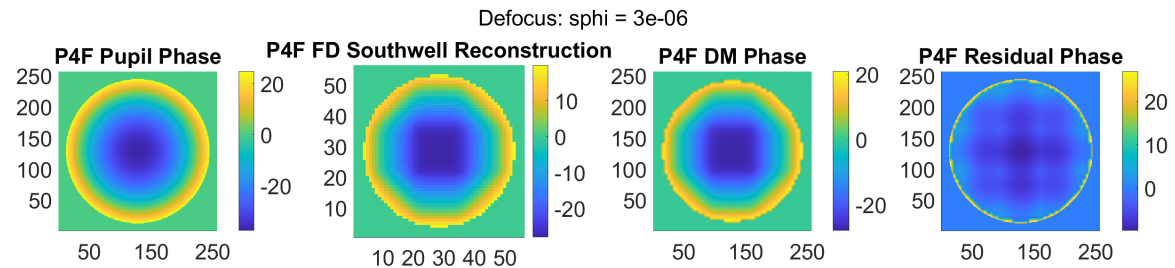


Figure 5-30: Zonal reconstruction of simulated P4F image with Z_2^0 and magnitude $3 \cdot 10^{-6}$ [m].

Parameter description	SH sensor	P4F sensor
Objective lens pitch $2r_{obj}$	-	2.1mm
Objective lens focal distance f_{obj}	-	60mm
MLA lenslet pitch $2r_{mla}$	0.3mm	
MLA lenslet focal distance f_{mla}	10mm	
Number of MLA lenslets M ($M \times M$ MLA)	11	
Image sensor pixel size d_{pix}	$5.2\mu\text{m}$	
Wavelength of light λ	500nm	
Number of reconstructed Zernike modes	60	200

Table 5-4: Relevant parameters for isoplanatic phase reconstruction in the SH and P4F sensors.

5-4-2 Isoplanatic Kolmogorov Screens

The reconstruction results for the Kolmogorov screens will be presented in a similar fashion to Section 5-4-1. Both modal and zonal reconstruction algorithms are employed and design parameters of Table 5-4 are retained. At first a Kolmogorov screen with Fried parameter $r_0 = 5 \cdot 10^{-5}$ [m] is generated in the pupil plane, imaged by the SH sensor and reconstructed. The results are shown in Figure 5-31 and Figure 5-32. The Kolmogorov screens in Figure 5-31 and Figure 5-32 contain larger patches of equal phase as opposed to the Zernike modes, which is reflected by the small difference between the modal and zonal phase reconstructions. In fact, the lower grid size in the zonal reconstruction resulted in an overall smoother residual.

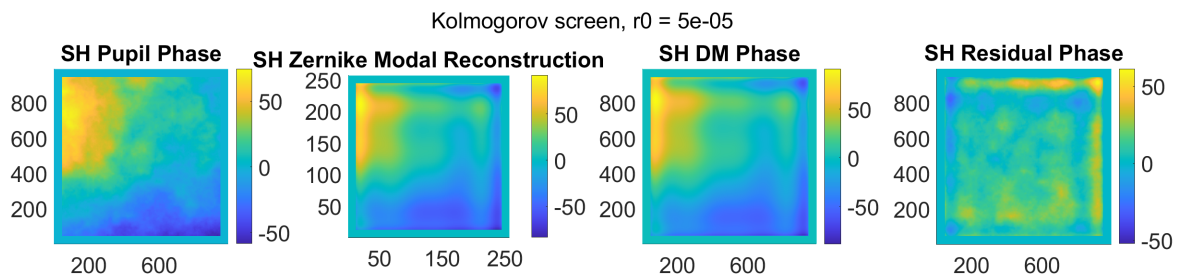


Figure 5-31: Modal reconstruction of a random Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m] and reconstructed using the SH sensor image.

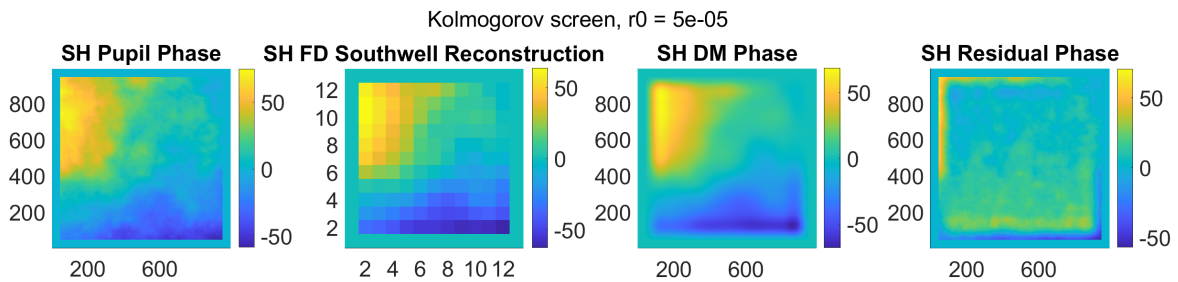


Figure 5-32: Zonal reconstruction of a random Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m] and reconstructed using the SH sensor image.

The next Kolmogorov screen is stronger and generated with $r_0 = 3 \cdot 10^{-5}$ [m]. Although the zonal reconstruction captures the positive-valued phase regions of the screen well, the modal algorithm captures both negative-valued regions and intermittent details better.

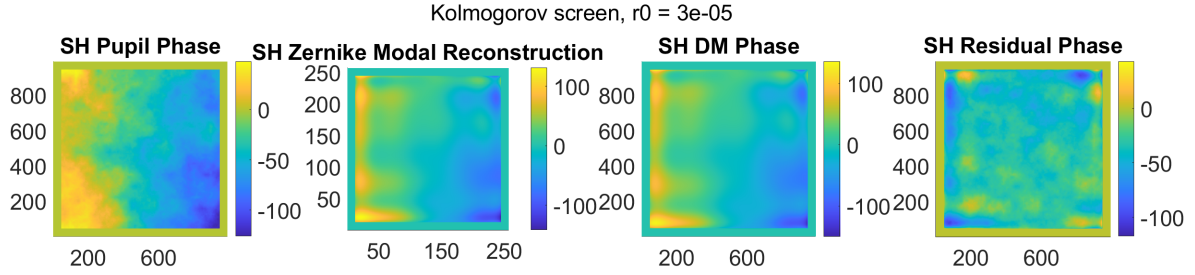


Figure 5-33: Modal reconstruction of a random Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m] and reconstructed using the SH sensor image.

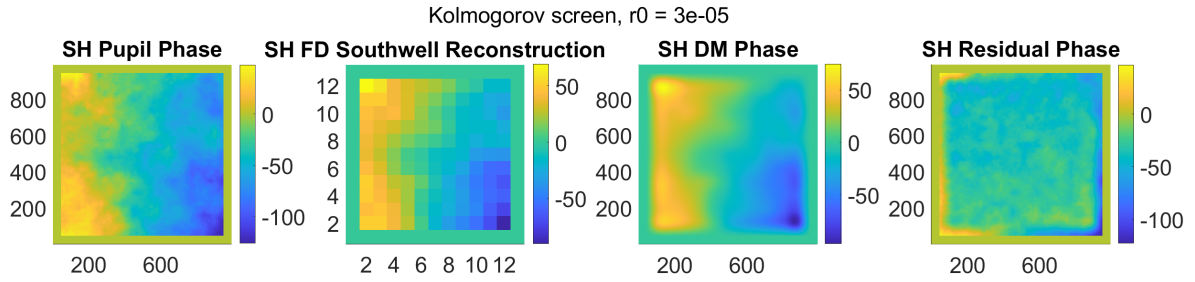


Figure 5-34: Zonal reconstruction of a random Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m] and reconstructed using the SH sensor image.

The same Kolmogorov screens are generated for the P4F sensor. The following results show that an increase in the number of obtained measurements significantly increases the reconstruction performance in both modal and zonal algorithms.

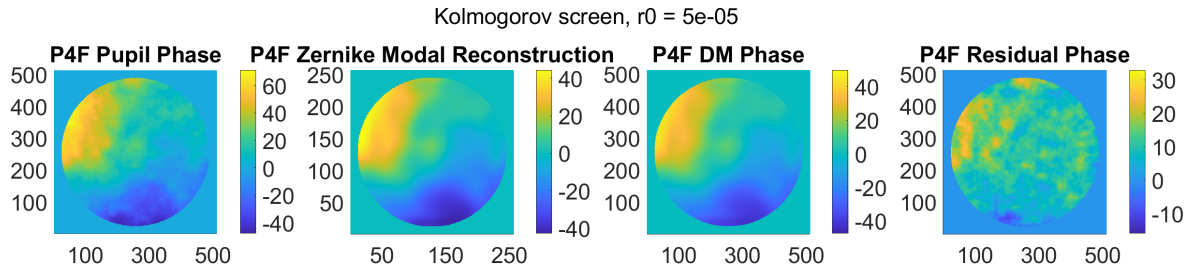


Figure 5-35: Modal reconstruction of a random Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m] and reconstructed using the P4F sensor image.

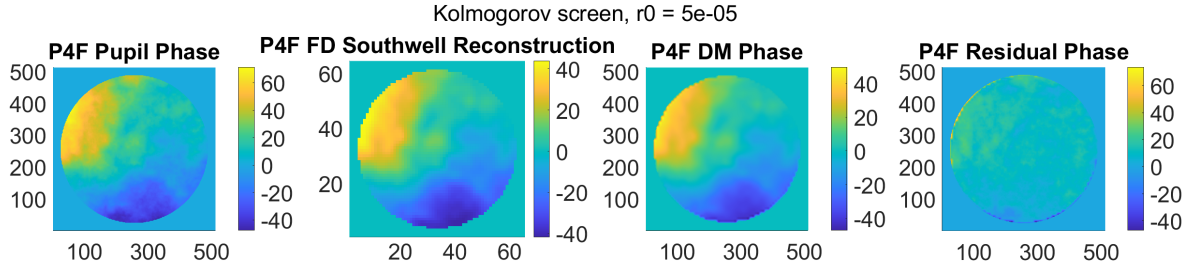


Figure 5-36: Zonal reconstruction of a random Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m] and reconstructed using the P4F sensor image.

The obtained residuals show that the zonal phase reconstruction fails at the edge of the aperture. This means that the edges of the corresponding residual are much stronger than its central part, overshadowing the central details. Overall it can be concluded that the modal and zonal phase reconstructions lead to the same phase residual, without considering the pixels around the aperture edge. Another interesting observation is that the zonal reconstruction is significantly better in terms of reconstructing the Kolmogorov screen details.

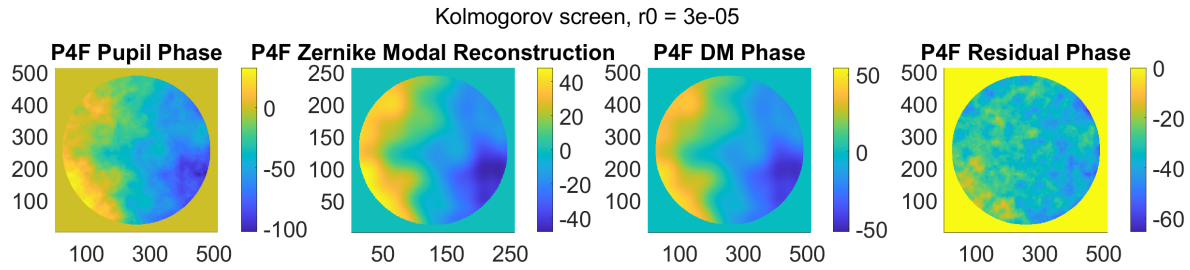


Figure 5-37: Modal reconstruction of a random Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m] and reconstructed using the P4F sensor image.

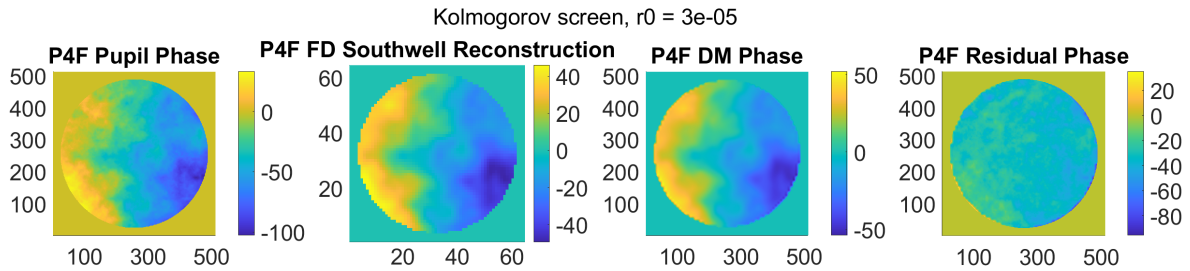


Figure 5-38: Zonal reconstruction of a random Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m] and reconstructed using the P4F sensor image.

A closer look is taken at the modal phase reconstruction and how the orthonormal Zernike coefficients change between pupil plane phase and its reconstruction. Function *checkcoeffs.m* generates M orthonormal $N \times N$ Zernike basis functions and stores them in matrix $Z_{\perp} \in \mathbb{R}^{N^2 \times M}$. An estimate is made of the orthonormal coefficients through a pseudo-inverse, equivalent to the basic MVUE (A-62):

$$\tilde{\alpha} = (Z_{\perp}^T Z_{\perp})^{-1} Z_{\perp}^T \phi,$$

where $\phi \in \mathbb{R}^{N^2 \times 1}$ is the vectorized phase and $\tilde{\alpha}$ the estimated coefficients. The coefficients of both Kolmogorov screen and modal reconstruction are computed, referred to as $\tilde{\alpha}_{ref}$ and $\tilde{\alpha}_\phi$ respectively. Figure 5-39 and Figure 5-40 show how these coefficients change between reference and reconstruction. Here the absolute value of the Kolmogorov coefficients $\tilde{\alpha}_{ref}$ are plotted against the absolute value of the difference $|\tilde{\alpha}_{ref} - \tilde{\alpha}_\phi|$. The magnitude of these differences is clarified by plotting it with a reversed (red) y -axis. This means that smaller red bars point out smaller absolute differences between the reference and reconstruction, indicating a better reconstruction of said mode. Figure 5-39 shows the coefficient decomposition for the Kolmogorov screen generated with $r_0 = 5 \cdot 10^5$ [m], for both the SH and P4F sensors. Figure 5-40 shows the same but for the stronger Kolmogorov screen, generated with Fried parameters $r_0 = 3 \cdot 10^{-5}$ [m].

From both figures it is clear that the modal reconstruction is generally mismatched on the coefficients. From the reconstruction results above it is evident that the aberration profile is reconstructed well, but there is a disagreement on the phase values. This is reflected by the significant absolute difference in all plots.

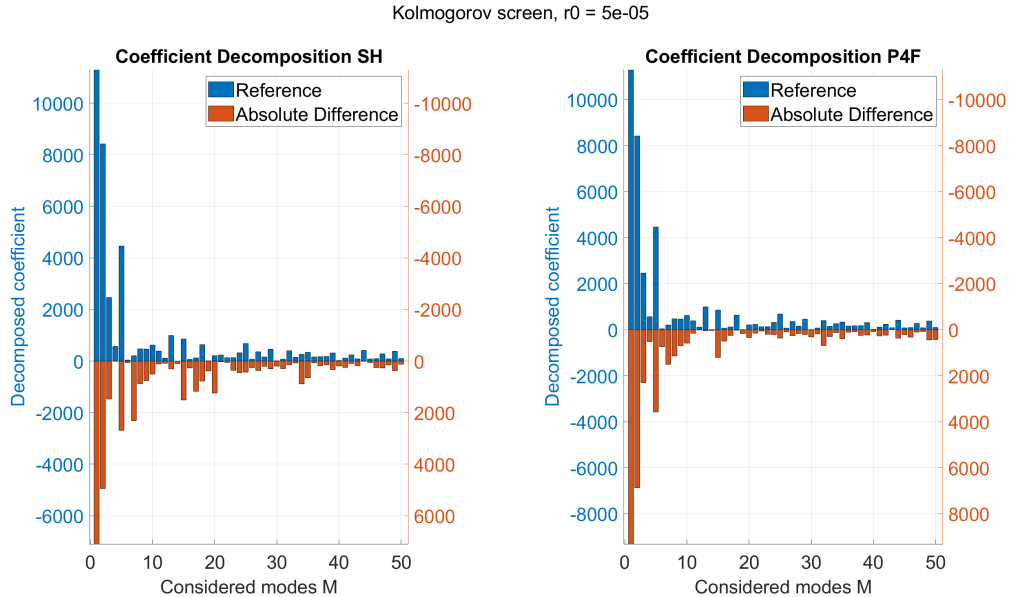


Figure 5-39: The modal coefficient decomposition of the SH sensor (left) and the P4F sensor (right), for a Kolmogorov screen generated with $r_0 = 5 \cdot 10^{-5}$ [m]. The absolute coefficients of the reference phase are shown in blue, the absolute difference between reference and reconstruction coefficients in red. The latter is plotted with its own vertical axis, such that the magnitude of the absolute difference per mode is clearly illustrated. Smaller red bars indicate smaller differences and thus a better reconstruction per mode. The decomposition of the first 50 modes is shown.

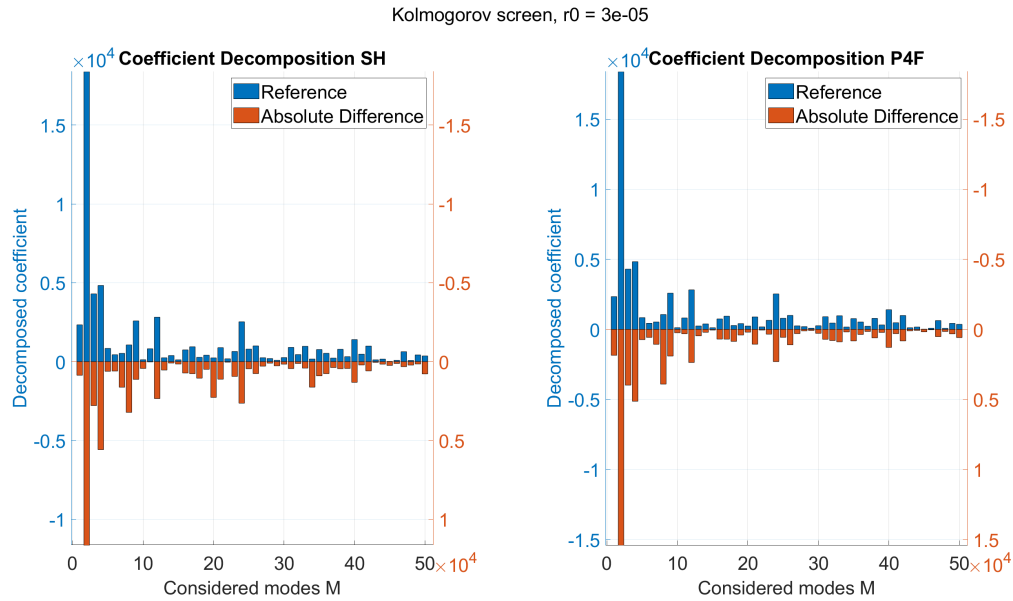


Figure 5-40: The modal coefficient decomposition of the SH sensor (left) and the P4F sensor (right), for a Kolmogorov screen generated with $r_0 = 3 \cdot 10^{-5}$ [m]. The absolute coefficients of the reference phase are shown in blue, the absolute difference between reference and reconstruction coefficients in red. The latter is plotted with its own vertical axis, such that the magnitude of the absolute difference per mode is clearly illustrated. Smaller red bars indicate smaller differences and thus a better reconstruction per mode. The decomposition of the first 50 modes is shown.

The results obtained from isoplanatic reconstruction, for both modal and zonal algorithms, lead to the following conclusions:

- If the number of measurements is scarce (~ 11), the modal algorithm is better. It captures the stronger phase values and details better and does not require significant resampling for correction. The zonal algorithm results in a smoother phase residual due to its significant resampling but misses larger phase values and intermittent screen details. Additionally, its reconstruction fails at the edges of the aperture.
- If a sufficient number of measurements is available (~ 50), both algorithms perform very similar. The zonal algorithms reconstruction the phase screen details better than the modal counterpart, but suffers from the same aperture edge problems as it did before.
- Neither reconstruction reduces the residual as expected. To obtain a flatter phase residual *iterative correction* is applied which will be detailed below.

5-4-3 Iterative Isoplanatic Phase Correction

The single-step isoplanatic phase corrections performed above does not flatten the phase residual as expected. This reconstruction mismatch is attributed to the imaging process and phase gradients retrieval. The most significant contributions to this mismatch are noise in the pixel values, DFT pixel shifts and truncation errors.

The first contribution results from light economy and is intrinsic to SH and P4F sensors, as explained in Section 3-2-2. The second originates from the discrete Fourier transform used in the imaging process. Depending on the field grid sizes the centres of these propagated fields may be shifted by a pixel or two, possibly resulting in a faulty gradient retrieval. The final contribution is a collection of multiple truncation errors, for example representing phase aberrations and propagation by discrete pixels, representing the phase by a finite number of modes M in the modal reconstruction or by a finite number of gradient measurements in the zonal algorithm.

To improve performance the phase reconstruction and correction will be performed iteratively. At each iteration the input phase screen is the residual phase aberration of the previous iteration. The stopping criterion for this iterative correction is that the change in the residual's RMSE of the past two iterations is less than 0.1. A second stopping criterion is set for a maximum number of 6 iterations. These stopping criteria are determined empirically, concentrating on quick and significant improvements instead of perfect corrections.

Indifferent of chosen parameters the closed-loop phase reconstruction is performed by the zonal algorithm. Initial tests of this closed-loop result in failed modal reconstructions after the first iteration, whereas the zonal algorithm does not fail. This failure is attributed to the finite number of modes considered in the modal reconstruction, which is upper-limited by:

$$M \leq \max\left(\frac{1}{2}N^2, 200\right),$$

where N^2 is the number of slope measurement in either x - or y -direction. This criterion is derived and explained in Section 5-2-7. This means that a maximum of 200 orthonormal Zernike modes can be reconstructed by the implemented algorithm.

During the first iteration the modal reconstruction is able to identify the dominant modes in the phase screen but cannot estimate the aberration completely - the same counts for the zonal algorithm. The residual phase mostly contains higher-order oscillations and smaller details, which can be described by higher-order Zernike modes. In the second iteration the modal algorithm would require more than 200 modes to describe the remaining residual. As these are unavailable, the modal algorithm fails to estimate the residual aberrations.

The zonal algorithm does not rely on this modal representation, which is reflected by its ability to resolve phase screen details better. It is limited by the number of measurements available which leads to large resampling factors, as explained above. From the second iteration on, the residual aberrations are much smoother and flatter than the original phase screens. As such, larger resampling factors are considered a minor hindrance.

To summarize, the iterative reconstruction algorithm implements the modal algorithm during the first iteration, to remove most dominant contributions in the aberrations. From the second iteration on the zonal algorithm is used instead to further flatten out the phase residual. The iterative algorithm is tested for an astigmatism aberration - Zernike mode Z_2^2 - with magnitude $6 \cdot 10^{-5}$ [m]. The results for the first iteration, using modal reconstruction are shown in Figure 5-41, whereas the results of the final iteration are shown in Figure 5-42, obtained using zonal reconstruction.

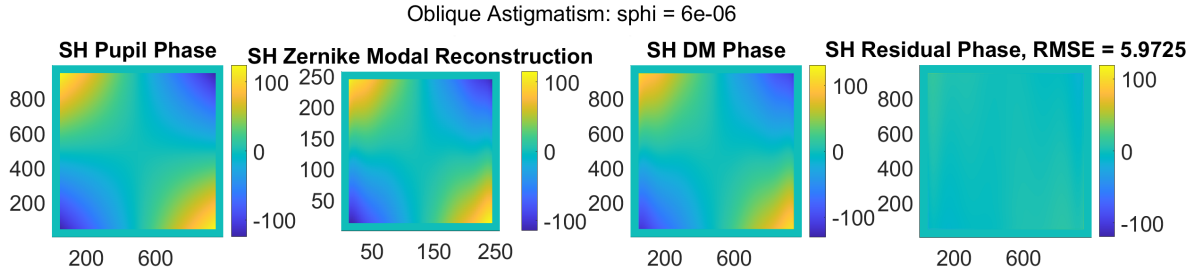


Figure 5-41: Isoplanatic iterative phase reconstruction for the SH sensor with a Zernike Z_2^2 aberration and $6 \cdot 10^{-6}$ [m] magnitude defined in the aperture plane. Iteration 1 out of 6.

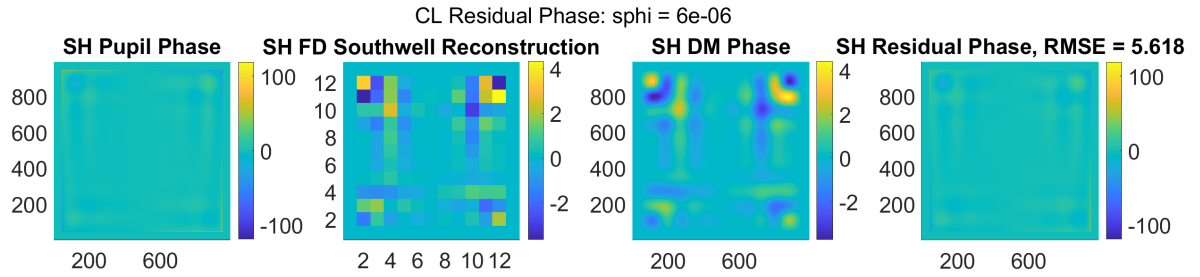


Figure 5-42: Isoplanatic iterative phase reconstruction for the SH sensor with a Zernike Z_2^2 aberration and $6 \cdot 10^{-6}$ [m] magnitude defined in the aperture plane. Iteration 6 out of 6.

The iterative reconstruction scheme does not reduce the residual phase as expected. It is most likely that the spatial resolution of the SH sensor cannot resolve the remaining smaller details of the residual phase.

The next images show the iterative results for a Kolmogorov screen, generated with Fried parameter $r_0 = 5 \cdot 10^{-5}$ [m]. The obtained results already show that the iterative scheme improves the phase reconstruction by 20 %. The reconstruction of the final iteration shows that the SH does not sense the remaining residual well enough for correct reconstruction.

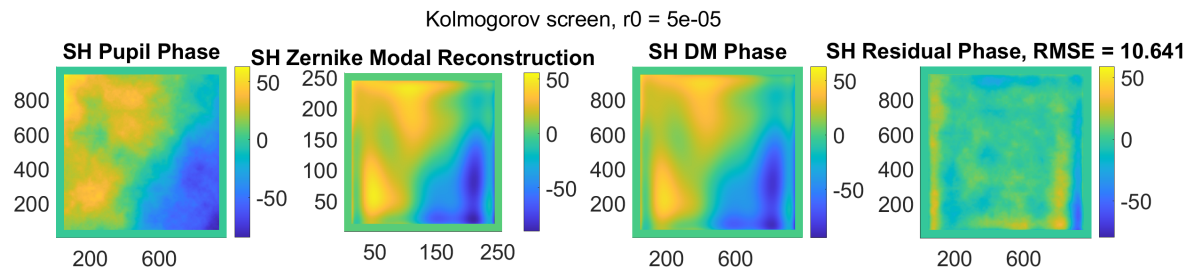


Figure 5-43: Isoplanatic iterative phase reconstruction for the SH sensor with a Kolmogorov screen and $r_0 = 5 \cdot 10^{-5}$ [m] defined in the aperture plane. Iteration 1 out of 6.

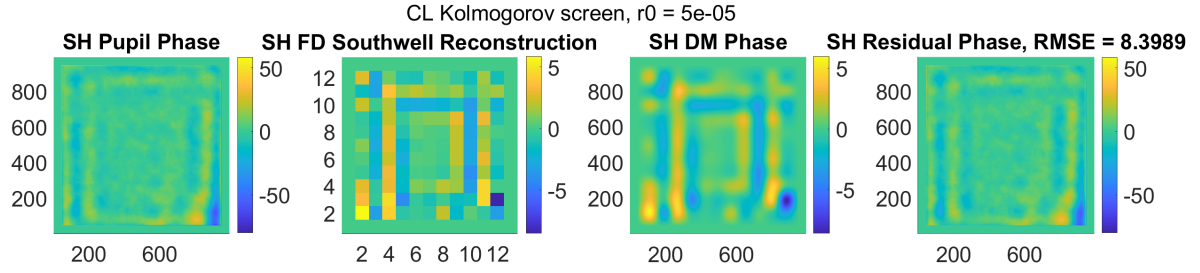


Figure 5-44: Isoplanatic iterative phase reconstruction for the SH sensor with a Kolmogorov screen and $r_0 = 5 \cdot 10^{-5}$ [m] defined in the aperture plane. Iteration 6 out of 6.

A final test is performed for a stronger Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m]. These results are shown below. As expected, the stronger aberration results in a higher RMSE but also an increased performance between iteration 1 and 5 of more than 40 %. Despite the improved performance of the iterative algorithm, it is clear that the phase aberrations cannot be fully corrected for.

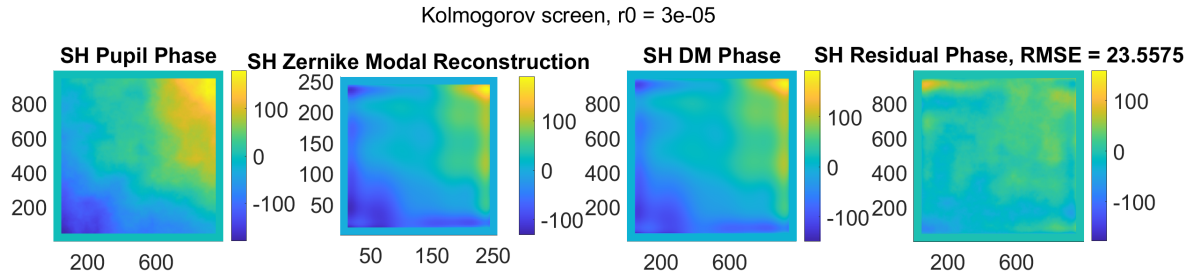


Figure 5-45: Isoplanatic iterative phase reconstruction for the SH sensor with a Kolmogorov screen and $r_0 = 3 \cdot 10^{-5}$ [m] defined in the aperture plane. Iteration 1 out of 6.

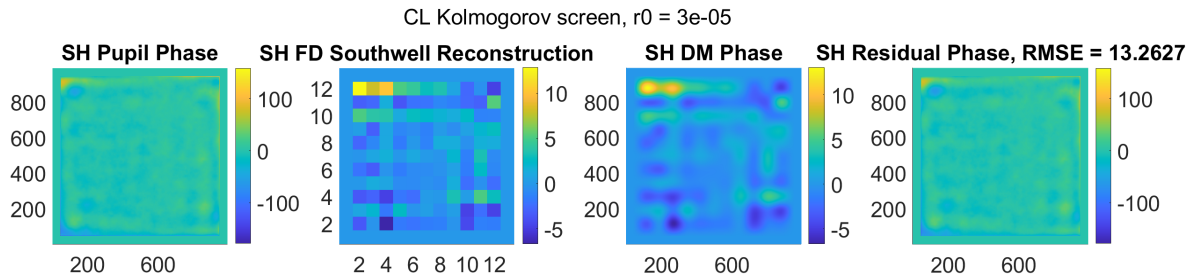


Figure 5-46: Isoplanatic iterative phase reconstruction for the SH sensor with a Kolmogorov screen and $r_0 = 3 \cdot 10^{-5}$ [m] defined in the aperture plane. Iteration 6 out of 6.

The algorithm's inability to further reduce the residual can be attributed to two factors:

1. The phase reconstruction errors identified in the single-step corrections. Iterative reconstruction would only help reduce these errors instead of solving them entirely.
2. The scattering of pixels over the sensor image. Higher-order oscillations that remain in the phase residual disperse the intensity patterns over the sub-images. This is especially detrimental for the image first moment algorithm.

The latter can be reinforced by observing the SH sensor images for the first and last iteration of the $r_0 = 3 \cdot 10^{-5}$ [m] Kolmogorov screen, shown in Figure 5-47 and Figure 5-48. Without the first phase reconstruction the Hartmannogram in Figure 5-47 already shows significant spreading per sub-image, something that can be identified for all simulated Kolmogorov screens due to the higher-order phase oscillations. In the 5-th iteration this spreading is only worsened, most likely leading to inaccurate centroid estimations and insufficient phase reconstructions.

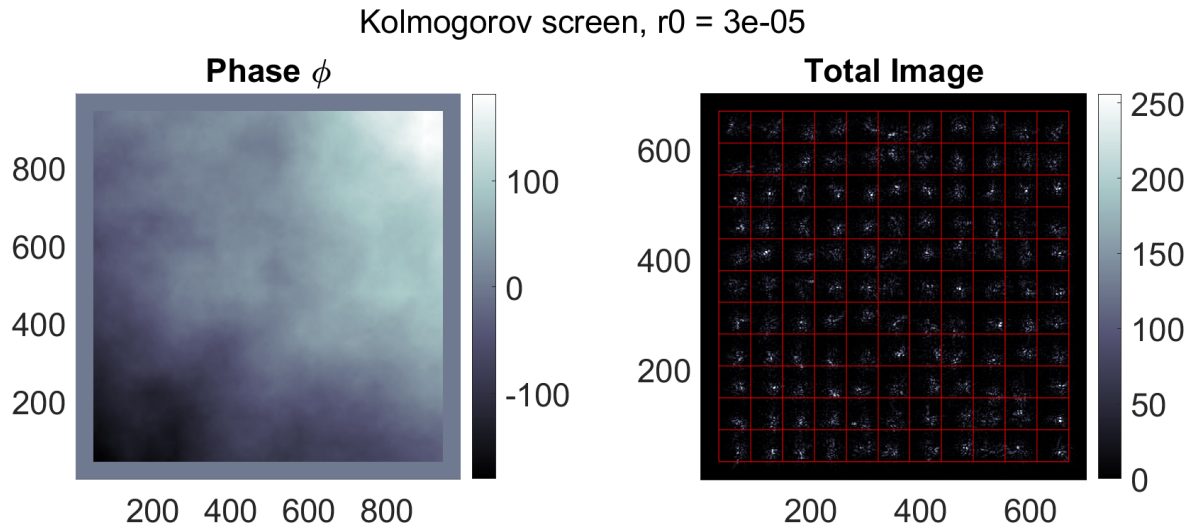


Figure 5-47: Iterative SH sensor simulation of a Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m]. The input phase aberration is shown on the left, the final SH sensor image on the right. Iteration 1 out of 6.

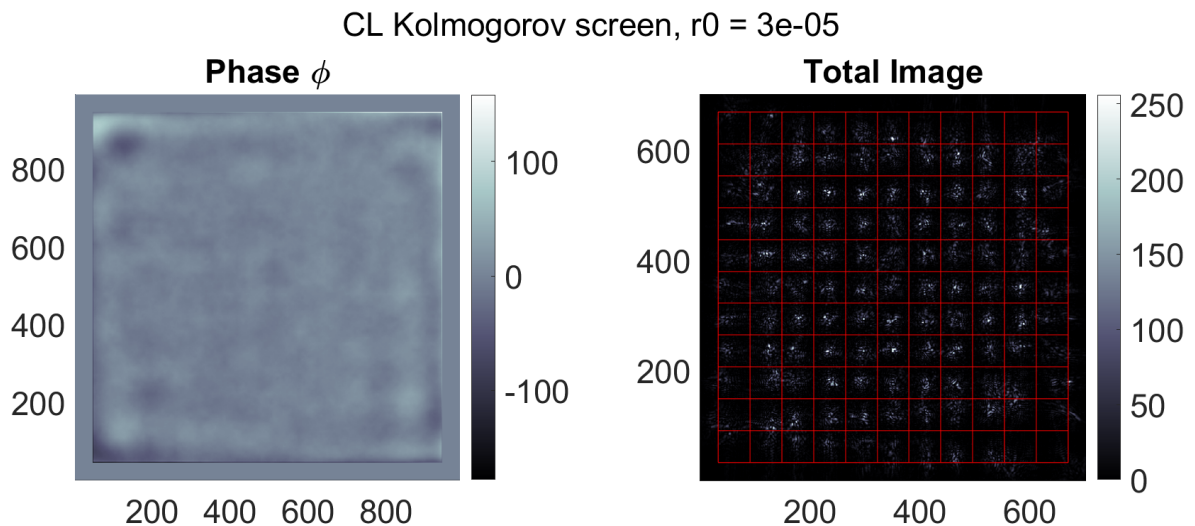


Figure 5-48: Iterative SH sensor simulation of a Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m]. The input phase aberration (phase residual of iteration 4) is shown on the left, the final SH sensor image on the right. Iteration 6 out of 6.

The same aberration simulated for the SH sensor is provided to the P4F sensor as well, using the same iterative scheme. The results are presented in Figure 5-49 up to Figure 5-54.

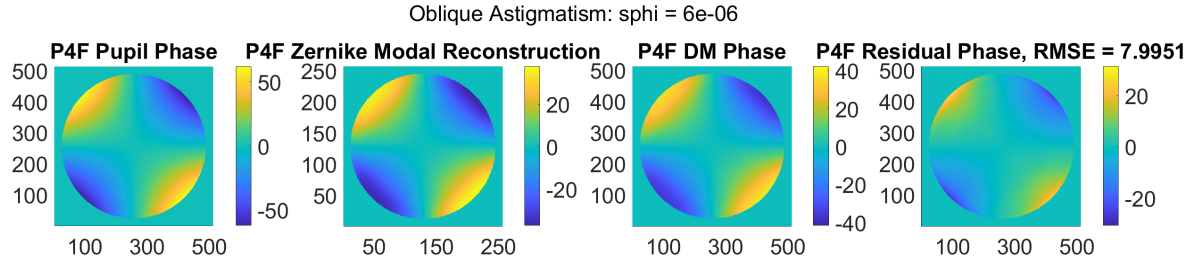


Figure 5-49: Isoplanatic iterative phase reconstruction for the P4F sensor with a Zernike Z_2^2 aberration and $6 \cdot 10^{-6}$ [m] magnitude defined in the aperture plane. Iteration 1 out of 6.

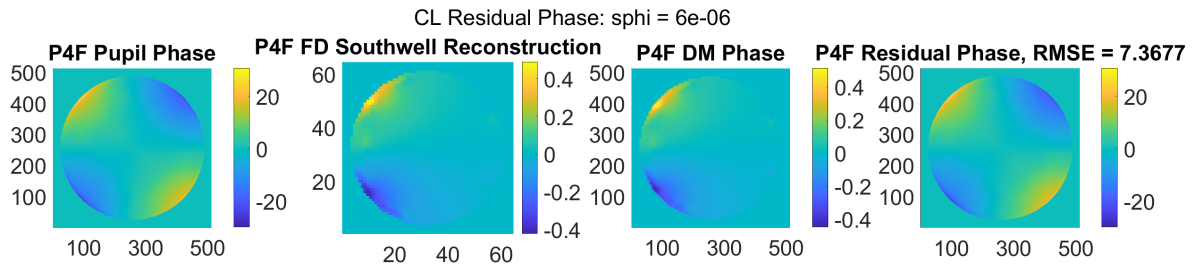


Figure 5-50: Isoplanatic iterative phase reconstruction for the P4F sensor with a Zernike Z_2^2 aberration and $6 \cdot 10^{-6}$ [m] magnitude defined in the aperture plane. Iteration 6 out of 6.

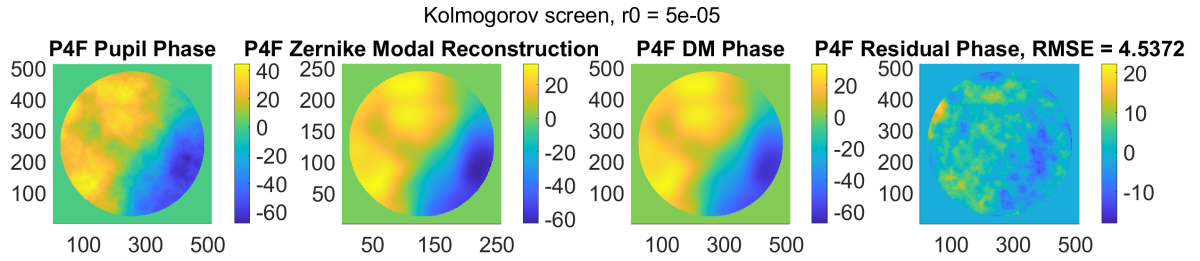


Figure 5-51: Isoplanatic iterative phase reconstruction for the P4F sensor with a Kolmogorov screen and $r_0 = 5 \cdot 10^{-5}$ [m] defined in the aperture plane. Iteration 1 out of 6.

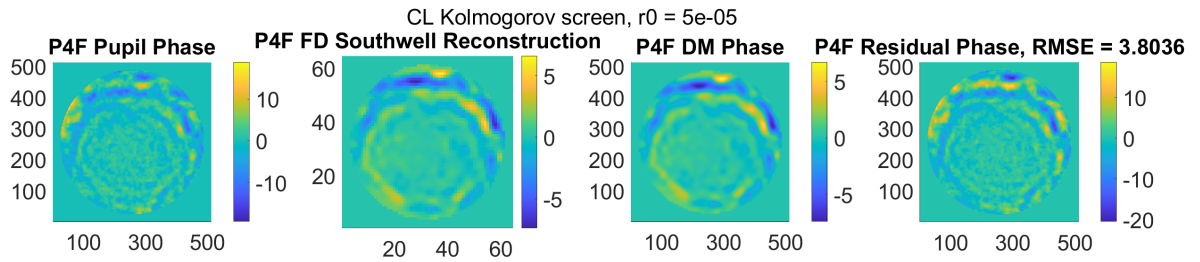


Figure 5-52: Isoplanatic iterative phase reconstruction for the P4F sensor with a Kolmogorov screen and $r_0 = 5 \cdot 10^{-5}$ [m] defined in the aperture plane. Iteration 6 out of 6.

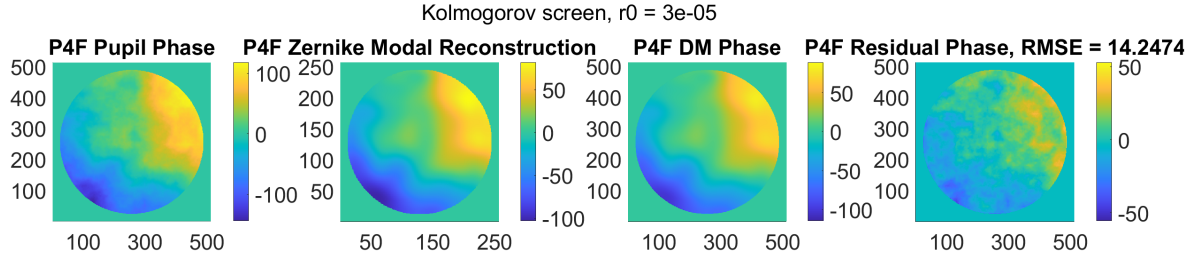


Figure 5-53: Isoplanatic iterative phase reconstruction for the P4F sensor with a Kolmogorov screen and $r_0 = 3 \cdot 10^{-5}$ [m] defined in the aperture plane. Iteration 1 out of 6.

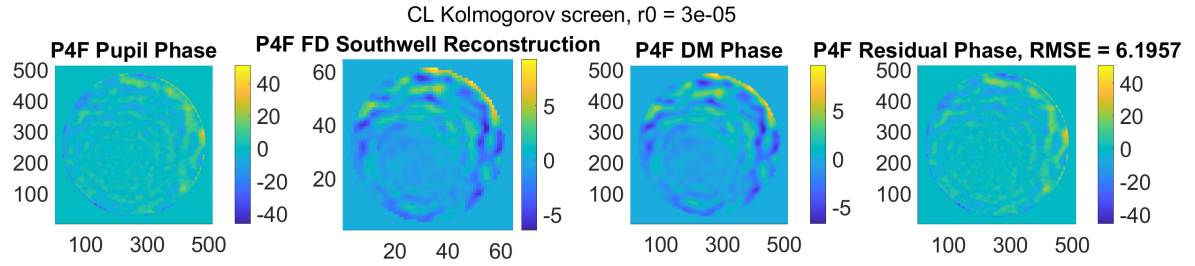


Figure 5-54: Isoplanatic iterative phase reconstruction for the P4F sensor with a CL Kolmogorov screen and $r_0 = 3 \cdot 10^{-5}$ [m] defined in the aperture plane. Iteration 6 out of 6.

The results of the iterative P4F reconstruction algorithm are the same as previous results presented for the SH sensor: The iterative algorithm offers almost no improvement for reconstructing Zernike modes, but does significantly increase the performance for Kolmogorov screens. For stronger Kolmogorov screens it is observed that both the initial reconstruction error is larger, but also the iterative improvement to the RMSE.

Despite these similarities the P4F sensor does show significantly better reconstruction results for the same input aberrations, an improvement that can be attributed to the following differences between the sensors:

1. Gradient sample size: Using the CAFADIS modes P4F sensor retrieves a 50×50 grid of phase gradients, whereas the SH sensor is limited to 11×11 grids for the gradients.
2. Imaged aberration: In this comparison both P4F and SH aperture plane sizes are kept equal for fair comparison, but the shapes of the aperture masks are not equal. The P4F only images the central part of the aberration and blocks the corners, resulting in a weaker sensed and reconstructed phase.
3. Sub-images: The P4F MLA performs angular sampling and thus enables better separation of high- and low-frequency content, compared to the SH sensor. This benefit is at the cost of sensitivity.

As with the observed Hartmannogram one can take a closer look at the P4F sensor images of the last Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m]. The input phases and resulting sensor images are shown in Figure 5-55 and Figure 5-56. It can be readily observed that scattering is also present in these images, due to the higher-order oscillations. As opposed to

the image first moment calculations, the CAFADIS slope model is better suited to manage this dispersion. On the other hand, the final iteration clearly shows the insufficient sensitivity of the P4F sensor, inherent to its design. During iterative phase correction the dominant phase values are reduced, pushing pixels in the plenoptic image to the centremost lenslets. If the phase residual is weak enough all light is captured by the centremost lenslet, returning phase gradients equal to 0.

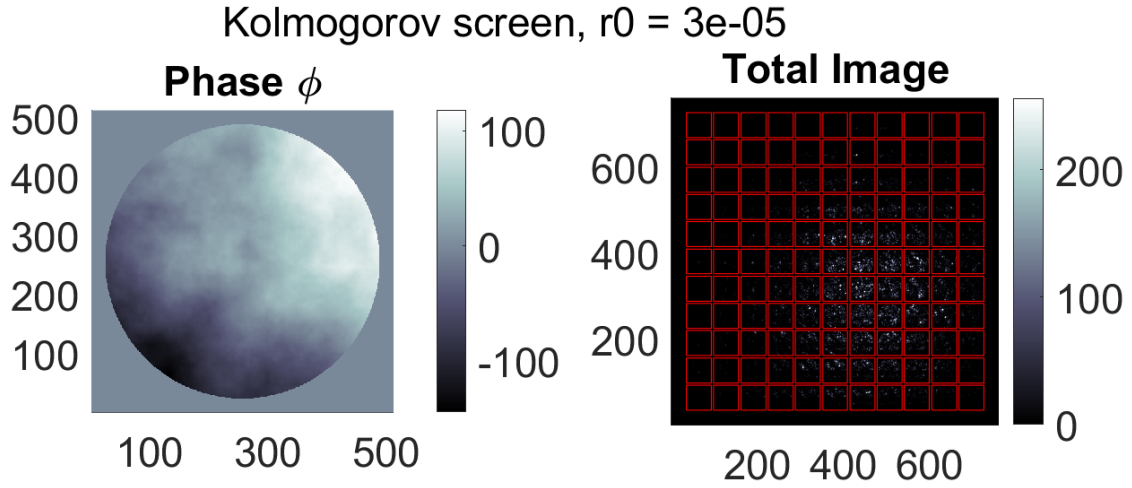


Figure 5-55: Iterative SH sensor simulation of a Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m]. The input phase aberration is shown on the left, the final SH sensor image on the right. Iteration 1 out of 6.

This behaviour is recognized in the final iteration, where most pixels in the P4F image are captured by the 3×3 sub-image grid in the centre, although the pixels in the centremost lenslet are the brightest. This results in smaller gradients to be sensed and thus a weaker phase reconstruction to be returned.

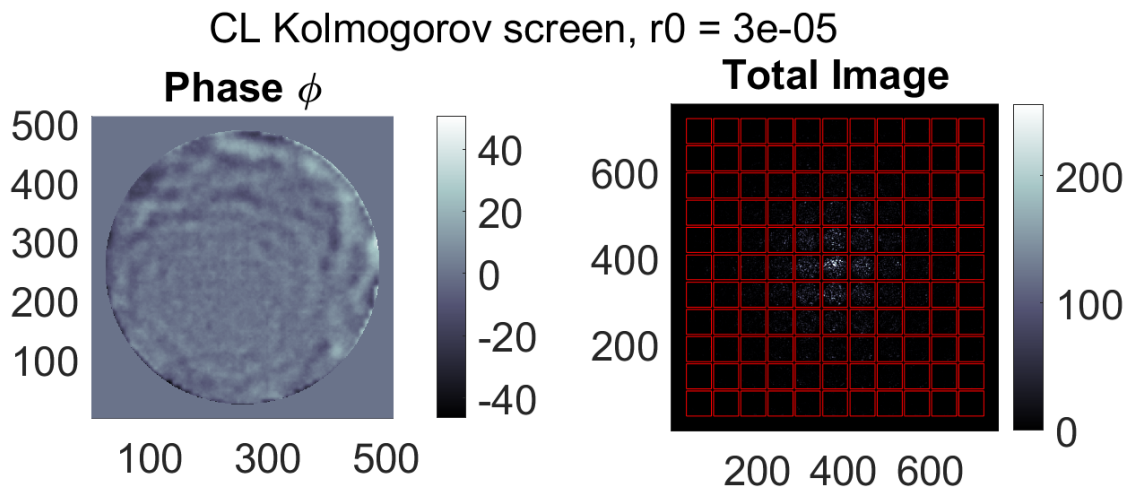


Figure 5-56: Iterative SH sensor simulation of a Kolmogorov screen, generated with $r_0 = 3 \cdot 10^{-5}$ [m]. The input phase aberration (phase residual of iteration 4) is shown on the left, the final SH sensor image on the right. Iteration 5 out of 6.

5-4-4 Conclusion on Isoplanatic Results

The isoplanatic reconstruction and correction results have demonstrated how the developed simulation toolbox can correct for phase aberrations in the aperture plane. Both zonal and modal algorithms succeeded in reconstructing the overall aberration profile, but failed at correctly capturing all phase values.

The comparison between modal and zonal reconstruction results lead to the conclusion that both perform alike, given enough gradient measurements are available. If the number of sensed gradients is scarce, the modal algorithm is better in removing dominant phase patches. In the SH sensor this is especially beneficial, as its number of sensed gradients are limited by the number of lenslets M^2 .

The single-step phase reconstruction is incapable of completely capturing the phase aberrations, leading to the implementation of an iterative reconstruction and correction algorithm. For simulated Zernike aberrations this iterative scheme does not improve results significantly. It does perform better in the simulation of Kolmogorov screens, where the RMSE of the phase residual is clearly reduced after 6 correction steps. Despite significant improvements the iterative algorithm does not converge but stops at the maximum number of iterations. As explained in Section 5-4-3, the stopping criteria are determined with emphasis on clear improvements without too many iterations, instead of performing perfect correction.

Finally, the presented results show that the plenoptic 4F outperforms the Shack-Hartmann sensor with the developed toolbox. It must be noted that this comparison is not conducted on equal grounds, as the number of P4F phase gradients surpasses that of the SH sensor, giving the P4F sensor the upper hand. Section 5-5 compares the sensors for isoplanatic aberrations once more, by considering more fair grounds. Section 5-6 investigates the performance of the anisoplanatic reconstruction algorithm.

5-5 Comparing the Shack-Hartmann and Plenoptic 4F Sensor

The isoplanatic results presented in Section 5-4 have shown the performance of the developed simulation toolbox for different types of simulations. This section uses the conclusions of Section 5-4 to appropriately choose the experimental conditions, such that a comparison on equal grounds between the SH and P4F sensors is conducted. Such a comparison should not favour either sensor but highlight each of their qualities. This means that the number of obtained gradient measurements, the gradient sample count, should be kept equal.

In order to answer the first research question posed in Chapter 1 the performance of both sensors should be compared as clearly as possible. The chosen comparison is thus conducted in an isoplanatic setting without iterative correction. The iterative scheme improves the correction result, but does not favour either of the two sensors and as such does not benefit the comparison. The phase reconstruction is conducted by the modal algorithm. The phase screens will be generated using Kolmogorov screens for varying Fried parameters r_0 . The parameters used in these simulations are listed in Table 5-5.

Previous isoplanatic simulations demonstrated the reconstruction results for varying input conditions, such as different Zernike modes or Kolmogorov screens. These results have led to interesting and clear conclusions on the developed algorithms themselves. For additional

comparisons between both sensors, it is interesting how the phase reconstructions change when design parameters are changed.

This comparison will be performed for three different Kolmogorov screens of varying strengths. Each screen is imaged by the SH and P4F sensors for two different microlens arrays. Each scenario is generated with the same aperture plane size, such that the same Kolmogorov screen is imaged for different MLAs. This also means that the smaller array only images the central part of the aberrations, which is reflected by the zero-padding present in the images.

Finally, the aperture plane sizes between SH and P4F sensor are kept equal. This was mentioned above but additional emphasis is needed, as it means that the objective lens parameters of the P4F sensor are adjusted to fit different MLA sizes. Looking back at the performance metrics defined for the plenoptic sensor in Section 3-3-4 this reduces the dynamic range and but may improve the sensitivity.

The first Kolmogorov screen is generated with $r_0 = 8 \cdot 10^{-5}$ [m] and can be considered a weak aberration, in comparison to previously employed screens. The sensors in the first scenario use an 11×11 MLA and are referred to as SH 1 and P4F 1 respectively. Their results are shown in Figure 5-57 and Figure 5-58, employing the same image structure as presented in Section 5-4.

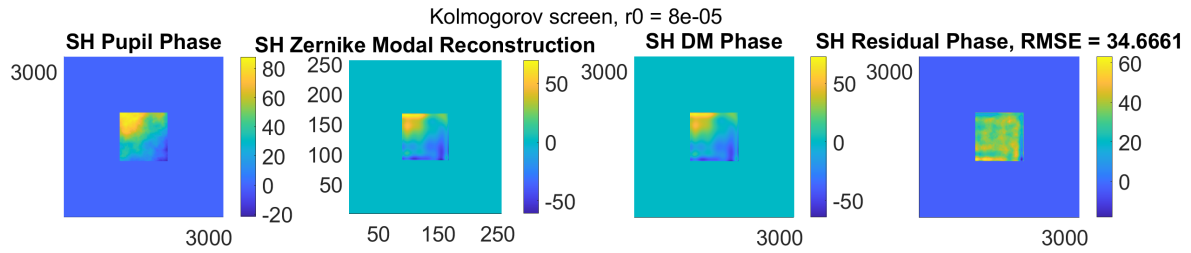


Figure 5-57: Isoplanatic phase reconstruction of the SH 1 sensor with an 11×11 MLA for a Kolmogorov screen, generated with $r_0 = 8 \cdot 10^{-5}$ [m].

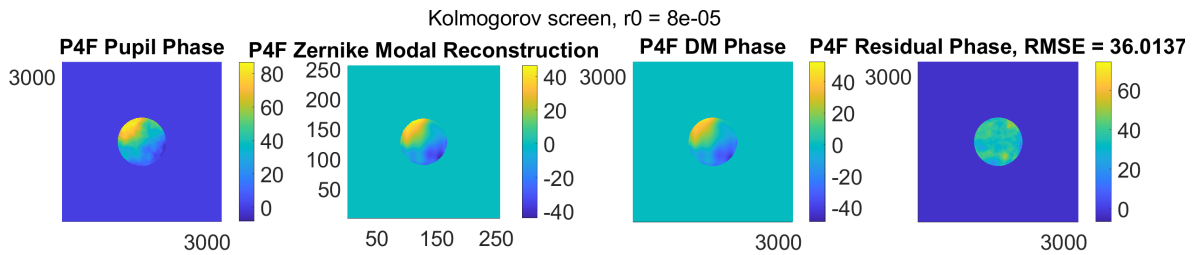


Figure 5-58: Isoplanatic phase reconstruction of the P4F 1 sensor with an 11×11 MLA for a Kolmogorov screen, generated with $r_0 = 8 \cdot 10^{-5}$ [m].

The second scenario uses a 33×33 MLA and the results for corresponding sensors SH 2 and P4F 2 are shown below. The aberration defined in their aperture planes are the fully illuminated versions of the Kolmogorov screen in Figure 5-57 and Figure 5-58.

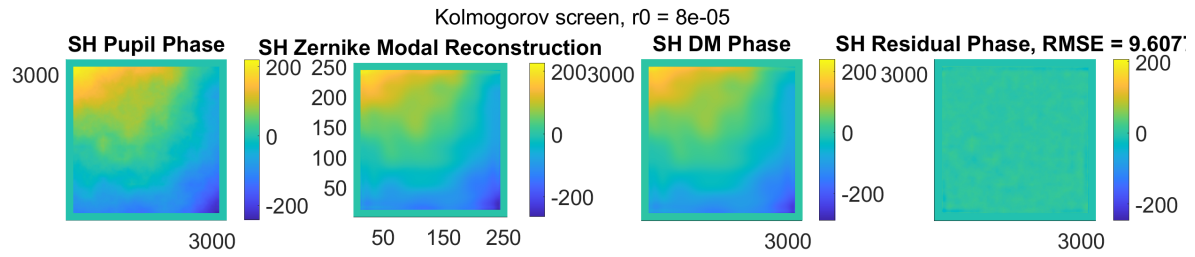


Figure 5-59: Isoplanatic phase reconstruction of the SH 2 sensor with a 33×33 MLA for a Kolmogorov screen, generated with $r_0 = 8 \cdot 10^{-5}$ [m].

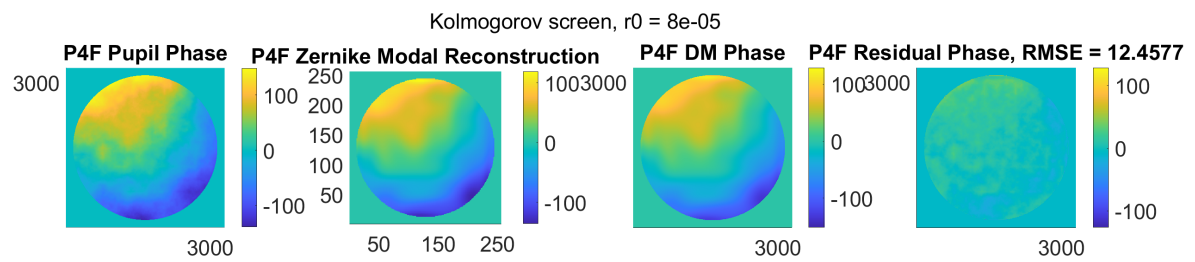


Figure 5-60: Isoplanatic phase reconstruction of the P4F 2 sensor with a 33×33 MLA for a Kolmogorov screen, generated with $r_0 = 8 \cdot 10^{-5}$ [m].

A clear observation is the improvement of the phase correction, where the residual RMSE is reduced by a factor 3 in both sensors. In the SH sensor this is simply due to the availability of more gradient measurements. There is no change in the spatial sampling of the aperture plane, only more spatial sections are imaged. In the P4F sensor it is both the increase in spatial illumination and sensitivity that improves the results. From Table 5-5 it is seen that the dynamic range of the P4F sensor remains approximately the same, while the sensitivity is significantly improved. As explained above, the radius of the objective lens is increased such that both SH 2 and P4F 2 have an equal aperture plane size. Due to the plenoptic equality (3-11) the objective lens focal distance is increased accordingly, as long as the MLA lenslet sizes remain unchanged. This results in an improvement of the P4F 2 sensitivity.

The same scenarios are simulated for a stronger Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m]. The results are shown below, in Figure 5-61 up to Figure 5-64.

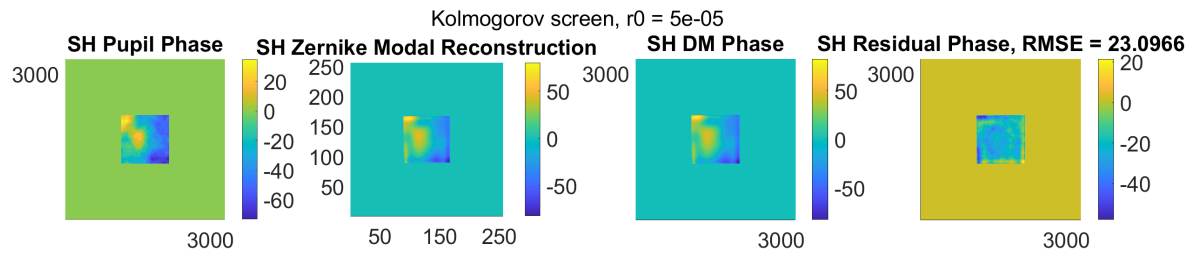


Figure 5-61: Isoplanatic phase reconstruction of the SH 1 sensor with an 11×11 MLA for a Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m].

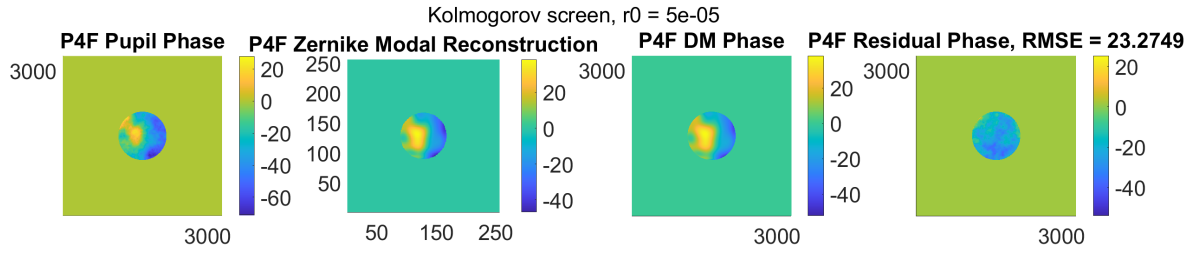


Figure 5-62: Isoplanatic phase reconstruction of the P4F 1 sensor with an 11×11 MLA for a Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m].

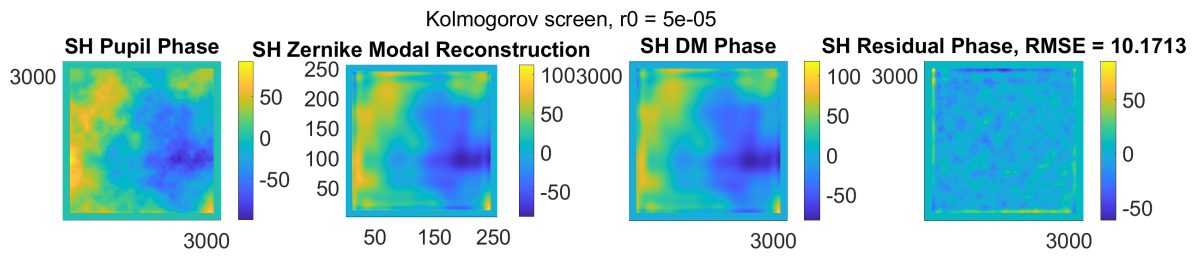


Figure 5-63: Isoplanatic phase reconstruction of the SH 2 sensor with a 33×33 MLA for a Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m].

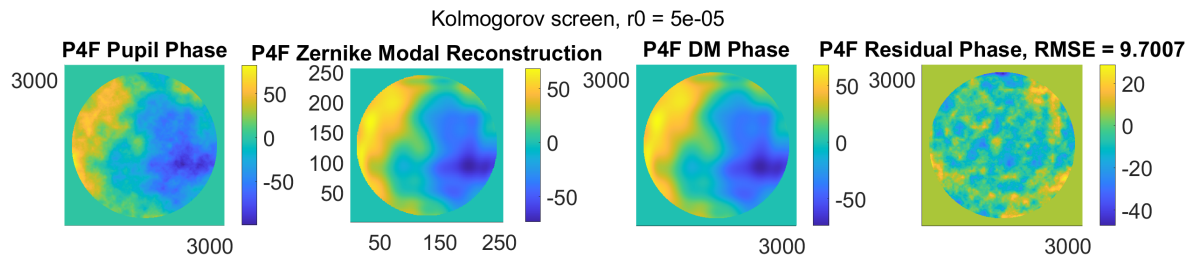


Figure 5-64: Isoplanatic phase reconstruction of the P4F 2 sensor with a 33×33 MLA for a Kolmogorov screen, generated with $r_0 = 5 \cdot 10^{-5}$ [m].

Again, a significant improvement is observed between the two scenarios, where the larger MLA improved correction results by a factor 2. In this experiment the P4F 2 sensor performs better than its Shack-Hartmann counterpart.

A third and final Kolmogorov screen is simulated with Fried parameter $r_0 = 2 \cdot 10^{-5}$ [m], which can be regarded as a strong aberration in the context of these simulations. As such, the SH 2 and P4F 2 are used for phase reconstruction, their results shown in Figure 5-65 and Figure 5-66.

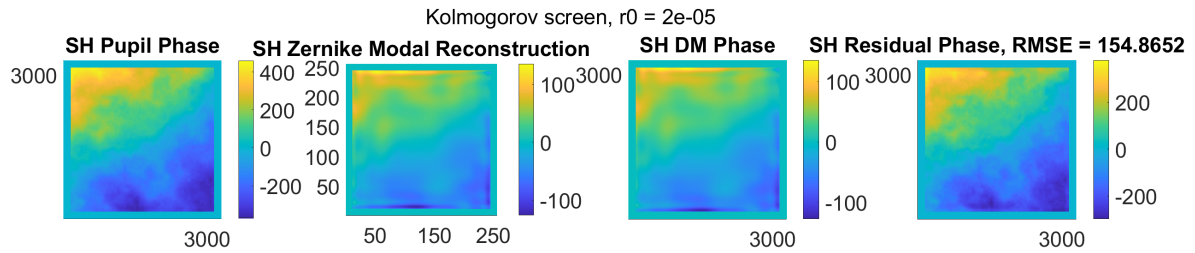


Figure 5-65: Isoplanatic phase reconstruction of the SH 2 sensor with a 33×33 MLA for a Kolmogorov screen, generated with $r_0 = 2 \cdot 10^{-5}$ [m].

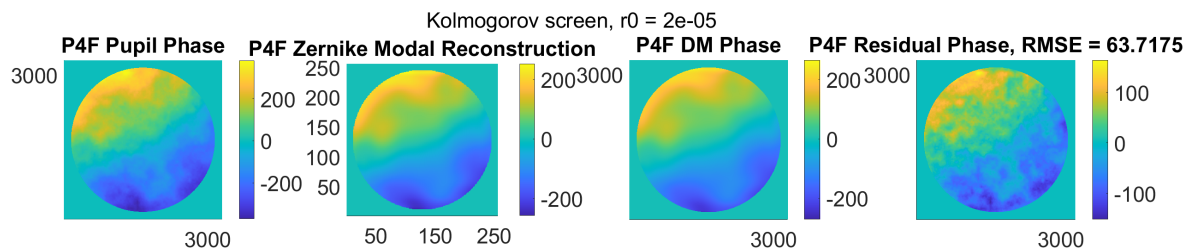


Figure 5-66: Isoplanatic phase reconstruction of the P4F 2 sensor with a 33×33 MLA for a Kolmogorov screen, generated with $r_0 = 2 \cdot 10^{-5}$ [m].

As expected, the stronger screen is devastating to the phase reconstructions. From the SH 2 reconstruction it is clear that the SH sensor lacks the dynamic range to fully sense the aberrations. P4F 2 is less hindered by this problem since the stronger corners of the Kolmogorov screen are blocked by the objective lens. Still, this example presents the opportunity for a third P4F design. This P4F 3 uses an objective lens twice as small increasing its dynamic range at the cost of sensitivity. The results are shown below.

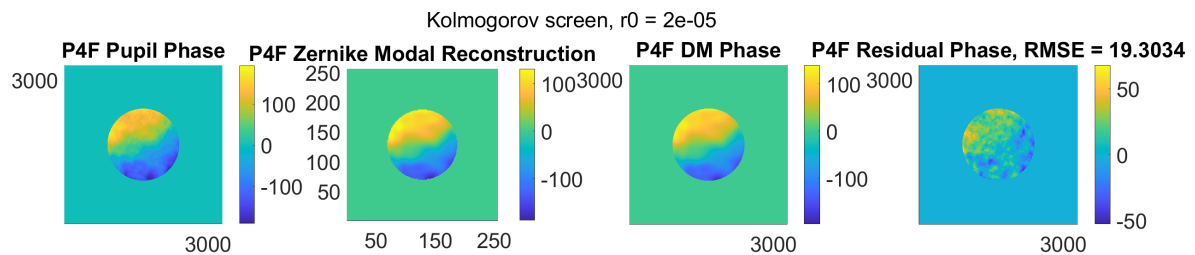


Figure 5-67: Isoplanatic phase reconstruction of the P4F 3 sensor with a 33×33 MLA for a Kolmogorov screen, generated with $r_0 = 2 \cdot 10^{-5}$ [m].

It should be noted that the smaller objective lens passes less light and thus the sensed aberrations are weaker. Nevertheless, the increase on dynamic range allows one to image even stronger aberrations. This can be observed from the obtained sensor images for this Kolmogorov screen, shown in Figure 5-68. The leftmost image shows the Hartmannogram obtained from the stronger screen. The P4F 2 image is shown in the middle, with the P4F 3 image on the right.

The SH sensor image shows a strong dispersion of the pixels over the sub-images. This spreading is detrimental for the image first moment calculation, as reflected by Figure 5-65.

The image of the P4F 2 sensor shows that many sub-images are filled with the scattered pixels, indicating that the dynamic range may not be sufficient for stronger aberrations. The image of the P4F 3 sensor shows exactly the necessary improvement to the dynamic range. Although a smaller section of the Kolmogorov screen is imaged, it is clear that even stronger aberrations can be imaged as the dynamic range is sufficient. In comparison to SH 2 it should be noted that the sensitivity is increased sixfold, which means that SH 2 is still better at resolving the weaker phase details.

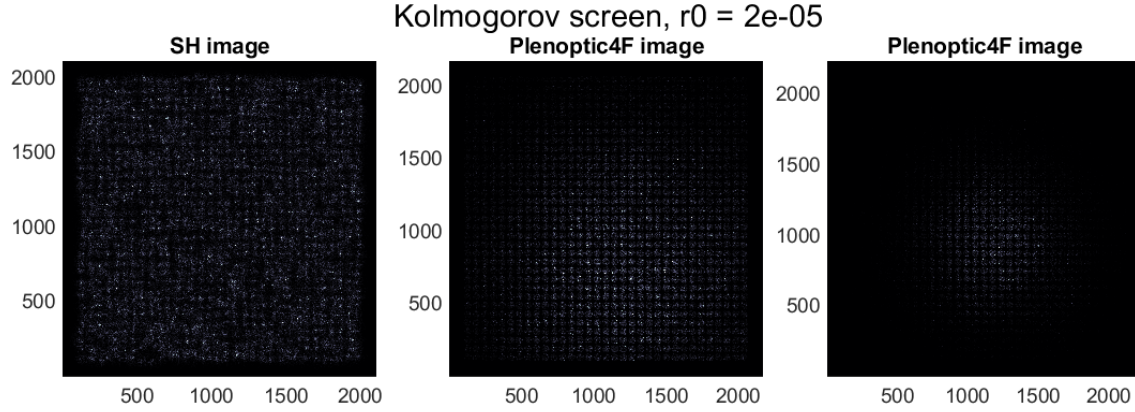


Figure 5-68: Obtained sensor images by simulating the SH 2 (left), P4F 2 (middle) and P4F 3 (right) sensors for a Kolmogorov screen generated with $r_0 = 2 \cdot 10^{-5}$ [m]. This screen is placed in the aperture plane of each sensor. The microlens contains 33×33 lenslets.

Parameter description	SH 1	P4F 1	SH 2	P4F 2	P4F 3
Objective lens pitch $2r_{obj}$	-	$1650\mu\text{m}$	-	$4950\mu\text{m}$	$2475\mu\text{m}$
Objective lens focal distance f_{obj}	-	100mm	-	320mm	155mm
MLA lenslet pitch $2r_{mla}$	$150\mu\text{m}$		$150\mu\text{m}$		
MLA lenslet focal distance f_{mla}	10mm		10mm		
MLA lenslet grid	11×11		33×33		
Image sensor pixel size d_{pix}	$5.2\mu\text{m}$		$5.2\mu\text{m}$		
Wavelength of light λ	500nm		500nm		
Reconstructed Zernike modes	60		200		
Performance metric					
Dynamic Range	0.015	0.0136	0.015	0.0145	0.0291
Sensitivity	$3 \cdot 10^{-4}$	0.0027	$3 \cdot 10^{-4}$	$9 \cdot 10^{-4}$	0.0018
Gradient sample size	121	121	1225	1225	1225

Table 5-5: Simulation parameters and performance metrics of the comparison conducted between the Shack-Hartmann and plenoptic 4F sensors.

Conclusions on the Comparisons The isoplanatic simulations conducted above support the conclusions drawn from Chapter 3, where the SH sensor can still outperform the P4F sensor,

based on the performance metrics. It is possible to improve the P4F performance, by for example adding more lenslets to the MLA or reducing the objective lens size, but these adjustments always results in trade-offs. In the example of a larger MLA one can choose to improve the plenoptic sensitivity, by keeping the aperture planes between the sensors equal. This was done for P4F 2, but resulted in a dynamic range inferior to that of SH 2.

On the other hand one can increase the plenoptic dynamic range by keeping the size of the objective lens fixed and adding more lenslets to the microlens array. This was done for P4F 3 and significantly improved the dynamic range. The resulting trade-off is that the section of the Kolmogorov screen imaged by P4F 3 is smaller than for SH 2. This trade-off leads to larger differences between the image phases of SH 2 and P4F 3, making a comparison less meaningful.

As a final statement, it can be concluded that the performance of the plenoptic sensor depends on both the microlens array design freedom and aberration strengths. It is clear that the stronger Kolmogorov screens, associated with $r_0 = 5 \cdot 10^{-5}$ [m] and $2 \cdot 10^{-5}$ [m] respectively, highlight the limited dynamic range of the SH sensor. The plenoptic WFS is able to reconstruct these strong aberrations better. On the other hand, the SH sensor will outperform the P4F WFS for weaker aberrations, for example shown in Figure 5-59 and Figure 5-60.

The last simulations performed with the toolbox are the reconstructions of anisoplanatic aberrations. Without considering the plenoptic Ray Tracing algorithm, anisoplanatic phase screen retrievals depend on the quality of the isoplanatic reconstructions. As such, the anisoplanatic simulations should show how the screens are retrieved and which method of retrieval performs best. The conclusions drawn on the anisoplanatic reconstructions are focussed on the different retrieval methods, not on differences between the SH and P4F reconstruction.

5-6 Anisoplanatic Simulation Results

Simulating anisoplanatic aberrations is interpreted by the simulation toolbox in shifting the phase screen between the object and pupil planes. The governing parameter is the distance ratio RL , which is equal to 1 for isoplanatic aberrations and a phase screen inside the pupil plane. For any value $RL \in (0, 1]$ the phase screen is placed at a distance $(1 - RL)$ from the WFS pupil plane.

The results presented in this section are focussed on the phase retrieval algorithm presented in Section 5-2-8, in order to answer the second research question posed in Chapter 1. As such WFS sensor images and pupil plane reconstructions will not be presented, but only the simulated phase screen and retrieved results.

The need for an anisoplanatic model, as elaborated in Section 4-2, arises from atmospheric turbulence introduced outside the pupil plane. In reality one cannot isolate one point source from the imaging process and multiple aberrations are defined in the pupil plane, as long as emitted light is considered to be incoherent. Additionally, performing anisoplanatic reconstruction for a single source does not illustrate the full functionality of the derived algorithm. Throughout these simulations it is assumed that each point source emits incoherent light.

As a final remark, the phase screen in these simulations is generated using Zernike modes. From the isoplanatic imaging of multiple Kolmogorov screens, shown in Section 5-4-3 and Section 5-5, it is clear that higher-order oscillations scatter pixels in each sub-image. If multiple

sources are imaged the sensor image contains the superposition of these dispersions, one for each point source. This will be impossible to process for the MPI gradient retrieval algorithms elaborated in Section 5-2-6, resulting in bad reconstructions.

As explained in Section 5-2-8 the phase screen is retrieved by either back-propagation of the pupil plane reconstructions and estimation of the hidden piston modes, or back-propagation of the phase gradients and a local reconstruction of the phase. Throughout the anisoplanatic results these methods are referred to as 'Retrieval 1' and 'Retrieval 2', respectively. The object plane contains 5 simulated point sources located at angular coordinates $(\pm 0.01, \pm 0.01)$ [rad] and $(0, 0)$ [rad]. This means one source lies on the optical axis and the others in the corners of the object plane. To observe the algorithm performance both methods are compared for a flat phase screen of $\phi_{screen} = 0$ with $RL = 0.9$, shown in Figure 5-69. The retrieved phase screens for both the SH and P4F sensor are shown in Figure 5-70 and Figure 5-71, where the local reconstruction of 'Retrieval 2' is performed by the zonal algorithm.

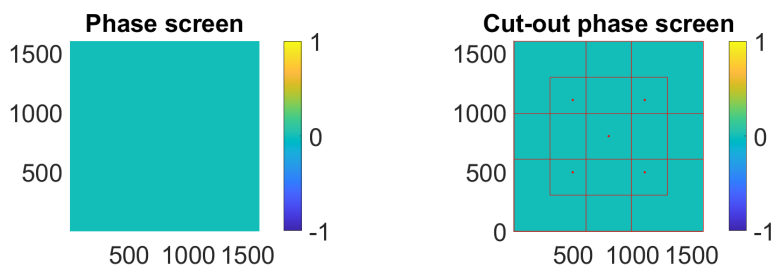


Figure 5-69: The simulated phase screen (left) and the selected phase screen cut-outs (right). Each red dot is the centre of a phase screen cut-out, the area of said cut-out outlined in the same colour. The width of each cut-out is approximately $3/5$ of the phase screen width. The simulated point sources are placed at angular coordinates $(\pm 0.01, \pm 0.01)$ [rad] and $(0, 0)$ [rad].

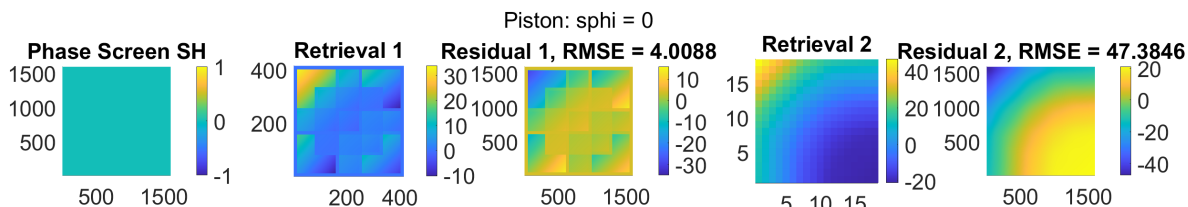


Figure 5-70: Phase screen retrieval using anisoplanatic aberration reconstruction with $RL = 0.9$ for the SH sensor. The phase screen is a piston aberration and the phase screen cut-outs are shown in Figure 5-69. The screen is retrieved using either back-propagation of reconstructed phases or local phase screen reconstruction, referred to as 'Retrieval 1' and 'Retrieval 2' respectively.

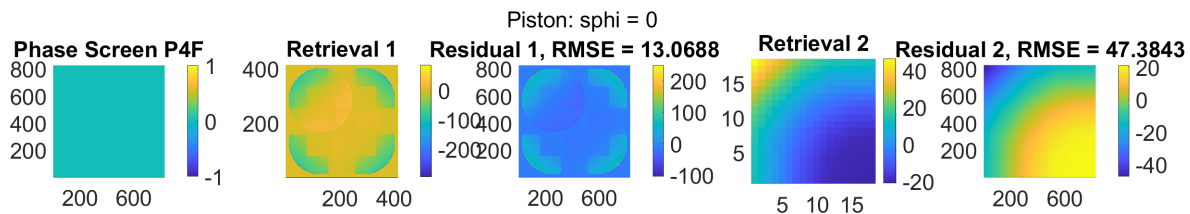


Figure 5-71: Phase screen retrieval using anisoplanatic aberration reconstruction with $RL = 0.9$ for the P4F sensor. The phase screen is a piston aberration and the phase screen cut-outs are shown in Figure 5-69. The screen is retrieved either by back-propagation of reconstructed phases or local phase screen reconstruction, referred to as 'Retrieval 1' and 'Retrieval 2' respectively.

The zonal algorithm is chosen for 'Retrieval 2', as more phase gradients are available in the phase screen compared to the pupil plane, thus increasing the performance of the zonal reconstruction. The objects have been chosen such that the central part of the phase screen is illuminated by multiple sources, providing more overlapping measurements. It should be noted that each imaged pupil plane phase consists of the projected phase screen plus object-based tilt phase, such that the angular position of the source shows up in the image. This OBT is removed from the pupil plane reconstruction before back-propagation.

From Figure 5-70 and Figure 5-71 one could conclude that 'Retrieval 1' is better than a local phase screen reconstruction. In this example the back-propagated phases are purely OBT tilt aberrations that are removed before the phase screen retrieval. From Section 5-4 it was clear that isoplanatic phase reconstruction could not fully correct for phase aberrations. In the example of Figure 5-70 and Figure 5-71 this means that there is a difference between the original object-based tilt added to each pupil plane - in *getpupil.m* - and reconstructed OBT that is removed before back-propagation. This is reflected in the images of 'Retrieval 1' by tilt-blocks in the retrieved phase screen, where the phase screen cut-outs are placed.

The same problem is present in 'Retrieval 2', where phase gradients are propagated back to the phase screen, averaged at overlapping pixels and used to locally reconstruct the phase screen. The only reason why 'Retrieval 1' performs better is due to the reconstruction error present in the data. This error in the reconstructed pupil planes is in the order of 10. In the back-propagated phase gradients this error is in the order 10^5 , since the gradients themselves are in the order of 10^6 . This means that 'Retrieval 2' is more sensitive to errors present in the data.

In the next experiment a defocus aberration is generated in the phase screen with magnitude $6 \cdot 10^{-6}$ [m] for $RL = 0.6$. This means that the phase screen is placed further from the pupil plane and the point sources should be placed closer to the optical axis, to ensure sufficient cut-out overlap. The sources are now located at angular coordinates $(\pm 0.004, \pm 0.004)$ [rad] and $(0, 0)$ [rad]. The results of these new simulation parameters are shown below.

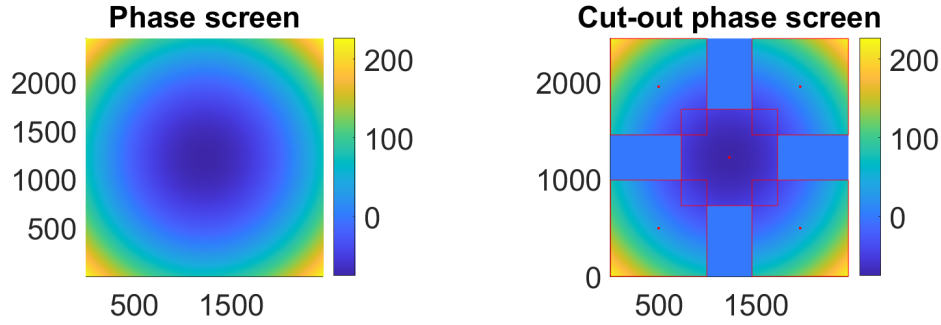


Figure 5-72: The simulated phase screen (left) and the selected phase screen cut-outs (right). Each red dot is the centre of a phase screen cut-out, the area of said cut-out outlined in the same colour. The width of each cut-out is approximately $3/5$ of the phase screen width. The simulated point sources are placed at angular coordinates $(\pm 0.004, \pm 0.004)$ [rad] and $(0, 0)$ [rad].

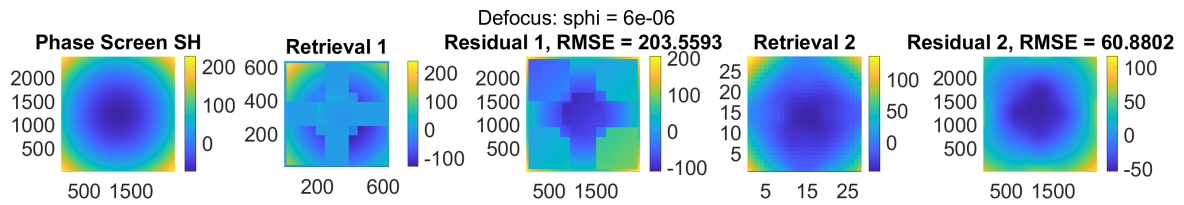


Figure 5-73: Phase screen retrieval using anisoplanatic aberration reconstruction with $RL = 0.6$ for the SH sensor. A Zernike Z_2^0 aberration with magnitude $6 \cdot 10^{-6}$ [m] is generated in the phase screen. The phase screen cut-outs are shown in Figure 5-72. The screen is retrieved either by back-propagation of reconstructed phases or local phase screen reconstruction, referred to as 'Retrieval 1' and 'Retrieval 2' respectively.

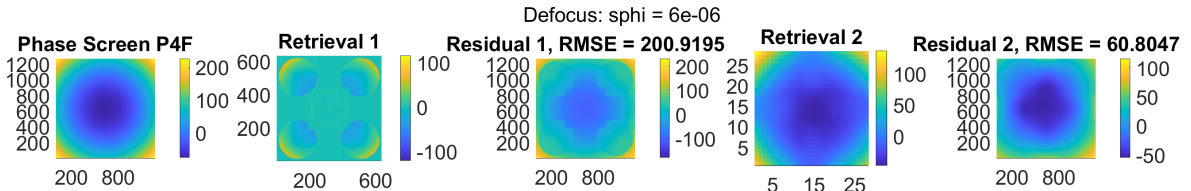


Figure 5-74: Phase screen retrieval using anisoplanatic aberration reconstruction with $RL = 0.6$ for the P4F sensor. A Zernike Z_2^0 aberration with magnitude $6 \cdot 10^{-6}$ [m] is generated in the phase screen. The phase screen cut-outs are shown in Figure 5-72. The screen is retrieved either by back-propagation of reconstructed phases or local phase screen reconstruction, referred to as 'Retrieval 1' and 'Retrieval 2' respectively.

With the addition of the defocus aberration 'Retrieval 2' performs significantly better. It is able to reconstruct the screen as a much smoother surface and offers an improved correction over 'Retrieval 1'. It should be noted that it does not perfectly correct for the phase screen. In the back-propagated gradients the aberration reconstruction errors are present, as well as the insufficient estimation of the OBT terms. This means that the reconstruction error returns in the phase screen retrieval twofold. Nevertheless, it performs better than 'Retrieval 1'.

A final experiment is conducted for a different phase screen, while keeping the point source coordinates and distance ratio RL the same. The phase screen is generated by a superposition

of Zernike modes Z_2^2 , Z_2^0 and Z_3^{-3} with magnitudes $\begin{bmatrix} 3 & 1 & 2 \end{bmatrix} \cdot 10^{-6}$ [m] respectively. The results of both retrieved phase screens are shown below, in Figure 5-75 for the SH sensor and in Figure 5-76 for the P4F sensor.

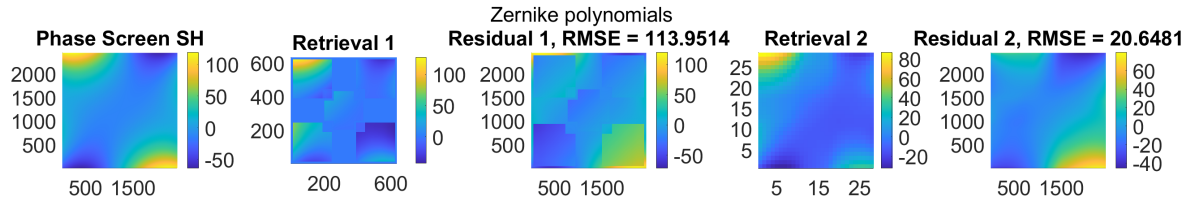


Figure 5-75: Phase screen retrieval using anisoplanatic aberration reconstruction with $RL = 0.6$ for the SH sensor. The phase screen is generated with a superposition of Zernike modes Z_2^2 , Z_2^0 and Z_3^{-3} , with magnitudes $3 \cdot 10^{-6}$, $1 \cdot 10^{-6}$ and $2 \cdot 10^{-6}$ respectively, all expressed in [m]. The phase screen cut-outs are the same as shown in Figure 5-72, albeit for a different phase screen. The screen is retrieved either by back-propagation of reconstructed phases or local phase screen reconstruction, referred to as 'Retrieval 1' and 'Retrieval 2' respectively.

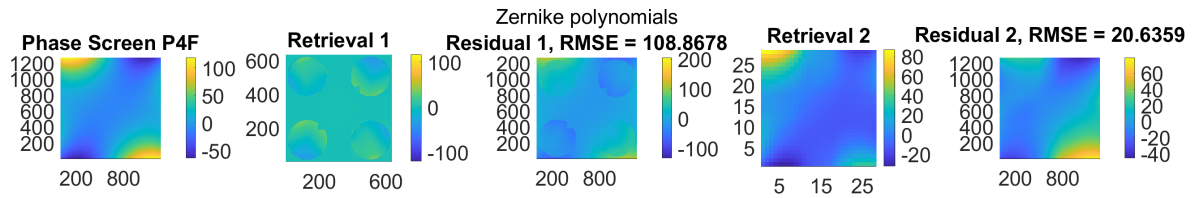


Figure 5-76: Phase screen retrieval using anisoplanatic aberration reconstruction with $RL = 0.6$ for the P4F sensor. The phase screen is generated with a superposition of Zernike modes Z_2^2 , Z_2^0 and Z_3^{-3} , with magnitudes $3 \cdot 10^{-6}$, $1 \cdot 10^{-6}$ and $2 \cdot 10^{-6}$ respectively, all expressed in [m]. The phase screen cut-outs are the same as shown in Figure 5-72, albeit for a different phase screen. The screen is retrieved either by back-propagation of reconstructed phases or local phase screen reconstruction, referred to as 'Retrieval 1' and 'Retrieval 2' respectively.

These phase screen retrievals confirm previous results, where 'Retrieval 1' performs best when the phase screen consists of piston and tilt phases. When aberrations are added to the phase screen 'Retrieval 2' shows better results.

It should be noted that the defined phase screen aberrations should be weaker than the object-based tilt terms enforced by the angular coordinates of the sources. This weakness in the toolbox is noted in Section 5-2-6 as well, where phase gradients are incorrectly assigned to imaged point sources if the aberrations are too strong. In the multi-point source imaging algorithms, pixels in the sub-images are assigned to their corresponding object based on the expected pattern. This pattern is determined by the aberration-free image of each point source, which is related to the OBT phase terms. The algorithm attempts to fit the pixels in the sub-image to this expected pattern. When the best fit is determined the pixels are assigned to the closest expected location and corresponding point source.

If the defined phase aberrations are stronger than the object-based tilts, the recorded sub-image may differ significantly from the expected pattern. In an attempt to find the best fit, pixels may be assigned incorrectly to point sources, resulting gradients being mixed between simulated objects. Without further possibility to correctly separate the phase gradients an ill-conditioned phase reconstruction is the only result.

Conclusion on Anisoplanatic Aberrations Which approach of phase screen retrieval is best depends on two conditions: 1) the illumination of the phase screen and 2) the dominant aberrations in the phase screen. In most realistic scenarios it can be assumed that the phase screen consists of modes other than piston and tilt. If this assumption holds true the retrieval performance rests on the phase screen illumination.

If the number of simulated objects is scarce, or the phase screen cut-outs only fill a small region of the screen, one should choose 'Retrieval 1'. This approach propagates the reconstructed pupil plane phases back to the phase screen and finds the optimal piston correction minimizing the variance between overlapping pixels. The benefit of this method is that it only considers known pixels, which are pixels in the phase screen whose phase- or gradient-value is known. The drawback of this approach is that the retrieved phase screen is discontinuous.

If the phase screen is sufficiently illuminated, most of the phase- and gradient-values in the pixel are determined. In this case it is beneficial to average the gradients at overlapping pixels and reconstruct the phase screen using a zonal or modal algorithm. The unknown pixels are filled with zeroes, but they do not influence the reconstruction result significantly, since they are outnumbered by the known gradients. This approach results in a much smoother retrieval of the screen and is preferred in most cases.

When comparing the SH and P4F sensors, there is little difference in retrieval performances between them. It should be repeated that the retrieval algorithm is sensitive to measurement noise, as it is returned twofold in the reconstructions. Reconstruction errors inherent in the isoplanatic algorithms are carried over to the retrieved phase screen, as well as mismatches in removing the object-based tilt phases. In general it can be concluded that a large gradients sample count is beneficial to the retrieval process.

With all simulation aspects performed and all reconstruction results concluded on, the final conclusions of this thesis are presented in Chapter 6.

Chapter 6

Conclusion

This chapter collects important results obtained in previous chapters and forms final conclusions for this thesis. These conclusions are followed by future recommendations and possible improvements for the author's work. Chapter 1 introduced the two research questions that determined the direction of this thesis. The conclusion presented in this chapter is aimed at answering both questions.

This thesis is separated into five parts. Chapter 1 and Chapter 2 introduce the problem tackled in this thesis and establish the basic notions in Adaptive Optics (AO). Chapter 3 presents the literature study focussed at the working principles of the Shack-Hartmann (SH) and plenoptic sensors. The second part of the literature study, that concentrates on atmospheric turbulence and anisoplanatic aberrations, is detailed in Chapter 4. The major contribution of the author is presented in Chapter 5, which is the development of the simulation toolbox. The final part is the conclusion of this thesis, presented in this chapter.

The conclusions drawn in Chapter 3 allows one to answer the first research question, which will be repeated here:

How does the plenoptic sensor improve modern wavefront sensing and will it provide a better alternative to the Shack-Hartmann sensor?

Both sensors are subject to a comparative investigation, which was conducted on theoretical performance metrics, as well as practical studies done by other authors.

From these comparisons it was concluded that the plenoptic sensor cannot outperform the SH sensor for a shared microlens array (MLA), based on the maximum and minimum detectable wavefront slopes. The corresponding performance metrics are dynamic range and sensitivity. On the other hand, the plenoptic design facilitates straightforward improvement of these metrics. For example, one can effortlessly increase the dynamic range by adding more lenslets to the MLA. By decreasing the individual lenslet size, one decreases the minimum detectable slope and thus improves the plenoptic sensitivity. This improvement does decrease the dynamic range, presenting a trade-off for the plenoptic sensor between dynamic range and sensitivity. The sensitivity of the plenoptic sensor can be improved by modulating the beam

focussed by the objective lens. It was shown that this modulation significantly enhanced the plenoptic sensitivity, up to the point that was roughly the same as the SH sensor.

In turn, the SH's sensitivity is effortlessly improved by increasing the resolution of the image sensor. If larger lenslets are considered in the MLA the dynamic range is improved, as well as the average slope error. Again, one is presented with a trade-off, now between dynamic range and slope error.

The conclusion presented in Chapter 3 states that the design freedom of the MLA determines how the plenoptic sensor can outperform the SH sensor. If large arrays with small lenslets are available it becomes significantly easier to improve the plenoptic dynamic range and sensitivity. To guide this statement the performance metrics of both sensors are compared to obtain the following inequalities (3-24):

$$\begin{aligned} M &\geq \frac{L}{f_{mla}} + 1, \\ \frac{d_{pix}}{2f_{mla}} &\geq \frac{d_{mla}}{L}, \\ \frac{d_{mla}}{d_{pix}} &\geq \frac{2M}{\sqrt{\pi}}, \end{aligned}$$

where the MLA consists of $M \times M$ lenslets, each with pitch d_{mla} and focal distance f_{mla} . The size of the image sensor pixels is given by d_{pix} . The separation between objective lens and MLA in the plenoptic sensor is given by L . Inequalities (3-24) are determined by comparing the dynamic ranges, sensitivities and gradient sample sizes respectively. Satisfaction of each inequality makes the plenoptic sensor outperform the SH sensor on the corresponding metric.

Beyond performance metrics, the design of the plenoptic sensor allows one to reconstruct the 4D light field. This reconstruction considers much more information than the standard methods described for the SH sensor, that are only able to retrieve 2D phase information on the aperture plane. The Ray Tracing model detailed in Section 3-3 reads the recorded plenoptic image and converts each illuminated pixel into a pencil of rays. Using geometrical optics these pencils are propagated back to the phase screen, such that direct phase reconstruction of this screen can be performed.

The second research question is answered by the second part of the literature study, conducted in Chapter 4. In this chapter, emphasis is placed on the nature of atmospheric turbulence and how to represent it by a mathematical model. The corresponding research question is repeated for clarity:

How can anisoplanatic phase reconstruction be conducted in a single frame?

Multiple representations have been investigated, leading to the conclusion that the phase screen model with the Propagation method is best suited for implementation in this thesis. This model describes turbulent three-dimensional volumes as a set of dominant two-dimensional layers. The wavefront sensor (WFS) input aberrations are generated by projecting each illuminated phase screen onto the pupil plane. By reversing the direction of projection a backwards model is described, that enables the reconstruction of the original phase screens, using WFS outputs.

The reconstruction of the phase screen can be performed by two methods. The first method places the WFS phase reconstructions back in the phase screen, whereas the second method places the obtained gradients in the phase screen and performs a local phase reconstruction.

The theoretical answers to both research questions are supported by experiments, conducted with the developed simulation toolbox. This toolbox is elaborated in Chapter 5 and presents the author's main contribution to this thesis. The simulations can be separated into three parts. The first part, detailed in Section 5-4 tests the toolbox functionality for isoplanatic aberrations under different experimental conditions and concludes on the best reconstruction methods. It was found that modal phase reconstruction outperforms the zonal algorithm. Additionally, the results of the single-step reconstructions and corrections were improved significantly when an iterative algorithm was used.

The second part, discussed in Section 5-5, conducts a more thorough comparison of the Shack-Hartman and plenoptic 4F sensor. To highlight the benefits of each sensor, isoplanatic single-step reconstruction is performed on Kolmogorov screens of varying strengths. The size of the WFS aperture plane is in the order of mm and as such, the Fried parameters r_0 are chosen in the order 10^{-4} and 10^{-5} [m]. In this comparison two microlens arrays are selected, one containing 11×11 lenslets and the other 33×33 . The results obtained from this comparative study confirmed the theoretical answer of the first research question: Using the same shared MLA, the SH sensor outperforms the plenoptic 4F (P4F) sensor in terms of performance metrics. The phase correction itself is also dependent on the aberration strength. In general, stronger aberrations are reconstructed better by the P4F sensor. Weaker aberrations are best corrected for using the SH WFS.

The final and third set of experiments test the anisoplanatic phase screen retrieval of the toolbox. The goal of these experiments is to choose the best retrieval method of the two available. Less emphasis is placed on a comparison between the SH and P4F sensor. The conclusions of these simulations depend on the illumination of the phase screen. If the phase screen is well-illuminated and sufficient measurements are available, the best retrieval algorithm is the back-propagation of phase gradients and performing a local reconstruction of the phase screen. This method uses all the pixels of the screen for reconstruction, filling in the unknowns with zero.

If the screen is not sufficiently illuminated and many pixels are unknown, it is best to retrieve the screen by back-propagating the WFS phase reconstructions. This results in a piece-wise discontinuous phase screen, which is smoothed as much as possible by estimating hidden piston terms. This retrieval method performs best when many screen pixels are unknown, as it only reconstructs over the known pixels.

6-1 Recommendations for Future Research

The results obtained from the simulation toolbox and drawn conclusions are based on the isoplanatic phase reconstructions. The performance of these reconstructions are carried over in the comparison of the sensors, as well as the anisoplanatic model. As such, the most important recommendation is the improvement of the simulation toolbox. The isoplanatic reconstruction error is attributed to three factors, as explained in Section 5-4-3.

The first contribution is measurement noise contained in the pixels of the SH and P4F sensor images. The second contribution is caused by the discrete Fourier transform during the imaging process, where propagated fields are not placed on the correct pixels. Instead, a shift of one or two pixels can occur, resulting in faulty gradient retrievals. This error is observed in the retrieval of phase screens, where the object-based tilt phases are not fully corrected for. The final contribution is the collection of multiple truncation errors that result from the approximation of continuous-space entities by discretely-sampled grids, or a finite number of Zernike modes. Improvements to these contributions will lead to better phase reconstruction results, enhancing the simulation toolbox.

Throughout this thesis reconstruction results are presented by the residual phases and root-mean-square (RMSE). If randomly-generated Kolmogorov screens are defined as input aberrations, it is interesting to observe the average reconstruction errors and their variations for k simulations. Such a study is not conducted in this work, but performing Monte Carlo Simulations on both SH and P4F sensors may offer an interesting comparison.

A final recommendation can be composed for the anisoplanatic aberration model. From the results presented in Section 5-6 it is clear that the phase screen retrievals are not as expected. Large errors dominate the retrieval processes, such that a comparison between the SH and P4F sensors on anisoplanatic aberrations is inconclusive. Future improvements can be focussed on the implementation of the anisoplanatic model, such as adding better optimization-routines that result in better reconstructions.

Appendix A

Background Information

This chapter details background information to Adaptive Optics (AO) and optical imaging which has proven useful throughout this thesis. The two different representations of light, geometrical and wave optics, are elaborated in Appendix A-1. The point-spread function (PSF) of a lens and diffraction-limited resolution described in Appendix A-3, followed by the mathematical framework of Zernike polynomials in Appendix A-4. The linear least squares problem and the minimum-variance unbiased estimate (MVUE) as solution is elaborated and proven in Appendix A-7.

A-1 Geometrical versus Wave Optics

The two common representations of light are *geometrical optics* and *wave optics* [16]. Geometrical optics treats light as a bundle of light rays or a *light-field* [4] as it travels through an optical system. It offers fast and simple reconstruction of the position and direction of each individual light ray using *ray tracing*. Wave optics on the other hand models light as a complex wave where diffraction effects and the phase govern the physics of the wave. As indicated by [16] geometrical optics is an approximation of wave optics, most prominently that individual light rays propagate in straight lines in homogenous media and effects such as diffraction and interference are not taken into account. Geometrical optics cannot reveal the true diffraction patterns and sensor images as wave optics can, but its simple and fast tool to predict how light should travel through optical systems. Additionally the geometric model allows one to show the three-dimensional propagation of light much more intuitively than wave optics.

A-1-1 Geometrical Optics

At each point throughout an optical system one can represent a light ray by its Cartesian position (x, y) and angular direction (α, β) . It is assumed that these rays travel through free space unhindered unless an optical element is introduced that affects their directions, such

as a thin lens. The geometrical model itself is governed by the *thin lens equation* (A-1), as stated by [16, 47]. This equation describes image formation by a thin lens and can be used to derive geometrical light propagation through such a lens. As stated in [16], a thin lens is defined as an optical element where rays enter and exit at approximately the same Cartesian location, meaning that it purely introduces a phase delay onto the incoming light. The thin lens equation is written as:

$$\frac{1}{b} + \frac{1}{v} = \frac{1}{f}, \quad (\text{A-1})$$

where b is the image distance (stems from "beeld", the Dutch word for image), v is the object distance (stems from "voorwerp", the Dutch word for object) and f is the focal length of the thin lens considered, all in [m]. These quantities are also shown in Figure A-1a that illustrates how an object is imaged by a thin lens.

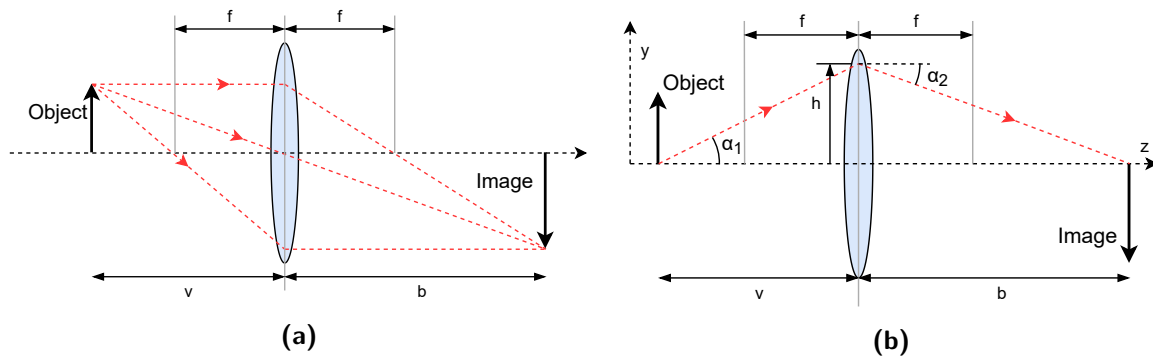


Figure A-1: 2D schematic of a thin lens showing geometrical optical quantities: a) Illustration of the thin lens equation and b) Illustration of light refraction through a thin lens

To describe the geometrical propagation, first assume a two-dimensional system such as shown in Figure A-1b. Each light ray can be fully represented by (y, z, α) , where y is the distance of the light ray from the optical axis, often the z -axis, and z is the propagation distance along the optical axis. The direction of the ray is defined by α , the angle in the yz -plane with respect to the optical axis. For example, a light ray travelling in positive y -direction away from the optical axis has a positive angle α . The effect of a thin lens on light rays is seen only by the change in angle α while keeping distance y the same. This *light refraction* is shown in Figure A-1b. The change in angle α can be found by combining the thin lens equation with the *paraxial approximation*. This approximation is valid for light rays with small angles that are close to the optical axis, such that the relative error between $\tan(\theta)$ and θ is less than 1%. Then one can write the basic trigonometric functions as:

$$\sin(\theta) \approx \theta, \quad \cos(\theta) \approx 1, \quad \tan(\theta) \approx \theta. \quad (\text{A-2})$$

This approximation allows one to reduce the expression of angle α_1 in Figure A-1b to

$\tan\left(\frac{h}{v}\right) \approx \frac{h}{v}$, such that the angle difference $\Delta\alpha$ can be simplified greatly:

$$\begin{aligned}\alpha_2 - \alpha_1 &= \frac{-h}{b} - \frac{h}{v}, \\ &= -\left(\frac{h}{b} + \frac{h}{v}\right), \\ \Delta\alpha &= \frac{-h}{f},\end{aligned}\tag{A-3}$$

where h is the height of the ray incident on the lens, positive in upward direction and expressed in [m]. The angle change induced by the thin lens is given by $\Delta\alpha$. Both $\Delta\alpha$ and α_2 are negative because the light ray is pushed towards to optical axis. Focal length f is introduced by invoking (A-1), directly showing how the lens strength affects the ray propagation. Using the above information a simple mathematical model can be derived to propagate a bundle of rays from start to finish, given the paraxial approximation (A-2) holds:

$$\begin{aligned}\alpha_2 &= \alpha_1 && \text{(free space),} \\ \alpha_2 &= \alpha_1 - \frac{y_1}{f} && \text{(thin lens),} \\ y_2 &= y_1 + (z_2 - z_1)\alpha_2,\end{aligned}\tag{A-4}$$

where the propagation from point 1, at distance along the optical axis z_1 and height y_1 , to point 2 at distance z_2 and height y_2 is considered. All distances and heights are expressed in [m]. If the ray travels through a thin lens with focal distance f the incident height on the lens is y_1 . If the ray travels through free space its angles remain unchanged and only height y changes linearly with propagation distance $z_2 - z_1$. Two special cases of (A-4) can be identified for the focal planes of a thin lens

Front focal plane If a ray passes through the front focal plane of a thin lens, located a distance f in front of the lens, it will exit the lens collimated. The propagation is considered from the front focal point (point 1) to the lens itself (point 2), meaning $z_2 - z_1 = f$. The ray enters the lens at height y_1 :

$$\begin{aligned}\alpha_1 &= \frac{y_1}{f}, \\ \alpha_2 &= \alpha_1 - \frac{y_1}{f} = 0, \\ y_2 &= y_1,\end{aligned}$$

showing that for every y_1 within the physical dimensions of the lens the ray exits the lens parallel to the optical axis.

Back focal plane Conversely, if a ray enters the lens collimated it will pass through the back focal point of the lens, often simply the lens focal point, at a distance f behind the lens.

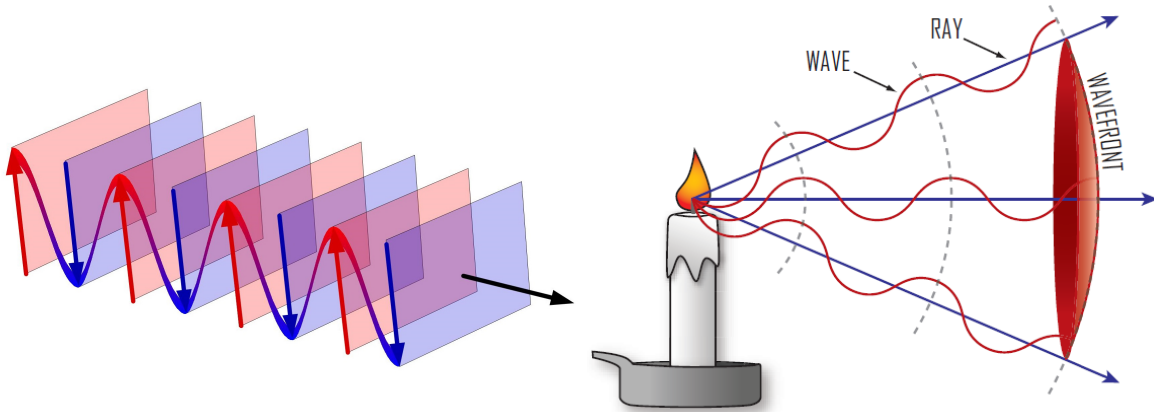
The ray enters the lens (point 1) at height y_1 and converges to the focal point (point 2):

$$\begin{aligned}\alpha_1 &= 0, \\ \alpha_2 &= -\frac{y_1}{f}, \\ y_2 &= y_1 - f \frac{y_1}{f} = 0,\end{aligned}$$

meaning that the ray ends up on the optical axis at a distance f behind the lens, exactly where the lens focal point is located.

A-1-2 Wave Optics

Instead of modelling each ray of light separately wave optics represents light as a complex wave or field $U = Ae^{j\phi}$ with magnitude A , phase ϕ and imaginary unit j . Within this field one can define surfaces of equal phase, referred to as *wavefronts*, for example the blue and orange planes shown in Figure A-2a. In the wave model one can define light rays as any vector perpendicular to such wavefronts that travel in the direction of propagation. An example of such a ray is shown in Figure A-2a by the black vector. In Figure A-2b one can see light emitted by a lit candle that is assumed to be a point source. Light can travel uniformly in all directions, which means that the wavefronts are described by spherical surfaces. Light rays travel away from this point source while remaining perpendicular to the spherical wavefronts, such as the blue vectors in Figure A-2b.



(a) A plane wave travelling through free space [50]. At **(b)** A lit candle as a point source [19]. Light is emitted each point along the optical axis, in direction of \vec{k} , one from all directions, such that the wavefront is spherical and the light rays are moving away from the source. The periodic with wavelength λ . Each plane wave represents radius of curvature equal to the distance between observed wavefront, all vectors perpendicular to this wavefront served wavefront and its source in the direction of propagation are light rays, such as \vec{v} .

At a certain distance z one can represent the wavefront by a two-dimensional function $w(x, y)$ of transverse Cartesian coordinates x and y , where $w(x, y)$, x and y are all expressed in [m]. The angles of light rays perpendicular to this wavefront are described by the wavefront

gradients:

$$\begin{aligned}\alpha_x &= \frac{\partial w(x, y)}{\partial x}, \\ \alpha_y &= \frac{\partial w(x, y)}{\partial y},\end{aligned}\tag{A-5}$$

where α_x and α_y are the ray angles between optical z axis and x - and y -axis respectively. These angles, the gradients of the wavefront, are sometimes also referred to as *wavefront tilts*. Depending on the type of application one can define ideal wavefront shapes. In the application of telescopes the objects to be imaged are stars separated by tremendous distances. This means they can be approximated by point sources located at optical infinity. As shown in Figure A-2b these stars emit spherical wavefronts but as the distance of propagation increases these wavefronts become progressively flatter. For telescopic purposes one refers to flat plane waves as ideal wavefront shapes, deviations from these ideals are referred to as *aberrations*. In Appendix A-1-1 the propagation of light was modelled by ray tracing individual light rays. Wave optics deals with complex fields instead of bundles of rays and the propagation is governed by diffraction effects. In a similar fashion to wavefronts $w(x, y)$ at distance z , one can represent light by complex field $U(x, y) = A(x, y)e^{j\phi(x, y)}$ with magnitude $A(x, y)$ and phase $\phi(x, y)$, as a function of transverse coordinates x and y . Phase and wavefront at the same distance z can be related through the following approximation:

$$\phi(x, y) = \frac{2\pi}{\lambda} w(x, y),\tag{A-6}$$

where λ is the wavelength of light in [m] and $w(x, y)$ is the wavefront surface. Combining (A-5) and (A-6) the gradients of the phase can be related to the angles of individual light rays:

$$\begin{aligned}\alpha_x &= \frac{\lambda}{2\pi} \frac{\partial \phi(x, y)}{\partial x}, \\ \alpha_y &= \frac{\lambda}{2\pi} \frac{\partial \phi(x, y)}{\partial y}.\end{aligned}\tag{A-7}$$

If at some distance z light is represented by complex field $U(x, y) = A(x, y)e^{j\phi(x, y)}$ the intensity $I(x, y)$ of this field is given by the squared magnitude [16]:

$$I(x, y) = |U(x, y)|^2 = |A(x, y)|^2\tag{A-8}$$

The propagation of wave $U(x, y)$ through free space follows a rectilinear behaviour as long as no objects or interferences cross the optical path - this is in congruence with the geometrical model. When the propagation passes an opaque screen or a diffractive object the propagation of wave $U(x, y)$ is dominated by diffraction theory. The history of this theory and wave propagation is extensively discussed in [16]. For brevity this thesis is restricted to scalar monochromatic diffraction theory, which means that the optical elements are large compared to the wavelength of light λ . The propagation model discussed below represents a summary of the elaborations in [16]. Without further ado, one can describe the propagation of a complex field $U(x, y)$ from one plane defined by coordinates $(\xi, \eta, z = z_1)$ to another defined by $(x, y, z = z_2)$ through the following integral:

$$U_2(x, y) = \frac{A}{j\lambda} \iint_{\Sigma} U_1(\xi, \eta) \frac{e^{jkr_{01}}}{r_{01}^2} \cos(\theta) d\xi d\eta,\tag{A-9}$$

where it is assumed that the input plane contains the diffractive element. The imaginary unit is represented by j , $k = \frac{2\pi}{\lambda}$ is the wave number, r_{01} is the length of the vector connecting the origins of both planes and θ the angles this vector makes with the optical axis. The distance r_{01} is given by:

$$r_{01} = \sqrt{z^2 + (x - \xi)^2 + (y - \eta)^2}. \quad (\text{A-10})$$

Integral (A-9) can be simplified by using approximations. The most common approximations are the Fresnel and Fraunhofer integrals, that implement a binomial expansion of (A-10) and retaining only the first two terms. This expansion leads to the famous Fresnel integral.

Fresnel diffraction In the regions of *near field* propagation integral (A-9) can be approximated by the Fresnel integral as explained above. By retaining the first two terms of a binomial approximation to r_{01} the Fresnel integral reveals itself:

$$U_2(x, y) = \frac{e^{jkz}}{j\lambda z} \cdot e^{j\frac{k}{2z}(x^2+y^2)} \iint_{-\infty}^{+\infty} \left[U_1(\xi, \eta) e^{j\frac{k}{2z}(\xi^2+\eta^2)} \right] e^{-j\frac{2\pi}{\lambda z}(x\xi+y\eta)} d\xi d\eta. \quad (\text{A-11})$$

The validity of the Fresnel approximation relies on the contribution of higher order terms in the binomial expansion of (A-10). If the first two terms sufficiently approximate r_{01} (A-10) and higher order terms are negligible the Fresnel approximation can accurately represent wave propagation. This condition is expanded in [16] by limiting oneself to small angles and upholding the paraxial approximation (A-2). Alternatively one can check the condition with the dimensionless Fresnel number:

$$F = \frac{a^2}{\lambda L}, \quad (\text{A-12})$$

where a is the characteristic length of the lens aperture and L is the propagation distance from lens to image plane, both in [m]. As explained in [47] the Fresnel integral (A-11) and thus (A-14) is valid when $F \gg 2$. Division of the output coordinates (x, y) by (λz) enables the rewriting of the Fresnel integral into a much more tractable two-dimensional Fourier transform. The coordinate transform can be written as:

$$\begin{aligned} x' &= \frac{x}{\lambda z}, & y' &= \frac{y}{\lambda z}, \\ x &= x'\lambda z, & y &= y'\lambda z \end{aligned} \quad (\text{A-13})$$

and the Fresnel integral becomes:

$$\begin{aligned} U_2(x'\lambda z, y'\lambda z) &= \frac{e^{jkz}}{j\lambda z} \cdot e^{j\frac{k}{2z}(x^2+y^2)} \iint_{-\infty}^{+\infty} \left[U_1(\xi, \eta) e^{j\frac{k}{2z}(\xi^2+\eta^2)} \right] e^{-2j\pi(x'\xi+y'\eta)} d\xi d\eta \\ &= \rho(x, y, z) \mathcal{F}\{U_1(\xi, \eta) e^{j\frac{k}{2z}(\xi^2+\eta^2)}\}, \end{aligned} \quad (\text{A-14})$$

where $\mathcal{F}\{\}$ represents the Fourier transform operation. The preceding terms are collected by function $\rho(x, y, z)$ which can retain the original coordinates (x, y) since it can be computed and used outside the Fourier transform operation:

$$\rho(x, y, z) = \frac{e^{jkz}}{j\lambda z} \cdot e^{j\frac{k}{2z}(x^2+y^2)}. \quad (\text{A-15})$$

Fraunhofer diffraction If the conditions for near field propagation cannot be met one can divert to the *far field* approximation, the Fraunhofer integral. The validity of this approximation can be checked by:

$$z \gg \frac{k(\xi^2 + \eta^2)_{max}}{2}. \quad (\text{A-16})$$

It has been reported by [16] that satisfying this condition results in severe parameters, such as a tremendously far away image plane. Thus an alternative approach is used, referred to as the *antenna designer's formula*:

$$z > \frac{2a^2}{\lambda}, \quad (\text{A-17})$$

Which may prove valid for some special cases. When (A-17) can be upheld the propagation of complex field U from one plane to the other is described by:

$$U_2(x, y) = \frac{e^{jkz} e^{\frac{jk}{2z}(x^2+y^2)}}{j\lambda z} \iint_{-\infty}^{+\infty} U_1(\xi, \eta) e^{-j\frac{2\pi}{\lambda z}(x\xi+y\eta)} d\xi d\eta, \\ U_2(x'\lambda z, y'\lambda z) = \rho(x, y, z) \mathcal{F}\{U_1(\xi, \eta)\}. \quad (\text{A-18})$$

Focal plane propagation Under special conditions one can simplify the Fresnel integral further. When the propagation is considered from the front to the back focal plane of an optical element, many terms of (A-14) drop out. Any optical element introduces a transformation onto the incident field. For thin lenses one can describe this transformation as a pure phase delay that, if the paraxial approximation (A-2) holds, can be written as:

$$t_l(\xi, \eta) = e^{-j\frac{k}{2f}(\xi^2+\eta^2)}, \quad (\text{A-19})$$

where f is the lens focal distance in [m]. Additionally, the lens has finite physical dimensions that it enforces on the field, which is done by multiplying the field with the lens *pupil function* P_l :

$$P_l(x, y) = \begin{cases} 1 & \text{if } (x^2 + y^2) \leq r^2 \\ 0 & \text{otherwise} \end{cases} \quad (\text{Circular lens}), \\ P_l(x, y) = \begin{cases} 1 & \text{if } |x| \leq r \text{ AND } |y| \leq r \\ 0 & \text{otherwise} \end{cases} \quad (\text{Square lens}), \quad (\text{A-20})$$

where r is the radius or half-width of the lens. When input field U_1 is placed at the front focal plane of this lens the prior term $\rho(x', y', z)$ in (A-14) is greatly reduced, as well as the term within the Fourier transform. The output field U_2 is then proportional to the Fourier transform of the pupil-limited input field U_1 :

$$U_2(\lambda f x', \lambda f y') = \frac{1}{j\lambda f} \mathcal{F}\{P_l(\xi, \eta) U_1(\xi, \eta) e^{-j\frac{k}{2f}(\xi^2+\eta^2)} e^{j\frac{k}{2f}(\xi^2+\eta^2)}\} \\ = \frac{1}{j\lambda f} \mathcal{F}\{P_l(\xi, \eta) U_1(\xi, \eta)\}. \quad (\text{A-21})$$

A more detailed elaboration of (A-21) is described by Goodman [16]. The Fraunhofer integral can be modified in similar manner, by including the lens pupil function P_l inside the Fourier transform of (A-18).

A-1-3 Free Space Propagation

The propagation of free space is a special case of light propagation, as light often travels enormous distances through near-vacuum which can be considered aberration-free. These distances will influence the Fresnel or Fraunhofer integrals significantly through parameter z . Additionally, diffraction effects will be to a minimum in near-vacuum, which means that one can resort to other models to describe free space propagation.

One of such methods is the Projection model. In the absence of atmospheric turbulence wavefronts emitted by distant stars arrive at an optical system as approximately plane waves. This means that wavefronts can be represented by two-dimensional tilted planes, where the tilt is determined solely by the stars location with respect to the optical system. Atmospheric turbulence can be added by phase screens, located between object and optical system, as implemented by [46] and illustrated in Figure A-3. The free space propagation is then approximated by projecting the phase screen onto the entry of the optical system, the pupil or aperture plane. As illustrated by Figure A-3, an arbitrary aberration profile is defined in the phase plane and two point sources are defined in the object plane. Each point source emits light that passes through a part of the phase plane, its *phase screen cut-out*. This cut-out is then projected onto the pupil plane, as shown by the blue and green dashed lines. If light is emitted incoherently, the pupil plane consists of as many distinct phase projections as there are simulated point sources in the object plane. Since no real propagation is performed but the phase screen cut-outs are simply placed in the pupil plane, this method provides a good approximation as long as weak aberrations are considered.

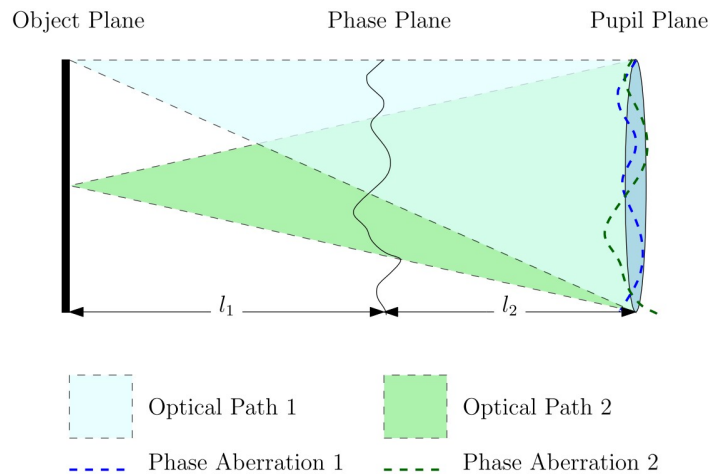


Figure A-3: Wave propagation using projection, where atmospheric turbulence is represented by a single phase screen [46]. Two objects, blue and green in the 'Object Plane', 'see' different parts of the phase screen and project this on the 'Pupil Plane'.

Another method is a straightforward extension of the geometrical optics model, where light rays travel rectilinearly from simulated object to the optical system. The influence of aberrations, such as atmospheric turbulence, can be included as well by considering phase profiles at certain points of the optical path. Using the phase gradient one can find the angular change of light rays passing through the aberration with (A-7). In the absence of diffracting

elements such as physical masks or gratings, this model provides a straightforward and good approximation for free space propagation. The grid size of light rays arriving at the optical system cannot be guaranteed in this approximation, as light rays are discarded when they move beyond the ranges of the optical system, as explained in the Ray Tracing algorithm of Wu, Ko and Davis [52]. This means that very dense light ray beams must be simulated at the objects, in order to achieve sufficient grid size at the optical system. In this work this free space model will carry the name of Ray Tracing model, as the free space propagation described in [52] does exactly this.

Compared to the Ray Tracing model, the Projection model does guarantee a grid size density as no information is discarded. If the pupil plane is to be represented by an $N \times N$ discretely-sampled grid and one knows the locations of objects, the size of phase screen cut-outs and thus the minimum grid size density of the phase screen can be determined. This advantage of the Project model may compensate its drawback of being valid for weak aberrations.

A-2 The Discrete Fourier Transform

The Fourier transform in the wave propagation equations (A-14), (A-18) and (A-21) is performed in discrete-space by the Fast Fourier Transform (FFT) algorithm in MATLAB. The behaviour of the FFT is explained by [25], stating that it behaves just as the continuous-time Fourier transform but the number of represented spatial frequencies is severely limited by the grid size. To quantify this, one defines a one-dimensional function $g(x_n)$, sampled at N equidistant points separated by Δx . The FFT of $g(x_n)$ results in function $G(u_n)$ that is sampled at N equidistant points as well, separated by Δu . For clarity the sequences x_n and u_n are referred to as the spatial coordinates and spatial frequencies, respectively. The spatial frequency step size Δu is equal to the spatial range $r_x = \max(x_n) - \min(x_n)$:

$$\Delta u = \frac{1}{r_x}. \quad (\text{A-22})$$

The FFT is performed in MATLAB by combining the *fft*- and *fftshift*-functions, which results in spatial frequencies with the zero frequency in the middle. The minimum and maximum frequencies are then found by:

$$\begin{aligned} u_{min} &= -\Delta u \frac{N}{2}, & u_{max} &= \Delta u \frac{N}{2} & (N \text{ is even}), \\ u_{min} &= -\Delta u \frac{N-1}{2}, & u_{max} &= \Delta u \frac{N-1}{2} & (N \text{ is uneven}) \end{aligned} \quad (\text{A-23})$$

and the sequence of spatial frequencies can be written as:

$$u_n = u_{min} + n\Delta u, \quad n = 0, \dots, N. \quad (\text{A-24})$$

Input and Output Ranges With these definitions the frequency range $r_u = \max(u_n) - \min(u_n)$ can be defined:

$$\begin{aligned} r_u &= u_{max} - u_{min} \\ &= \begin{cases} \frac{N}{r_x} & (N \text{ is even}) \\ \frac{N-1}{r_x} & (N \text{ is uneven}) \end{cases}. \end{aligned} \quad (\text{A-25})$$

Definition (A-25) describes how the ranges of input and output coordinates are related in the one-dimensional case. This result is readily generalized to multiple dimensions.

In (A-14) coordinate change (A-13) was introduced to rewrite the integral as a Fourier transform. By performing a dimensional analysis on these transformed coordinates, shown below only for coordinate x as an example, reveals that these transformed coordinates are actually spatial frequencies:

$$\dim x' = \dim \left(\frac{x}{\lambda f} \right) = \frac{m}{m \cdot m} = m^{-1}.$$

This means that by propagating a discretely-sampled input field $U_1(\xi, \eta) \in \mathbb{R}^{N \times N}$ through a lens with (A-21) results in an output field $U_2(x', y') \in \mathbb{R}^{N \times N}$, where ξ , η , x and y are the spatial coordinates. To obtain the output field at its spatial coordinates x and y instead of its spatial frequencies, one must transform coordinates (x', y') back to their spatial counterparts by inversely applying (A-13). Such a coordinate transform does not alter the discretely-sampled matrix U_2 but scales the plane this matrix is represented on. The physical size of this plane, the output spatial plane, is found by combining (A-25) and (A-13):

$$x = \lambda f x' \rightarrow r_x = \lambda f r_{x'} = \frac{\lambda f N}{r_\xi}.$$

where λ is the wavelength of light and f the lens focal distance, both expressed in [m]. If grid size N is uneven one can substitute $(N - 1)$ for N such that the bottom range definition of (A-25) is used instead.

Minimum Grid Size for Lenses Not only does grid size N affect the physical plane sizes in the discrete Fourier transform (DFT), also the frequency representation of the Fourier transformed field is affected. Common pitfalls of the FFT algorithm are discussed in [15], one of which is *aliasing*. In discrete signal processing aliasing occurs when high frequency components of a spatial or temporal signal are translated into low frequencies if the sample rate is too low. This is due to the cyclic nature of the DFT, where frequencies that fall outside of the sampling range are assumed to be lower frequencies of a different cycle. This means that these higher frequencies are 'folded back' to lower frequencies, resulting in an incorrect representation. The effects of aliasing can be reduced by including more frequencies in the frequency coordinates, defined in (A-22)-(A-25). This is done by increasing the ratio N/r_x in (A-25). By increasing grid size N one *increases the range of detectable frequencies*. If spatial coordinate range r_x is decreased one obtains a *coarser sampled grid of frequencies*. It is clear from [15] that exact requirements to grid size and spatial range depend on the processed signal, in this case the complex wave U and especially phase ϕ . In practice information on the type of phase aberration is not known *a priori*, but one does have information on the lenses in the wavefront sensor (WFS). In continuous-space one can describe the complex field directly behind a this lens, resulting from focussing field $U(x, y) = A(x, y)e^{j\phi(x, y)}$, by:

$$U_l(x, y) = P(x, y)A(x, y)e^{j(\phi(x, y) - \phi_l(x, y))}, \quad (\text{A-26})$$

where $P(x, y)$ is the lens pupil function as defined by (A-20), $\phi(x, y)$ the unknown to-be-sensed aberration and $\phi_l(x, y)$ the phase delay induced by the lens as defined by (A-19):

$$\phi_l = -\frac{k}{2f} (x^2 + y^2),$$

with wave number $k = 2\pi/\lambda$. The goal is to correctly sample the complex field that ends up on the sensor, but without knowledge on aberration ϕ the problem is ill-posed. Instead one can assume that the lens phase delay is significantly stronger than the unknown aberration, such that the field directly behind the lens can be approximated by:

$$U_l(x, y) \approx P(x, y)A(x, y)e^{-\frac{jk}{2f}(x^2+y^2)}. \quad (\text{A-27})$$

This approximation allows one to formulate requirements to correctly sample lens propagation, assuming the lens phase delay is stronger than the aberrations. Correct sampling can be defined with the Nyquist-Shannon sampling theorem [15, 39, 45], which requires the maximum frequency in the Fourier transformed field to be less than or equal to half of the sampling period. The Fourier transform samples at instances of 2π and the represented frequencies are dependent on the phase differential, meaning that the following condition can be derived for the lens phase differential:

$$|\delta\phi_l| < \pi. \quad (\text{A-28})$$

In the case of a circular symmetric lens with radial coordinate $\rho^2 = x^2 + y^2$ and radius R , this differential can be written as a function of the radial differential $\delta\rho$:

$$\begin{aligned} \phi_l(\rho) &= -\frac{2\pi}{2\lambda f}\rho^2 \\ \delta\phi_l(\rho) &= -\frac{4\pi}{2\lambda f}\rho\delta\rho \\ |\delta\phi_l(\rho)| &= \frac{2\pi}{\lambda f}\rho\delta\rho < \pi. \end{aligned}$$

The field directly behind the lens is pupil limited by $P(x, y)$ such that the phase outside this pupil function is undefined and thus not considered. The lens phase ϕ_l valid inside the pupil function has its maximum value and phase gradient at the outer edge, where $\rho = R$:

$$\pi > \frac{2\pi R}{\lambda f}\delta\rho \rightarrow \delta\rho < \frac{\lambda f}{2R}.$$

If the radial coordinate, defined from $-R$ to $+R$, is represented by an $N \times N$ discretely-sampled grid then differential $\delta\rho$ becomes the fixed radial step size $\Delta\rho = 2R/(N-1)$. Combining this step size with the above inequality results in a minimum grid size, to ensure correct sampling of the lens phase delay:

$$\frac{2R}{N-1} < \frac{\lambda f}{2R} \rightarrow N > \frac{4R^2}{\lambda f} + 1. \quad (\text{A-29})$$

Condition (A-29) should be checked for each lens in the optical system to compute the minimum grid size or densities required for propagation.

Minimum Grid Size Image Plane Result (A-29) presents the minimum grid size inside a lens ensuring large enough frequency ranges in the Fourier transform. Instead of increasing the grid size one can change the spatial range of input coordinates, as shown in (A-25). This is convenient when a fixed pixel size is required. Assume the one-dimensional case where the input spatial coordinate is ξ , sampled at N equidistant points. The range of output spatial coordinates, resulting from discretized propagation (A-21), r_x is given by:

$$r_x = \frac{\lambda f N}{r_\xi}. \quad (\text{A-30})$$

If the spatial output coordinates are sampled at M equidistant points separated by fixed pixel size p , one can substitute $r_x = Mp$ into (A-30) to obtain:

$$M = \frac{\lambda f N}{r_\xi p}. \quad (\text{A-31})$$

The DFT preserves grid sizes, which means that the input coordinates ξ must be sampled with grid size M as well. To retain the same range in the output plane r_x sample density N/r_ξ should be kept constant. This means that the input range and grid size are scaled linearly such that coordinate x is sampled at M equidistant points with range:

$$h_{in} = \frac{M}{N} r_\xi \quad (\text{A-32})$$

where h_{in} is the new spatial range of input coordinate ξ . Physical signals that are sampled as a function of these coordinates, the increase in spatial range is filled by *padding* the physical signal with zeroes. This is demonstrated in Figure A-4, for the same one-dimensional case. In the front focal plane of a lens phase $\phi \in \mathbb{R}^N$ is initially sampled on spatial grid $\xi \in \mathbb{R}^N$. The pupil mask of the lens is represented by the same coordinates, where the spatial range of ξ is equal to the lens diameter D . In order to fix pixel size p a grid size of M is required for ξ , which results in padding both ϕ and the pupil mask with zeros to obtain spatial range h_{in} . The two-dimensional pupil mask of the circular lens is shown on the right of Figure A-4, showing the dashed outline of the pupil function, the initial $N \times N$ mask and the zero-padded $M \times M$ mask. The spatial size of the image $p \cdot M$ is not necessarily equal to h_{in} .

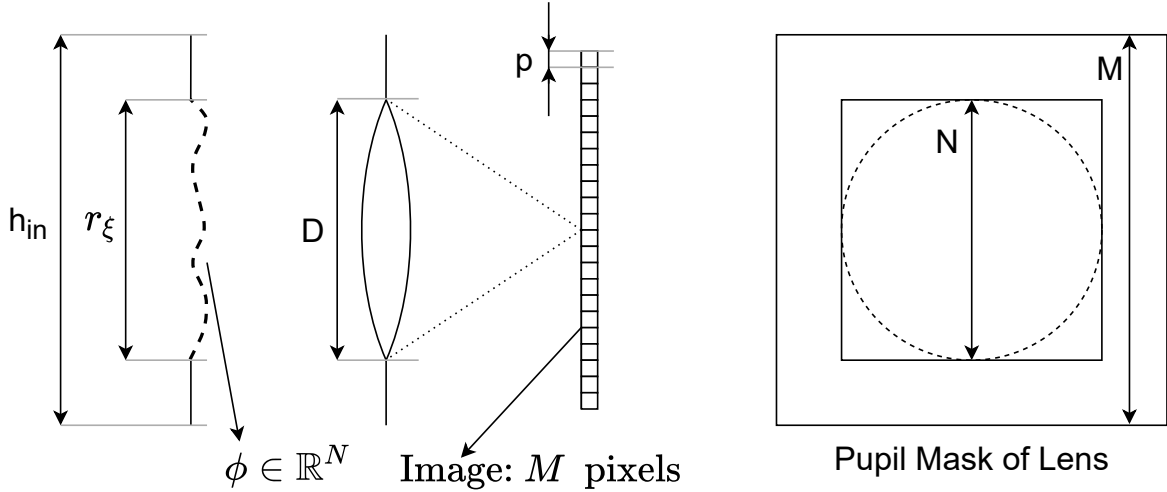


Figure A-4: Illustration of numerically simulating a lens with fixed pixel size. The leftmost illustration shows the one-dimensional case where aberration $\phi \in \mathbb{R}^N$ and pupil mask are padded with zeros to obtain a grid size of M and ensure pixel size p . The rightmost illustration shows how a two-dimensional pupil mask of the circular lens is padded. The dashed outline traces the circular pupil function, which is initially represented by the $N \times N$ mask and padded with zeros to obtain the required $M \times M$ mask.

A-3 The Point-Spread Function

Following the explanation in [46], any lens focusses incoming light into a spot in the focal point, but due to finite lens sizes and aberrations this spot will never be a single infinitesimal point. The image of a point source in the back focal plane is called the *point-spread function* and can be found by taking the Fourier transform of the pupil-limited field in the front focal plane, as explained in Appendix A-1-2. Under aberration-free situations the incoming wavefront will be a plane wave and the PSF reduces to the Fourier transform of the pupil function. The resulting image is the *Airy pattern* shown in Figure A-5. Even for a perfect lens and absence of aberrations remains a blob accompanied by circular lobes, instead of a single pixel. This pattern is caused by diffraction effects of the finite pupil size and will always be present in the image. Imaging under aberration-free conditions is thus referred to as *diffraction-limited* imaging. The resolution of a diffraction-limited image is quantified by the *Rayleigh resolution criterion* [17]:

$$\sin(\theta) \approx 1.22 \frac{\lambda}{d}, \quad (\text{A-33})$$

where λ is the wavelength of light, d is the aperture diameter in [m] and θ in [rad] is the angular distance from the centre of the Airy pattern to its first dark ring. This resolution is based on the resolving power of two independent point sources that result in two Airy patterns on the image. Both Airy patterns can be distinguished as long as the angular distance between their centres is larger than (A-33). If this distance is smaller the centre of one of the Airy spots is within the first dark ring of the other and they will be imaged as one spot rather than two.

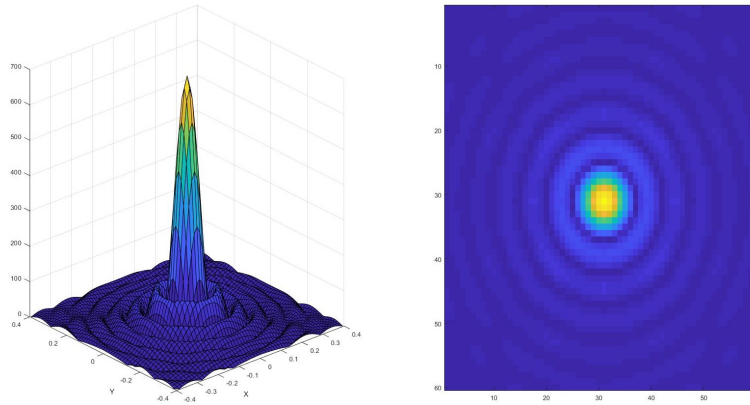


Figure A-5: An example of an Airy pattern

Using the paraxial approximation (A-2) one can simplify the sine in (A-33) to obtain an expression of absolute the image plane distance y :

$$\sin(\theta) = \sin\left(\frac{y}{f}\right) \approx \frac{y}{f},$$

where f is the distance from lens to imaging plane which is equal to the lens focal distance due to back focal plane imaging. This approximation enables one to write (A-33) in terms of distance y instead of angle θ :

$$y \approx 1.22 \frac{\lambda f}{d}. \quad (\text{A-34})$$

If aberrations are present the PSF will change its shape and location depending on the aberration profile. For example one can find the PSFs of the first 15 Zernike polynomials in Figure A-7.

A-4 Zernike Polynomials

Zernike polynomials are defined by polar coordinates ρ and θ , but specified by their two indices m and n . These indices are both non-negative for which the following must hold:

$$m, n \in \mathbb{Z}_{\geq 0}, \quad n \geq m, \quad n - m \in 2\mathbb{Z}, \quad (\text{A-35})$$

meaning that non-negative integers m and n have the same *parity*, both are either even or odd and n is always equal or larger than m . The polynomials themselves are determined by these integers:

$$Z_n^{\pm m}(\rho, \theta) = R_n^m(\rho) \begin{cases} \cos(m\theta) & \pm m > 0 \\ \sin(m\theta) & \pm m < 0 \\ 1 & m = 0 \end{cases}, \quad (\text{A-36})$$

with $R_n^m(\rho)$ is the radial polynomial defined by:

$$R_n^m(\rho) = \sum_{s=0}^{\frac{n-m}{2}} \frac{(-1)^s (n-s)!}{s! (\frac{n+m}{2} - s)! (\frac{n-m}{2} - s)!} \rho^{n-2s}. \quad (\text{A-37})$$

For presentation purposes these Zernike polynomials are arranged in the Zernike pyramid, shown in Figure A-6 for a number of these polynomials. An more straightforward single index numbering was introduced by Noll [31] that resulted in modified Zernike polynomials with a different normalisation, shown for the first 15 modes in Figure A-7. Noll's ordering has been used throughout this thesis to generate Zernike polynomials.

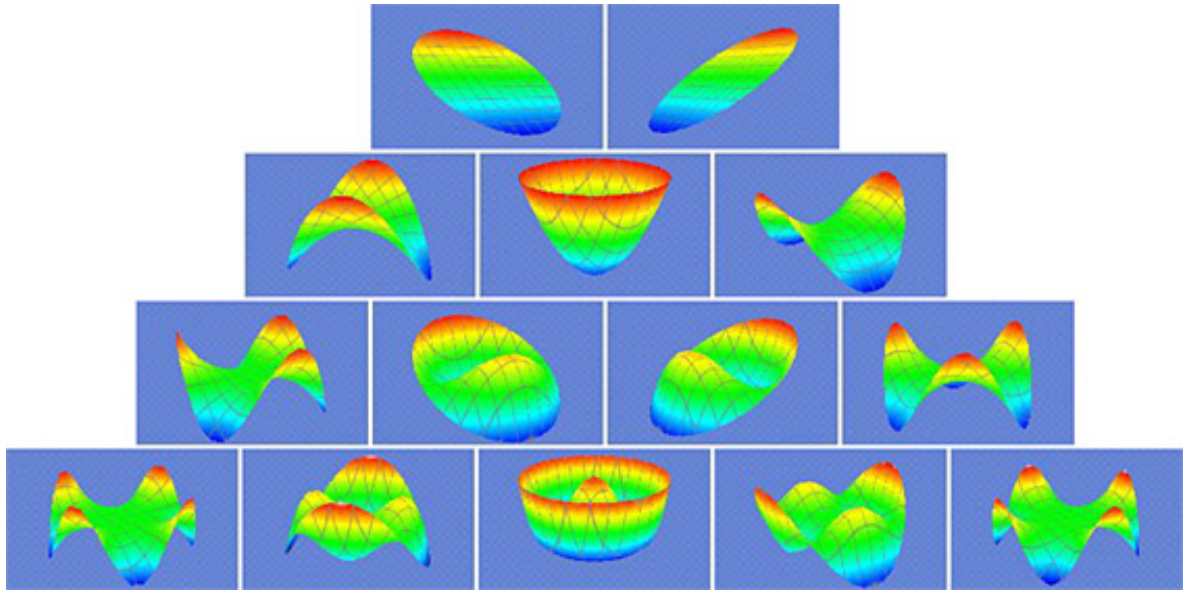


Figure A-6: The Zernike modes arranged in a pyramid for the first 14 modes, excluding the first mode which is the constant piston [3].








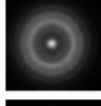







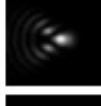











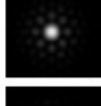


Index	Noll's ordering	Name	Expression	Shape	PSF
$Z_0^0(x, y)$	1	piston	1		
$Z_1^1(x, y)$	2	tip	x		
$Z_1^{-1}(x, y)$	3	tilt	y		
$Z_2^0(x, y)$	4	defocus	$2x^2 + 2y^2 - 1$		
$Z_2^2(x, y)$	5	astigmatism	$x^2 - y^2$		
$Z_2^{-2}(x, y)$	6	astigmatism	$2xy$		
$Z_3^1(x, y)$	7	coma	$3x^3 + 3y^2x - 2x$		
$Z_3^{-1}(x, y)$	8	coma	$3y^3 + 3x^2y - 2y$		
$Z_3^3(x, y)$	9	trefoil	$x^3 - 3xy^2$		
$Z_3^{-3}(x, y)$	10	trefoil	$3x^2y - y^3$		
$Z_4^0(x, y)$	11	spherical	$6x^4 + 12y^2x^2 - 6x^2 + 6y^4 - 6y^2 + 1$		
$Z_4^2(x, y)$	12		$4x^4 - 3x^2 - 4y^4 + 3y^2$		
$Z_4^{-2}(x, y)$	13		$8yx^3 + 8y^3x - 6yx$		
$Z_4^4(x, y)$	14		$x^4 - 6y^2x^2 + y^4$		
$Z_4^{-4}(x, y)$	15		$4x^3y - 4xy^3$		

Figure A-7: The Zernike modes arranged according to the Noll ordering with a single index, shown for the first 15 modes [49].

A-5 Basic Modal Reconstruction

The modal approach, as explained in [21, 23, 31], represents the incoming wavefronts as an infinite sum of basis functions:

$$w(x, y) = \sum_{j=1}^{+\infty} a_j Z_j(x, y), \quad (\text{A-38})$$

where $w(x, y) \in \mathbb{R}^{N \times N}$ is the two-dimensional wavefront, $Z_j(x, y) \in \mathbb{R}^{N \times N}$ the basis function of the j -th mode and $a_j \in \mathbb{R}$ its corresponding magnitude or coefficient. There are many bases to choose from, such as Zernike polynomials, Karhonen-Loève functions or Lukosz-Braat functions but the orthogonality of the selected basis is crucial, as noted by [21, 23, 24, 42]. Most importantly, orthogonality enables the reconstruction to be represented by independent functions and that the reconstructed coefficients are uncorrelated. Additionally the error propagation and computational complexity is significantly reduced with orthogonal bases.

For this work the Zernike polynomials, that are orthogonally defined over the unit circle, have been chosen because of their straightforward description, clear mathematical definition, tractability and correspondence to common optometric aberrations [5, 21]. The infinite sum in (A-38) is not tractable in reality so a finite sum up to a number of M modes is used instead. Furthermore, the summation can be written as a matrix-vector multiplication resulting in:

$$w \approx ZA, \quad (\text{A-39})$$

where $w \in \mathbb{R}^{N^2 \times 1}$ is the vectorized wavefront, $Z \in \mathbb{R}^{N^2 \times M}$ contains the Zernike polynomials in its M columns and $A \in \mathbb{R}^{M \times 1}$ contains the M coefficients. In reconstructing the first M Zernike modes the first mode is often left out [7, 23]. This polynomial represents the global piston, a constant mean phase value that does not affect the phase differences between phase points and cannot be detected by a sloped-based WFS [5, 43]. Fortunately it does not affect the slope measurements and can thus be safely ignored from reconstruction [7, 43]. For more details on the mathematical description of Zernike polynomials the reader is referred to Appendix A-4.

The goal of the modal algorithm is to estimate the coefficients A that represent the weighing of the individual basis functions. It is not possible to solve (A-39) using a gradient-based WFS such as the Shack-Hartmann (SH) or plenoptic sensor, since neither w or A is known. Instead, one obtains a set of slopes (Δ_x, Δ_y) that are related to the gradients of Z :

$$(\Delta_x, \Delta_y) \propto \left(\frac{\partial Z}{\partial x}, \frac{\partial Z}{\partial y} \right) A + \epsilon, \quad (\text{A-40})$$

where ϵ represent the noise contributions, such as a truncation error that arises by approximating the wavefront by a finite sum. The system of equations (A-40) can be rewritten as an matrix-vector multiplication (MVM) by appropriate stacking of the slopes and gradient matrices. Throughout this thesis this stacking is done by alternating between x - and y -components as shown in (A-41). The stacking for modal reconstruction is arbitrary, but is chosen to be consistent with the zonal reconstruction method Appendix A-6. Slopes Δ_x and Δ_y are collected by matrix $S \in \mathbb{R}^{N^2 \times M}$, whereas the gradients of each Zernike mode are

collected by matrix $G \in \mathbb{R}^{2N^2 \times M}$. For each j -th mode the gradients in x and y -direction are represented by G_x^j and G_y^j respectively. The stacking is then defined by:

$$S = \begin{bmatrix} \Delta_{x,1} \\ \Delta_{y,1} \\ \vdots \\ \Delta_{x,N^2} \\ \Delta_{y,N^2} \end{bmatrix}, \quad G = \begin{bmatrix} G_{x,1}^1 & \cdots & G_{x,1}^M \\ G_{y,1}^1 & \cdots & G_{y,1}^M \\ \vdots & \ddots & \vdots \\ G_{x,N^2}^1 & \cdots & G_{x,N^2}^M \\ G_{y,N^2}^1 & \cdots & G_{y,N^2}^M \end{bmatrix}, \quad (\text{A-41})$$

where the subscript in the entries of S and G indicate the measurement point on the $N \times N$ grid. The stacked are used to describe the slope data equation.

$$S = GA + \epsilon, \quad (\text{A-42})$$

The data equation (A-42) can be solved through least-squares minimization [12, 17, 21, 48], elaborated in Appendix A-7. As stated by [21] this estimation is unique if $M < 2N^2$ or in other words, the number of modes considered should be always smaller than the total number of measurement points available. The same condition could be set for (A-39), since the phase should be represented by an independent set of modes. This results in the condition $M < N^2$. In practice this means that the grid size N of the sensor must be large, in order to accurately represent that phase by a large collection of polynomials. At each time-step k one can estimate the coefficients and reconstruct the wavefront by invoking (A-39):

$$\tilde{w}(k) = Z\tilde{A}(k), \quad (\text{A-43})$$

with $\tilde{w}(k) \in \mathbb{R}^{N^2 \times 1}$ the estimated vectorized wavefront at time-step k , $Z \in \mathbb{R}^{N^2 \times M}$ the polynomial matrix as defined above and $\tilde{A}(k) \in \mathbb{R}^{M \times 1}$ the estimated coefficients of the considered modes at time-step k . By taking approximation (A-6) into account one can obtain the reconstructed phase from (A-43).

A-5-1 Implementation with Square Sub-images

This thesis considers square microlens array (MLA) lenslets, which means that the obtained slopes are represented by a square grid and the orthogonality of the chosen Zernike basis will be lost. The same problem has been described by [23] and solved by generating two orthonormal sets of polynomials. These sets have been produced such that each column has unit length and is orthogonal to the other columns. The obtained sensor slopes Δ_x and Δ_y are both represented on the same $N \times N$ grid as each mode in Z (A-39). They are stacked according to (A-41) and collected in matrix $S \in \mathbb{R}^{2N^2 \times M}$.

The first set is obtained by generating Zernike polynomials on a unit square and performing orthonormalization using the Gram-Schmidt algorithm. This set is represented by matrix $Z_\perp \in \mathbb{R}^{N^2 \times M}$. The second set is obtained by computing the gradients of the original Zernike polynomials and performing orthonormalization on these x - and y -gradients as a whole. This second set is represented by matrix $G_\perp \in \mathbb{R}^{2N^2 \times M}$ and is stacked according to (A-41). The result is an orthonormal set of Zernike polynomials to describe the phases in a linearly independent manner and an orthonormal set of gradients to estimate magnitudes A .

Matrices Z_{\perp} and G_{\perp} are obtained using the Gram-Schmidt method, implemented by the MATLAB code obtained from Chen [8]. The orthonormalized matrices are related to their originals through conversion matrices β :

$$\begin{aligned} Z &= Z_{\perp} \beta_Z, \\ G &= G_{\perp} \beta_G. \end{aligned} \tag{A-44}$$

The orthonormal results retain the dimensions of the original matrices such that $\{Z, Z_{\perp}\} \subset \mathbb{R}^{N^2 \times M}$, $\{G, G_{\perp}\} \subset \mathbb{R}^{2N^2 \times M}$ and $\{\beta_Z, \beta_G\} \subset \mathbb{R}^{M \times M}$. The phase and gradients are then represented by their orthonormal sets resulting in the following data equations:

$$\begin{aligned} \phi &= \frac{2\pi}{\lambda} Z_{\perp} \alpha, \\ S &= \frac{2\pi}{\lambda} G_{\perp} \gamma + n, \end{aligned} \tag{A-45}$$

where (A-6) is invoked to approximate the phase and its gradients from the wavefront. Both α and γ are the magnitudes of the orthonormal Zernike modes and orthonormal gradients, respectively. It should be noted that these coefficients are not equal, in contrast to representations (A-39) and (A-40) where orthonormalization was not employed.

The noise contributions in (A-45) have been contained to the slope measurement data equation since these slopes contain the majority of noise, such as detector noise. Additionally, the actual reconstruction in the modal algorithm is finding γ using S , adding the modal truncation error introduced in (A-40) to the reconstruction of γ . The noise contributions in the phase data equations are thus implicitly included through α and do not need to be added explicitly.

In order to find α from reconstructed coefficients γ one must apply (A-44) with the knowledge that β_Z and β_G are both square full-rank matrices and thus invertible - This will be illustrated below. The goal of the phase reconstruction is reduced to finding an expression for α , the magnitudes of the orthonormal Zernike modes, using the slope measurements:

$$\begin{aligned} S &= \frac{2\pi}{\lambda} G_{\perp} \gamma + n \\ &= \frac{2\pi}{\lambda} G \beta_G^{-1} \gamma + n \\ &= \frac{2\pi}{\lambda} G A + n, \end{aligned}$$

where A is a vector containing the magnitudes of the original gradients G , which is equal to the magnitudes of original modes Z :

$$\begin{aligned} \phi &= \frac{2\pi}{\lambda} Z_{\perp} \alpha + n \\ &= \frac{2\pi}{\lambda} Z \beta_Z^{-1} \alpha + n \\ &= \frac{2\pi}{\lambda} Z A + n \end{aligned}$$

and thus the orthonormalized coefficients can be related through:

$$\begin{aligned} \beta_G^{-1} \gamma &= A = \beta_Z^{-1} \alpha, \\ \alpha &= \beta_Z \beta_G^{-1} \gamma. \end{aligned} \tag{A-46}$$

The γ coefficients are estimated using the minimum-variance unbiased (MVU) estimate $\tilde{\gamma}$:

$$\tilde{\gamma} = C_\gamma G_\perp^T (G_\perp C_\gamma G_\perp^T + C_\epsilon)^{-1} S, \quad (\text{A-47})$$

where $C_\gamma \in \mathbb{R}^{M \times M}$ and $C_n \in \mathbb{R}^{2N^2 \times 2N^2}$ are the coefficient and noise covariance matrices and $G_\perp \in \mathbb{R}^{2N^2 \times M}$ the set of orthonormal Zernike gradients. More information on the elaboration and proof of the MVUE is found in Appendix A-7. Combining (A-46) and (A-47) enables one find an expression for the estimate of orthonormal magnitudes α :

$$\tilde{\alpha} = \beta_Z \beta_G^{-1} C_\gamma G_\perp^T (G_\perp C_\gamma G_\perp^T + C_\epsilon)^{-1} S. \quad (\text{A-48})$$

Finally one can reconstruct the phase by invoking the previously stated relationships:

$$\tilde{\phi} = \frac{2\pi}{\lambda} Z_\perp \tilde{\alpha}. \quad (\text{A-49})$$

Up to this point no limitation to the number of modes is taken into account. Lane and Tallon posed in [21] that the number of modes M should always be smaller than the number of measurements N^2 or $2N^2$ for either wavefront (A-39) or slope representation (A-42). This condition was presented under circular sub-images without additionally orthonormalization of the matrices. The next paragraph sheds some light on how the orthonormality of the matrices changes with increasing number of modes M .

The number of modes is upper-limited by $M < N^2$, such that ill-conditioning is prevented in both phase reconstruction (A-42) and representation (A-39).

A-6 Basic Zonal Reconstruction

Instead of basis functions, zonal reconstruction describes the wavefront by spatial sampling and considering local deformations, as explained by [17]. The obtained slope measurements are related to wavefront gradients through sensor geometry:

$$\begin{aligned} \Delta x &= \kappa_x \frac{\partial w(x, y)}{\partial x} + \eta_x, \\ \Delta y &= \kappa_y \frac{\partial w(x, y)}{\partial y} + \eta_y, \end{aligned} \quad (\text{A-50})$$

where κ_x and κ_y are determined through sensor geometry and η_x and η_y are noise attributions. With an approximation method, such as Finite Differences, the wavefront can be expressed by its gradients, such that (A-50) is rewritten into a direct relation between obtained slopes and the phase. In a similar fashion to the modal approach, this equation can be solved by the MVUE from Appendix A-7.

The Finite Difference approximation is one of the most well-known zonal reconstruction methods, as put by [12] and supported by [7, 17, 34, 42, 43]. These reports identified the three most common geometry models as the Fried, Hudgins and Southwell geometries, which define the exact approximation between phase and its gradients. In this thesis the Finite Difference model is used for zonal reconstruction. Its straightforward application and the three common geometry models will be elaborated below.

Fried geometry Fried geometry places the phase in the corners of each cell, where the phase is approximated by summation of these corners, as shown in Figure A-8 and Figure A-9 on the left. If the coordinate system chosen has the vertical coordinate positive upwards and the horizontal component positive rightwards, the gradient is approximated by:

$$\begin{aligned}\frac{\partial\phi_1}{\partial x} &\approx \frac{\phi_4 + \phi_2 - \phi_3 - \phi_1}{2D} + \eta_{x,1}, \\ \frac{\partial\phi_1}{\partial y} &\approx \frac{\phi_4 + \phi_3 - \phi_2 - \phi_1}{2D} + \eta_{y,1},\end{aligned}\tag{A-51}$$

with D being the size of the cells and η is a noise term containing for instance approximation errors, measurement noise and higher order aberrations effects.

Hudgins geometry Hudgins geometry places the phase in the centers of each cell and the gradients in between phase points. The gradient in each direction is approximated by the phase in the current and adjacent cell. Taking the same coordinate system as defined in the Fried case, the Hudgins approximation is formulated in (A-52) and illustrated by Figure A-8 and Figure A-9, both in the middle.:

$$\begin{aligned}\frac{\partial\phi_1}{\partial x} &\approx \frac{\phi_2 - \phi_1}{D} + \eta_{x,1}, \\ \frac{\partial\phi_1}{\partial y} &\approx \frac{\phi_3 - \phi_1}{D} + \eta_{y,1}.\end{aligned}\tag{A-52}$$

Southwell geometry Southwell geometry places the phase points just as the Hudgins model does, with gradients in the cell centres, similar to the Fried model. This results in an approximation (A-53) that uses both the adjacent phase points, as well as the adjacent phase gradients and is illustrated by Figure A-8 and Figure A-9, both on the right.

$$\begin{aligned}\frac{1}{2} \left(\frac{\partial\phi_1}{\partial x} + \frac{\partial\phi_2}{\partial x} \right) &\approx \frac{\phi_2 - \phi_1}{D} + \eta_{x,1}, \\ \frac{1}{2} \left(\frac{\partial\phi_1}{\partial y} + \frac{\partial\phi_3}{\partial y} \right) &\approx \frac{\phi_3 - \phi_1}{D} + \eta_{y,1}\end{aligned}\tag{A-53}$$

and

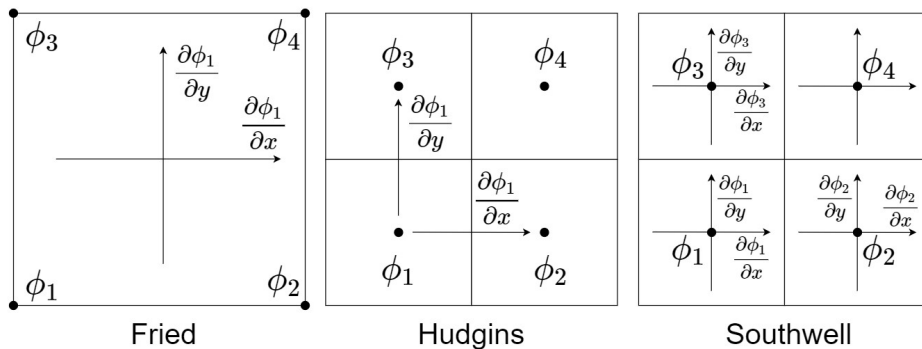


Figure A-8: Representation of the three approximation geometries for a single phase gradient, showing how each model places the phase and gradients across the grid and which ones are needed to find phase gradients $\frac{\partial\phi_1}{\partial x}$ and $\frac{\partial\phi_1}{\partial y}$. Each square block represents an individual cell.

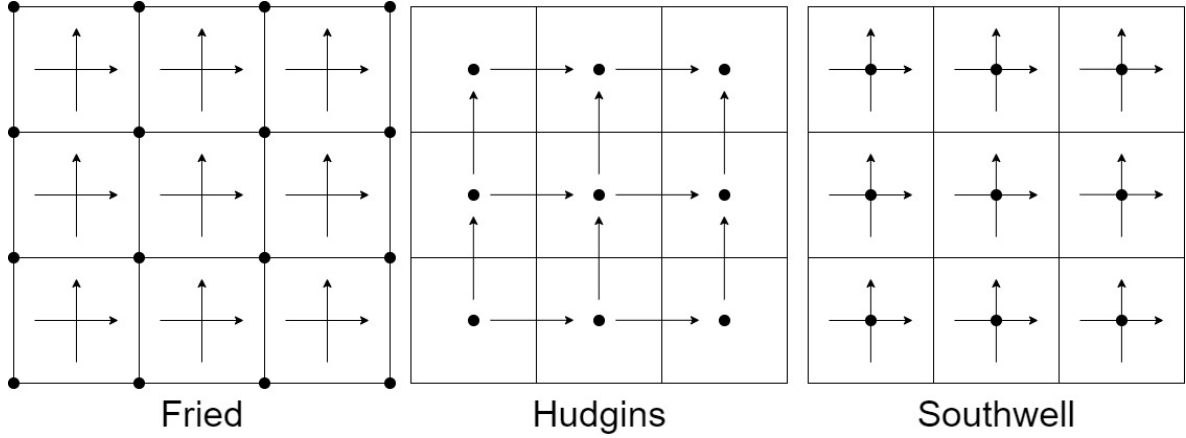


Figure A-9: Representation of the three approximation geometries. Each grid represents the image plane subdivided into cells. The phase gradients sensed by the sensor are represented by the arrows, whereas the points of the reconstructed phase ϕ are indicated by the black dots. The Fried model approximates the gradient in each block by using all four phase samples on each corner, the Hudgins model only uses two adjacent phase points in each direction and the Southwell model approximates the average of two gradients using two adjacent phase samples.

In order to obtain solvable systems of equations one needs to stack the slope measurements appropriately, analogous to the modal approach. Considering that the zonal approach represent the phase by its local deformations instead of an orthogonal basis, one does not need to enforce orthogonality in the stacking process. Assume that the wavefront is represented on a discretely sampled square $N \times N$ grid, as well as its gradients. By stacking the phase gradients, alternating between x - and y -gradients and vectorizing the phase points ϕ_i the geometry equations (A-51)-(A-53) can be written for all available data points:

$$\begin{bmatrix} \frac{\partial \phi_1}{\partial x} \\ \frac{\partial \phi_1}{\partial y} \\ \vdots \\ \frac{\partial \phi_N}{\partial x} \\ \frac{\partial \phi_N}{\partial y} \end{bmatrix} = G \cdot \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{M-1} \\ \phi_M \end{bmatrix}, \quad (\text{A-54})$$

with geometry matrix $G \in \mathbb{R}^{2N^2 \times M}$. The number of phase point obtained depends on the geometry model used, for example $M = (N + 1)^2$ using Fried geometry and $M = N^2$ using Southwell geometry. Stacking (A-50) in an analogous fashion leads to:

$$\begin{bmatrix} \Delta x_1 \\ \Delta y_1 \\ \vdots \\ \Delta x_N \\ \Delta y_N \end{bmatrix} = \kappa \cdot \begin{bmatrix} \frac{\partial w_1}{\partial x} \\ \frac{\partial w_1}{\partial y} \\ \vdots \\ \frac{\partial w_N}{\partial x} \\ \frac{\partial w_N}{\partial y} \end{bmatrix} + n = \kappa \frac{\lambda}{2\pi} \cdot \begin{bmatrix} \frac{\partial \phi_1}{\partial x} \\ \frac{\partial \phi_1}{\partial y} \\ \vdots \\ \frac{\partial \phi_N}{\partial x} \\ \frac{\partial \phi_N}{\partial y} \end{bmatrix} + n = \kappa G \frac{\lambda}{2\pi} \cdot \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{M-1} \\ \phi_M \end{bmatrix} + n, \quad (\text{A-55})$$

where (A-6) is invoked to rewrite wavefront into phase, the geometry is assumed symmetric such that $\kappa_x = \kappa_y = \kappa$ and all noise contributions are collected in vector $n \in \mathbb{R}^{2N^2}$. For

readability one can define $S = [\Delta x_1, \Delta y_1, \dots, \Delta x_N, \Delta y_N]^T$, $\phi = [\phi_1, \dots, \phi_N]^T$ and $G = \kappa G \frac{\lambda}{2\pi}$, reducing (A-55) to:

$$S = G\phi + n, \quad (\text{A-56})$$

which reveals clear similarities with (A-42) and thus can be solved with the MVU least squares estimation [12, 17, 21, 48], elaborated in Appendix A-7. This leads to the estimated phase $\tilde{\phi}$.

A-6-1 Implementation with Southwell Geometry

Of the three geometries listed above the Fried and Hudgin models are known to be common and successful in phase reconstruction [12, 17, 34, 42], but it has been proposed that the Southwell geometry outperforms both [12, 42]. The nullspace of each geometry has been investigated in [12], concluding that the Southwell and Hudgin geometries have a nullspace of dimension 1 and the Fried geometry a nullspace of dimension 2. This has the physical meaning that the Fried geometry can detect one mode less than the Southwell and Hudgin models. The remaining nullmode represents piston mode, a mode that no gradient-based WFS can sense [7, 43]. Southwell also proposed in [42] that the Southwell geometry has a smaller error propagation compared to both Fried and Hudgins geometries. Additionally, Southwell posed that the Hudgin model neglects boundary effects since the gradients are placed between phase points. This notion is also observed from illustration Figure A-9. With these arguments in mind the Southwell geometry is chosen to implement the Finite Difference approach.

Assuming one has obtained WFS slope measurements Δ_x and Δ_y , one can rewrite (A-53) as an MVM, by stacking measurements as done in (A-55). If the slopes are represented on an $N \times N$ discretely-sampled grid this stacking results in slope vector $S \in \mathbb{R}^{2N^2 \times 1}$ and phase vector $\phi \in \mathbb{R}^{N^2 \times 1}$. Due to the Southwell geometry the number of phase points is equal to the number of slope measurements. The Southwell data equation can then be written as:

$$\frac{1}{2}G_{grad}S = \frac{\lambda}{2\pi D}G_S\phi + n \quad (\text{A-57})$$

where matrices $G_{grad} \in \mathbb{R}^{2N^2 \times 2N^2}$ and $G_S \in \mathbb{R}^{2N^2 \times N^2}$ implement the geometric relationships on the left-hand and right-hand sides of (A-53). The size of the cells, which is equal to the distance between phase points, is defined by D .

The structure of matrices G_{grad} and G_S depends completely on how directions are defined on the grid. For example, if the x - and y - coordinates are defined positive in direction of the vectors shown in Figure A-9, the matrices are structured as shown in Figure A-10. Before Southwell geometry can be applied one must define the boundary conditions. In Figure A-9 one can identify problems around the borders of the grid, where geometry equation (A-53) refers to the current and a neighbouring phase point, while this neighbour does not exist. One example of this is in the slope in y -direction for the upper-left cell, which relates the current phase point to the one in the cell above that does not exist. The non-existent slope and phase are reflected in G_{grad} and G_S by rows that contain a single entry, as can be seen by the topmost red rectangle in Figure A-10. The rectangle below refers to the same issue of the cell next to the first, since the grid is stacked column-wise. The lowest rectangle refers to the cells in the last column of the grid, that experience the same boundary problem with the gradients in x -direction. These boundary problems are considered to be a disadvantage to a successful reconstruction so rows with a single entry are removed from G_{grad} and G_S . In each

cell on the upper and rightmost border one of such boundary problems arise, meaning that for $2N^2$ slope measurements one discards $2N$ of these. The number of phase points reconstructed does not change, but they are estimated with only 80% or $2N(N-1)$ measurements. The trimmed matrices are shown in Figure A-11.

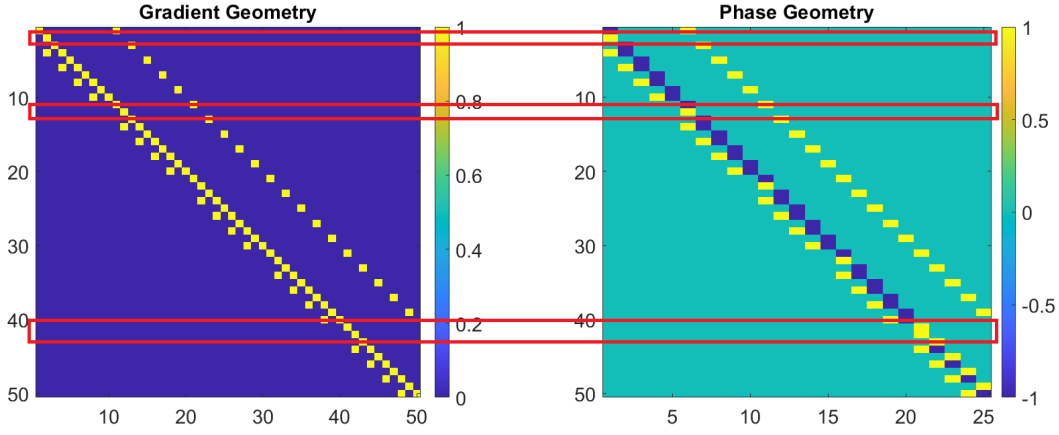


Figure A-10: The structure of the Southwell geometry matrices, shown for a 5×5 grid of phase points. The leftmost figure shows the averaging of neighbouring slopes, performed by G_{grad} in (A-57). The rightmost figure shows the structure of matrix G_S in (A-57) for neighbouring phase points. The red boxes indicate incomplete approximations, where the algorithm refers to non-existent measurements and phase points.

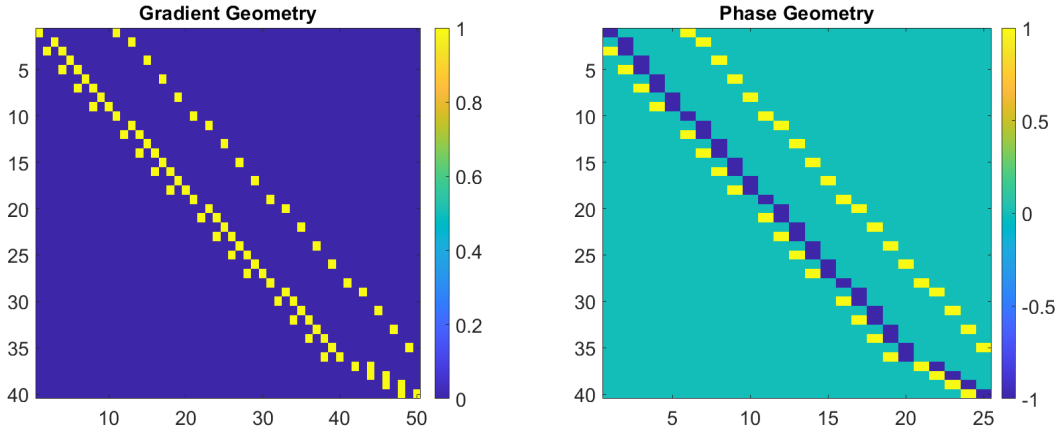


Figure A-11: The structure of the trimmed Southwell geometry matrices, shown for a 5×5 grid of phase points. The leftmost figure shows the averaging of neighbouring slopes, performed by G_{grad} in (A-57). The rightmost figure shows the structure of matrix G_S in (A-57) for neighbouring phase points. All entries that refer to incomplete approximations have been removed, using the remaining 80% of slope measurements.

To verify that this reduction leads to a better result both original and trimmed matrices are used to reconstruct a number of Zernike modes. Using a grid size of $N = 15$ Zernike modes Z_2^{-2} and Z_3^1 are individually simulated and reconstructed. The modes are generated on a square aperture, consistent with how phase screens are generated in this thesis. In order to eliminate detector noise and only test the performance of the matrices, the slopes are

obtained by taking the partial derivatives of the wavefronts, instead of sensing them with the SH or P4F sensor. Results of this verification are shown in Figure A-12 and Figure A-13. The reconstruction using the complete Southwell matrices from Figure A-10 is referred to as the 'Full Reconstruction', whereas the process using trimmed matrices from Figure A-11 is referred to as the 'Reduced Reconstruction'. From left to right one can identify: The ground truth simulated wavefront, the fully reconstructed wavefront, the corresponding full residual wavefront, the reduced reconstruction and finally the reduced residual wavefront.

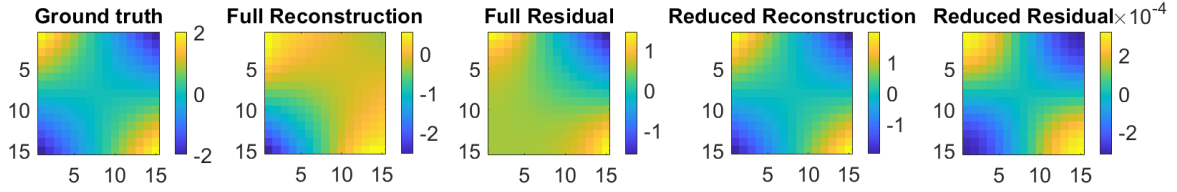


Figure A-12: Reconstruction of the Zernike mode Z_2^{-2} with magnitude 1 on a 15×15 sampled grid.

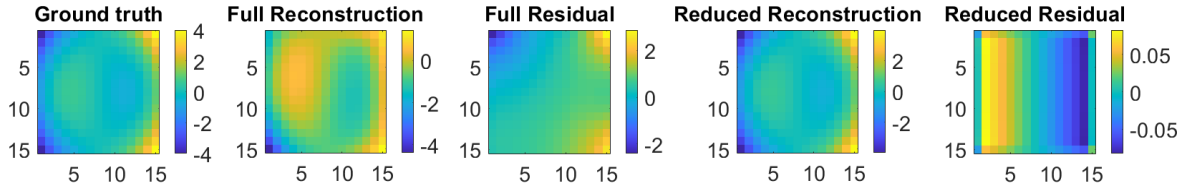


Figure A-13: Reconstruction of the Zernike mode Z_3^1 with magnitude 1 on a 15×15 sampled grid.

These results clearly show that the trimmed matrices result in a better reconstruction and a significantly lower residual wavefront. For example, the 'Full Reconstruction' of Z_2^{-2} results in a $\sim 50\%$ residual wavefront, whereas the trimmed matrices result in a $\sim 0.02\%$ residual. For the Z_3^1 reconstruction the result is a $\sim 60\%$ residual wavefront versus a $\sim 2\%$ residual. The same hypothesis is tested for an increased grid size of $N = 50$. The two wavefronts simulated are Z_3^{-3} with a magnitude of 3 and a superposition of modes Z_2^{-2} , Z_2^0 and Z_3^3 with magnitudes 0.5, 0.5 and 1, respectively. The results shown in Figure A-14 and Figure A-15 indicate that even for a larger grid size and mode superposition the 'Reduced Reconstruction' method significantly improves the performance of the Southwell zonal algorithm.

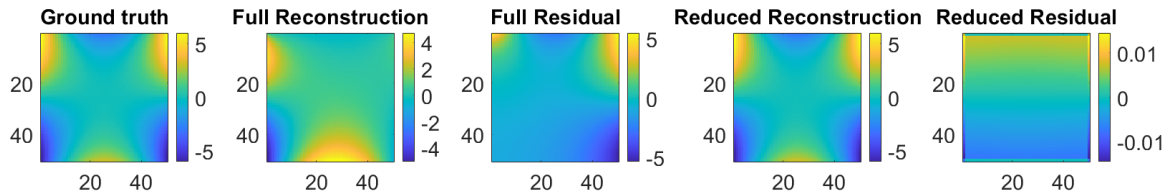


Figure A-14: Reconstruction of the Zernike mode Z_3^{-3} with magnitude 3 on a 50×50 sampled grid.

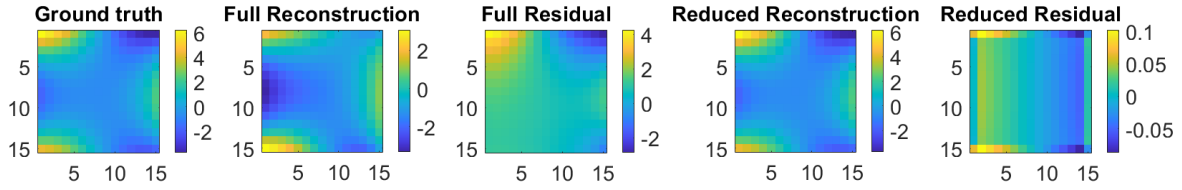


Figure A-15: Reconstruction of the superposition of Zernike modes Z_2^{-2} , Z_2^0 and Z_3^3 with magnitudes 0.5, 0.5 and 1, respectively. The wavefronts and gradients are represented on a 50×50 sampled grid.

Now that the structure of the geometry matrices $G_{grad} \in \mathbb{R}^{2N(N-1) \times 2N^2}$ and $G_S \in \mathbb{R}^{2N(N-1) \times N^2}$ is clearly defined (A-57) can be simplified by introducing $S_a = \frac{1}{2}G_{grad}S$ and $G = \frac{\lambda f}{2\pi D}G_S$:

$$S_a = G\phi + n \quad (\text{A-58})$$

where $S_T \in \mathbb{R}^{2N^2 \times 1}$ contains all averaged and stacked slope measurements and $G \in \mathbb{R}^{2N^2 \times N^2}$ contains the summations of the phase points multiplied with the geometric factors. The simplified systems of equations (A-58) can be solved through a linear least squares minimization, leading to the estimated vectorized phase $\tilde{\phi} \in \mathbb{R}^{N^2 \times 1}$:

$$\tilde{\phi} = C_\phi G^T (G C_\phi G^T + C_n)^{-1} S_T \quad (\text{A-59})$$

where $C_\phi \in \mathbb{R}^{M \times M}$ is the phase covariance matrix, $G \in \mathbb{R}^{2N^2 \times M}$ is the geometry matrix, $C_n \in \mathbb{R}^{2N^2 \times 2N^2}$ the noise covariance matrix and finally $S_T \in \mathbb{R}^{2N^2 \times 1}$ contains the stacked slope measurements. The covariance matrices C_ϕ and C_n impose the stochastic nature of phase ϕ and noise n and must be known before (A-59) can be solved. This a priori knowledge can be found by clearly defining the random nature of the phase to be reconstructed, such as Kolmogorov statistics. For more details on linear least squares minimization, the solution and its proof the reader is referred to Appendix A-7.

A-7 The Minimum-Variance Unbiased Least Squares Estimator

Many systems of equations can be reduced to a MVM such as (A-60), where stacked measurements $S \in \mathbb{R}^{N \times 1}$ are related to unknown quantity $\phi \in \mathbb{R}^{M \times 1}$:

$$S = G\phi + n, \quad (\text{A-60})$$

with $G \in \mathbb{R}^{N \times M}$ the geometry matrix and $n \in \mathbb{M}^{N \times 1}$ possible, though often unknown noise. With both ϕ and n unknown it is impossible to obtain an exact solution for ϕ . A common approach is finding the estimate of ϕ , indicated by $\hat{\phi}$. Considering that additive noise n is unknown and unwanted its effects should be minimized over all measurements. This can be described by minimizing the two-norm of noise n , rewriting (A-60) into a *least squares problem*:

$$\min_{\phi} n^T n, \text{ subject to: } S = G\phi + n. \quad (\text{A-61})$$

The precise process and elaboration is carefully explained in [48]. If noise n is *Gaussian white noise*, meaning it is zero-mean and identity variance, unknown ϕ is zero-mean then the minimum-variance unbiased estimator is found to be:

$$\tilde{\phi} = (G^T G)^{-1} G^T S = G^+ S, \quad (\text{A-62})$$

where G^+ is the Moore-Penrose pseudo-inverse of G , as $G^T G$ is often singular [48]. Note that the estimator in (A-62) is written as $\tilde{\phi}$ instead of the usual estimator notation $\hat{\phi}$. This is done to distinguish between ordinary estimators $\hat{\phi}$ and the MVUE estimator $\tilde{\phi}$. This estimator has the following properties, as put by and proven in [48]:

- Unbiased: The mean value of the estimate equals the mean value of the quantity to be estimated, meaning $E[\tilde{\phi}] = E[\phi] = \bar{\phi}$.
- Minimum-Variance: When compared to other unbiased estimators, the MVUE has the smallest possible variance. This is done by minizing the covariance between estimator and unknown: $\min E[(\tilde{\phi} - \phi)(\tilde{\phi} - \phi)^T]$.

It cannot always be guaranteed that additive noise is Gaussian white noise and its variance may be different from the identity matrix. This is circumvented by the *weighted least squares problem*, that rewrites the noise as a product of a weight matrix and Gaussian white noise:

$$n = L_n \epsilon, \quad \epsilon \sim (0, I), \quad (\text{A-63})$$

where the noise covariance is written as $C_n = L_n L_n^T$. The weighted least squares problem is then formulated as:

$$\min_{\phi} n^T n, \quad \text{subject to: } S = G\phi + L_n \epsilon. \quad (\text{A-64})$$

The solution to the weighted least squares problem starts by formulating linear estimator $\hat{\phi}$ in its most general form, as explained in [48]:

$$\hat{\phi} = MS + N\bar{\phi}, \quad (\text{A-65})$$

where unknown signal ϕ is assumed to have mean value $\bar{\phi}$. By solving both unbiased and minimum-variance properties for M and N , one can obtain the MVUE from this general estimator, detailed below.

Unbiased Linear Estimator The unbiased property dictates that the expected value of the estimator $\hat{\phi}$ should equal the estimated value of the unknown signal. This can be rewritten as:

$$\begin{aligned} E[\phi - \hat{\phi}] &= 0 \\ &= E[\phi - M(G\phi + L_n \epsilon) - N\bar{\phi}] \\ &= (I - MG - N)\bar{\phi} + E[-ML_n \epsilon] \\ &= (I - MG - N)\bar{\phi} = 0, \end{aligned}$$

where $E[\phi] = \bar{\phi}$ and $E[\epsilon] = 0$ since the noise is assumed to be zero mean. In order for estimator $\hat{\phi}$ to be unbiased M and N should be related by:

$$MG + N = I, \quad (\text{A-66})$$

where G was the geometry matrix and I is the identity matrix. Before elaborating the minimum-variance property the Schur complement is explained as it will prove a useful tool.

Schur Complement The Schur complement allows decomposition of matrices to simplify matrix-vector multiplications. If a matrix M can be partitioned as follows:

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \quad (\text{A-67})$$

with $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{m \times n}$ and $D \in \mathbb{R}^{m \times m}$. Matrix M can then it be decomposed into:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I_n & BD^{-1} \\ 0 & I_m \end{bmatrix} \begin{bmatrix} A - BD^{-1}C & 0 \\ 0 & D \end{bmatrix} \begin{bmatrix} I_n & 0 \\ D^{-1}C & I_m \end{bmatrix}, \quad (\text{A-68})$$

where I_n and I_m are an $n \times n$ and $m \times m$ matrices with ones on the diagonal. Decomposition (A-68) is valid as long as matrix D is invertible or in other words *full rank*. If this is not the case but matrix A is invertible, another decomposition is possible:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I_n & 0 \\ CA^{-1} & I_m \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & D - CA^{-1}B \end{bmatrix} \begin{bmatrix} I_n & A^{-1}B \\ 0 & I_m \end{bmatrix}. \quad (\text{A-69})$$

Minimum-Variance Unbiased Estimator For brevity the estimation error between ϕ and its unbiased estimator can be rewritten as:

$$\begin{aligned} \phi - \hat{\phi} &= (I - MG)\phi - N\bar{\phi} - ML_n\epsilon \\ &= (I - MG)(\phi - \bar{\phi}) - ML_n\epsilon, \end{aligned}$$

where $N = I - MG$ (A-66) is invoked. In order to minimize the variance of this estimation error the covariance matrix of the error is to be computed. For brevity matrix $C_\phi = \mathbb{E}[(\phi - \bar{\phi})(\phi - \bar{\phi})^T]$ refers to the covariance matrix of ϕ . It is assumed that this matrix is positive definite. The estimation error covariance matrix is written as:

$$\begin{aligned} \mathbb{E}[(\phi - \hat{\phi})(\phi - \hat{\phi})^T] &= \mathbb{E}[(I - MG)(\phi - \bar{\phi})(\phi - \bar{\phi})^T(I - MG)^T + ML_n(\epsilon\epsilon^T)L_n^T M^T] \\ &= (I - MG)C_\phi(I - MG)^T + MC_nM^T, \end{aligned}$$

where $\mathbb{E}[\epsilon\epsilon^T] = I$ and it is assumed that noise ϵ and signal ϕ are uncorrelated, meaning any term involving $\mathbb{E}[\phi\epsilon^T]$ or $\mathbb{E}[\epsilon\phi^T]$ is equal to zero. By completion of squares it is possible to separate terms dependent and independent of M :

$$\mathbb{E}[(\phi - \hat{\phi})(\phi - \hat{\phi})^T] = \begin{bmatrix} I_n & -M \end{bmatrix} \begin{bmatrix} C_\phi & C_\phi G^T \\ GC_\phi & GC_\phi G^T + C_n \end{bmatrix} \begin{bmatrix} I_n \\ -M^T \end{bmatrix}. \quad (\text{A-70})$$

The noise covariance matrix $C_n = L_n L_n^T$ is positive definite so square matrix $GC_\phi G^T + C_n$ is positive definite as well. This means that the Schur complement (A-68) can be used to decompose the central matrix. To keep the elaboration clear the Schur decomposition of the central matrix is written in its partitions:

$$\begin{bmatrix} C_\phi & C_\phi G^T \\ GC_\phi & GC_\phi G^T + C_n \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I_n & BD^{-1} \\ 0 & I_m \end{bmatrix} \begin{bmatrix} A - BD^{-1}C & 0 \\ 0 & D \end{bmatrix} \begin{bmatrix} I_n & 0 \\ D^{-1}C & I_m \end{bmatrix}. \quad (\text{A-71})$$

The estimate error covariance matrix can then be rewritten as:

$$\begin{aligned}
E[(\phi - \hat{\phi})(\phi - \hat{\phi})^T] &= \begin{bmatrix} I_n & -M \end{bmatrix} \begin{bmatrix} I_n & BD^{-1} \\ 0 & I_m \end{bmatrix} \begin{bmatrix} A - BD^{-1}C & 0 \\ 0 & D \end{bmatrix} \begin{bmatrix} I_n & 0 \\ D^{-1}C & I_m \end{bmatrix} \begin{bmatrix} I_n \\ -M^T \end{bmatrix} \\
&= \begin{bmatrix} I_n & BD^{-1} - M \end{bmatrix} \begin{bmatrix} A - BD^{-1}C & 0 \\ 0 & D \end{bmatrix} \begin{bmatrix} I_n \\ D^{-1}C - M^T \end{bmatrix} \\
&= \begin{bmatrix} A - BD^{-1}C & (BD^{-1} - M)D \end{bmatrix} \begin{bmatrix} I_n \\ D^{-1}C - M^T \end{bmatrix} \\
&= A - BD^{-1}C + (BD^{-1} - M)D(D^{-1}C - M^T).
\end{aligned}$$

Compared to (A-70) the above elaboration does not look any simpler or clearer, but the contrary is true. The goal was to minimize the variance as a function of M and N , meaning that the term $A - BD^{-1}C$ is not affecting the minimization. The goal is thus reduced to:

$$\min_{M,N} E[(\phi - \hat{\phi})(\phi - \hat{\phi})^T] = \min_M (BD^{-1} - M)D(D^{-1}C - M^T). \quad (\text{A-72})$$

Considering that term $D = GC_\phi G^T + C_n$ is positive definite, the value of (A-72) will be positive definite as well and thus minimization is accomplished by setting (A-72) to zero. One possible solution is $M = BD^{-1}$. In terms of the original problem (A-70) this leads to:

$$M = C_\phi G^T (GC_\phi G^T + C_n)^{-1}. \quad (\text{A-73})$$

Invoking the unbiased relation (A-66) one can find the solution for matrix N :

$$N = I - MG = I - C_\phi G^T (GC_\phi G^T + C_n)^{-1} G. \quad (\text{A-74})$$

The results for M and N lead to the MVU estimator $\tilde{\phi}$ for the weighted least squares problem, based on the general estimator $\hat{\phi}$ (A-65). One final simplification can be done by assuming unknown signal ϕ to be zero mean. Then the weighted least squares MVUE reduces to:

$$\tilde{\phi} = C_\phi G^T (GC_\phi G^T + C_n)^{-1} S. \quad (\text{A-75})$$

The assumption $\bar{\phi} = 0$ can be physically supported in the framework of wavefront reconstruction. As mentioned in Section A-6-1 and in [5, 43] the mean value of the phase to be reconstructed can be attributed to the Zernike piston mode. This mode cannot be sensed by gradient-based wavefront sensors, such as the SH and plenoptic sensors and is thus left out of any modal reconstruction scheme [7, 43]. For the same reasons it is left out of zonal reconstruction schemes, as it simply cannot be sensed and thus impossible to reconstruct. This also means that the reconstructed phase $\tilde{\phi}$ is zero-mean.

Estimation of Covariance from Data The MVUE is defined using covariance matrices for phase aberration ϕ and noise contribution n , but these matrices are generally not known. The default values used in this thesis, implemented in *getphase.m*, are set the $C_\phi = I$ and $C_n = 0.001 \cdot I$, where I is the identity matrix. Naturally, this default value is not optimal so one can attempt to optimize these default values. Chapter 4 detailed how the structure function

of Kolmogorov turbulence can lead to an expression for the phase covariance matrix. By determining the Kolmogorov power spectral density (PSD), which was presented in multiple forms by Schmidt in [39], the phase covariance matrix can be found through an inverse Fourier transform.

This leaves the determination of noise covariance C_n , which depends on the imaging conditions, hardware used and reconstruction algorithms. The following method is written to determine all necessary covariance matrices, to provide an alternative to inverse Fourier transforms of the Kolmogorov PSD. The most straightforward method is obtaining open-loop data, either experimentally or through simulations, and approximate the covariance matrices by:

$$C_\phi \approx \frac{\phi\phi^T}{M}, \quad (\text{A-76})$$

where $\phi \in \mathbb{R}^{N^2 \times M}$ contains M independent measurements of each phase, that is represented on an $N \times N$ discretely sampled grid and subsequently vectorized to an $N^2 \times 1$ vector. In order to obtain an expression of the covariance matrices an analysis of the modal and zonal data equations is performed. Without considering noise these equations can be written as:

$$\begin{aligned} S &= G_\perp \gamma & (\text{modal reconstruction}), \\ S &= G\phi & (\text{zonal reconstruction}), \end{aligned} \quad (\text{A-77})$$

which can be solved by the ordinary least squares estimator (A-62). The simulation framework discussed in this chapter does not include any additional noise so one can assume that any noise originates from the obtained slope measurements coined as *detector noise*. It can be included in (A-77) by expanding $S = S_0 + \epsilon$, where S_0 is the true slope and ϵ captured detector noise. Through noise in the slopes the phase reconstruction will contain additive noise as well:

$$\begin{aligned} S_0 + \epsilon &= G_\perp (\gamma + n) & (\text{modal reconstruction}), \\ S_0 + \epsilon &= G (\phi + n) & (\text{zonal reconstruction}), \end{aligned} \quad (\text{A-78})$$

where γ and ϕ represent the true and noiseless reconstruction results. For the purpose of these simulations one can interpret noise ϵ to consist of two elements: (1) a truncation error due to representing the phase gradient by a discretely sampled grid and (2) a mismatch error due to the discrete Fourier transform, where the images are shifted by a pixel or two. Despite equal notation and quantification, the reconstruction noises n in both equations of (A-78) are not equal. One can isolate both noise contributions in (A-78) following the reasoning of error propagation discussed in [21, 23]. Considering that true slope measurements S_0 lead to true reconstructions γ and ϕ it follows from linearity that ϵ leads to error n :

$$\begin{aligned} \epsilon_{\text{modal}} &= G_\perp n, \\ \epsilon_{\text{zonal}} &= Gn, \end{aligned} \quad (\text{A-79})$$

and the reconstruction errors n can be found by the ordinary least squares estimator (A-62):

$$\begin{aligned} \tilde{n}_{\text{modal}} &= (G_\perp^T G_\perp)^{-1} G_\perp^T \epsilon_{\text{modal}} = G_\perp^+ \epsilon_{\text{modal}}, \\ \tilde{n}_{\text{zonal}} &= (G^T G)^{-1} G^T \epsilon_{\text{zonal}} = G^+ \epsilon_{\text{zonal}}, \end{aligned} \quad (\text{A-80})$$

where G_{\perp}^{+} is the pseudo-inverse of the orthonormal Zernike gradient matrix (A-44) and G^{+} the pseudo-inverse of the zonal geometry matrix. Using (A-76) the estimate for the reconstruction noise covariance can be expressed as a function of the detector error covariance C_{ϵ} :

$$\begin{aligned} C_n^{modal} &\approx \frac{1}{K} \tilde{n}_{modal} \tilde{n}_{modal}^T = G_{\perp}^{+} C_{\epsilon}^{modal} \left(G_{\perp}^{+}\right)^T, \\ C_n^{zonal} &\approx \frac{1}{K} \tilde{n}_{zonal} \tilde{n}_{zonal}^T = G^{+} C_{\epsilon}^{zonal} \left(G^{+}\right)^T, \end{aligned} \quad (\text{A-81})$$

where K is the number of independent measurements, such that $\tilde{n} \in \mathbb{R}^{N^2 \times K}$. Covariance matrix C_{ϵ} can be found using open-loop data, comparing the slopes of the reference phase to the detected slopes. Expression (A-81) solves the first half of the covariance estimations. For the zonal reconstruction model the second half is simply found by computing the covariance matrix of the open-loop reference phases using (A-76). The modal reconstruction requires knowledge of C_{γ} instead of C_{ϕ} , so an additional step is required that starts with the phase data equation from (A-45):

$$\begin{aligned} \phi &= \frac{2\pi}{\lambda} Z_{\perp} \alpha + n \\ &= \frac{2\pi}{\lambda} Z_{\perp} \beta_Z \beta_G^{-1} \gamma + n, \end{aligned}$$

where $\phi \in \mathbb{R}^{N^2 \times K}$. By invoking (A-76) one can express the phase covariance matrix as a function of the coefficient covariance C_{γ} . By recognising $Z_{\perp}^T Z_{\perp} = I$ and rewriting $\beta = \beta_Z \beta_G^{-1}$, such that $\beta^{-T} = \beta_G^T \left(\beta_Z^T\right)^{-1}$, one can isolate C_{γ} :

$$\begin{aligned} C_{\phi} &\approx \frac{1}{K} \phi \phi^T = \frac{1}{K} \left(\frac{2\pi}{\lambda}\right)^2 Z_{\perp} \beta_Z \beta_G^{-1} \gamma \gamma^T \left(Z_{\perp} \beta_Z \beta_G^{-1}\right)^T, \\ C_{\gamma} &= \frac{K \lambda^2}{4\pi^2} Z_{\perp}^T \beta^{-1} C_{\phi} \beta^{-T} Z_{\perp}. \end{aligned} \quad (\text{A-82})$$

Bibliography

- [1] Y. A. Adelson, Edward H.; Wang. Single Lens Stereo with a Plenoptic Camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):99 – 106, 1992.
- [2] Vyas Akondi, Sara Castillo, and Brian Vohnsen. Digital pyramid wavefront sensor with tunable modulation. *Optics Express*, 21(15):27–38, 2013. doi:10.1364/OE.21.018261.
- [3] All About Vision. Common shapes of aberrations created when a wavefront of light passes through eyes with imperfect vision. *Image obtained from: "https://www.allaboutvision.com/eye-exam/wavefront.htm" on 16/06/2020*, 2018. URL: <https://www.allaboutvision.com/eye-exam/wavefront.htm>.
- [4] Amit Ashok and Mark A. Neifeld. Compressive light field imaging. *Three-Dimensional Imaging, Visualization, and Display 2010 and Display Technologies and Applications for Defense, Security, and Avionics IV*, 7690:76900Q, 2010. doi:10.1117/12.852738.
- [5] University of Colorado Boulder. Zernike Polynomials. In *Wavefront Sensing*, pages 1–7, Boulder, Colorado, 2014. URL: <http://ecee.colorado.edu/ecen5606/2014/Wavefront{ }SensingAOL2014.pdf>, doi:10.1117/3.923078.ch13.
- [6] M.C. Britton. The Anisoplanatic Point - Spread Function in Adaptive Optics. *Astronomical Society of the Pacific*, 118(844):885–900, 2006. URL: <https://www.jstor.org/stable/10.1086/505547TheAnisoplanaticPoint-SpreadFunctioninAdaptiveOpti>.
- [7] Gary Chanan. Principles of Wavefront Sensing and Reconstruction. *Center for Adaptive Optics (CfAO) Proceedings: Summer School on Adaptive Optics*, pages 5–40, 2000. URL: <http://cfao.ucolick.org/aosummer/book/pdf/1.1{ }chanan.pdf>.
- [8] Mo Chen. Gram Schmidt Algorithms. *Code downloaded from " Mo Chen (2021). Gram-Schmidt orthogonalization (https://www.mathworks.com/matlabcentral/fileexchange/55881-gram-schmidt-orthogonalization), MATLAB Central File Exchange. Retrieved January 4, 2021." on 04-01-2021*, mar 2016.

- [9] Theam Yong Chew, Richard M. Clare, and Richard G. Lane. A comparison of the Shack-Hartmann and pyramid wavefront sensors. *Optics Communications*, 267(3):189–195, 2006. doi:[10.1016/j.optcom.2006.07.011](https://doi.org/10.1016/j.optcom.2006.07.011).
- [10] Richard M. Clare and Richard G. Lane. Wave-front sensing from subdivision of the focal plane with a lenslet array. *Journal of the Optical Society of America A*, 22(1):117, 2005. doi:[10.1364/josaa.22.000117](https://doi.org/10.1364/josaa.22.000117).
- [11] Marcos V A N Dam, Richard Clare, W M Keck Observatory, Marcos Van Dam, and Richard Clare. Wavefront sensing for adaptive optics.
- [12] Cornelis C. de Visser and Michel Verhaegen. Wavefront reconstruction in adaptive optics systems using nonlinear multivariate splines. *Journal of the Optical Society of America A*, 30(1):82, 2013. doi:[10.1364/josaa.30.000082](https://doi.org/10.1364/josaa.30.000082).
- [13] Jeffery S. Dennison and Jason D. Schmidt. Simulating the effects of an extended source on the Shack-Hartmann wavefront sensor through turbulence. *IEEE Aerospace Conference Proceedings*, 2011. doi:[10.1109/AERO.2011.5747385](https://doi.org/10.1109/AERO.2011.5747385).
- [14] Edmund Optics. Specifications of the Edmund Optics MLA #64-478. Website: "<https://www.edmundoptics.com/p/microlens-array-10-x-10mm-300um-pitch-05deg-divergence/19190/>". Accessed on 18/08/2020., 2020.
- [15] Adly A. Girgis and Fredric M. Ham. A Quantitative Study of Pitfalls in the FFT. *IEEE Transactions on Aerospace and Electronic Systems*, AES-16(4):434–439, 1980. doi:[10.1109/TAES.1980.308971](https://doi.org/10.1109/TAES.1980.308971).
- [16] J. W. Goodman. *Introduction to Fourier Optics*. McGraw-Hill, 2 edition, 1996.
- [17] Karel Hinnen. *Data-Driven Optimal Control for Adaptive Optics*. PhD Dissertation, Delft University of Technology, 2007. URL: <http://resolver.tudelft.nl/uuid:4f6533b5-eb0a-44af-9e9a-e3252b49f30e>, doi:[10.1109/TAES.1980.308971](https://doi.org/10.1109/TAES.1980.308971).
- [18] Jonathan Ko and Christopher C. Davis. Comparison of the plenoptic sensor and the Shack-Hartmann sensor. *Applied Optics*, 56(13):3689, 2017. doi:[10.1364/ao.56.003689](https://doi.org/10.1364/ao.56.003689).
- [19] Charl Laäs. Wavefront from a point-source. Image obtained from: "<https://www.charllaas.com/new-eye-test-technology/>" on 15/02/2020, 2011. URL: <https://www.charllaas.com/new-eye-test-technology/>.
- [20] R. G. Lane, A. Glindemann, and J. C. Dainty. Simulation of a Kolmogorov phase screen. *Waves in Random Media*, 2(3):209–224, 1992. doi:[10.1088/0959-7174/2/3/003](https://doi.org/10.1088/0959-7174/2/3/003).
- [21] R G Lane and M Tallon. Wave-front reconstruction using a Shack-Hartmann sensor. *Applied Optics*, 31(32):6902–6908, 1992. URL: <https://doi.org/10.1364/AO.31.006902>.
- [22] Olivier Lardière, John Pazder, Jean Pierre Véran, and Maaïke Van Kooten. Double-pyramid wavefront sensors: Tolerance relaxation and cheaper alternatives using achromatic double-roof prisms. *Adaptive Optics for Extremely Large Telescopes, 2017 AO4ELT5*, 2017-June(January), 2017. doi:[10.26698/ao4elt5.0042](https://doi.org/10.26698/ao4elt5.0042).

-
- [23] Mengyang Li, Dahai Li, Chen Zhang, Kewei E, Qionghua Wang, and Haiping Chen. Modal wavefront reconstruction from slope measurements for rectangular apertures. *Journal of the Optical Society of America A*, 32(11):1916, 2015. doi:[10.1364/josaa.32.001916](https://doi.org/10.1364/josaa.32.001916).
 - [24] Xinyang Li and Wenhan Jiang. Comparing zonal reconstruction algorithms and modal reconstruction algorithms in adaptive optics system. *High-Resolution Wavefront Control: Methods, Devices, and Applications IV*, 4825(November 2002):121, 2002. doi:[10.1117/12.451985](https://doi.org/10.1117/12.451985).
 - [25] O Manneberg. Design and Simulation of a High Spatial Resolution Hartmann-Shack Wavefront Sensor. Technical report, Royal Institute of Technology Stockholm, Stockholm, Sweden, 2005.
 - [26] Steffen Mauch and Johann Reger. Real-time spot detection and ordering for a shack-hartmann wavefront sensor with a low-cost FPGA. *IEEE Transactions on Instrumentation and Measurement*, 63(10):2379–2386, 2014. doi:[10.1109/TIM.2014.2310616](https://doi.org/10.1109/TIM.2014.2310616).
 - [27] C. Max. Introduction to adaptive optics and its history, 2008.
 - [28] Alastair D. McAulay. Generating Kolmogorov phase screens for modeling optical turbulence. *Laser Weapons Technology*, 4034(July 2000):50, 2000. doi:[10.1117/12.391877](https://doi.org/10.1117/12.391877).
 - [29] Merriam-Webster. Principal ray. *Dictionary definition, obtained from "https://www.merriam-webster.com/dictionary/principal ray" on 13/05/2020*, 2020. URL: <https://www.merriam-webster.com/dictionary/principalray>.
 - [30] Ren Ng, Stanford Tech, Marc Levoy, Report Ctsr, Gene Duval, Mark Horowitz, and Pat Hanrahan. Light Field Photography with a Hand-held Plenoptic Camera. *Stanford Tech Report CTSR*, 02(June):1–11, 2005.
 - [31] Robert J Noll. Zernike polynomials and atmospheric turbulence*. *Optical Society of America*, 66(3):207–211, 1976.
 - [32] Jiang Pengzhi, Xu Jieping, Liang Yonghui, and Mao Hongjun. Comparison of the Shack-Hartmann and plenoptic sensor in closed-loop adaptive optics system. *Optical Engineering*, 55(3):033105, 2016. URL: <https://www.spiedigitallibrary.org/journals/optical-engineering/volume-55/issue-03/033105/Comparison-of-the-ShackHartmann-and-plenoptic-sensor-in-closed-loop/10.1117/1.OE.55.3.033105.full?SS0=1>, doi:[10.1117/1.OE.55.3.033105](https://doi.org/10.1117/1.OE.55.3.033105).
 - [33] Paolo Pozzi, Carlas Smith, Dean Wilding, Oleg Soloviev, Martin Booth, Gleb Vdovin, and Michel Verhaegen. Anisoplanatic adaptive optics in parallelized laser scanning microscopy, 2018. URL: <http://arxiv.org/abs/1809.07529>, arXiv:1809.07529.
 - [34] Swapnil Prabhudesai. Adaptive Optics Reconstruction Methods, 2011.
 - [35] R. Ragazzoni and J. Farinato. Sensitivity of a pyramidic Wave Front sensor in closed loop Adaptive Optics. *Astronomy and Astrophysics*, 350:L23–L26, 1999.

- [36] J.M. Rodríguez, B. Femenía, I. Montilla, L.F. Rodríguez-Ramos, J.G. Marichal-Hernández, J.P. Lüke, R. López, J.J. Díaz, and Y. Martín. The CAFADIS camera: a new tomographic wavefront sensor for Adaptive Optics. *Adaptive Optics for Extremely Large Telescopes*, 05011(1):05011, 2010. doi:[10.1051/ao4elt/201005011](https://doi.org/10.1051/ao4elt/201005011).
- [37] J. M. Rodríguez-Ramos, B. Femenía Castellá, F. Pérez Nava, and S. Fumero. Wavefront and distance measurement using the CAFADIS camera. *Adaptive Optics Systems*, 7015(July 2014):70155Q, 2008. doi:[10.1117/12.789380](https://doi.org/10.1117/12.789380).
- [38] J. M. Rodríguez-Ramos, E. Magdaleno Castelló, C. Domínguez Conde, M. Rodríguez Valido, and J. G. Marichal-Hernández. 2D-FFT implementation on FPGA for wavefront phase recovery from the CAFADIS camera. *Adaptive Optics Systems*, 7015:701539, 2008. doi:[10.1117/12.789312](https://doi.org/10.1117/12.789312).
- [39] Jason D. Schmidt. Numerical Simulation of Optical Wave Propagation. Technical Report 1, Society of Photo-Optical Instrumentation Engineers, Bellingham, Washington, 2010.
- [40] Iuliia Shatokhina. *Fast Wavefront Reconstruction Algorithms for eXtreme Adaptive Optics*. Dissertation, Johannes Kepler Universität Linz, 2014.
- [41] H. Song, R. Fraanje, G. Schitter, H. Kroese, G. Vdovin, and M. Verhaegen. Model-based aberration correction in a closed-loop wavefront-sensor-less adaptive optics system. *Optics Express*, 18(23):24070, 2010. doi:[10.1364/oe.18.024070](https://doi.org/10.1364/oe.18.024070).
- [42] W. H. Southwell. Wave-Front Estimation From Wave-Front Slope Measurements. *Journal of the Optical Society of America*, 70(8):998–1006, 1980. doi:[10.1364/JOSA.70.000998](https://doi.org/10.1364/JOSA.70.000998).
- [43] W Travis. *Wavefront reconstruction and mirror surface optimization for adaptive optics*. Dissertation, Naval Postgraduate School, 2014.
- [44] UK ELT Team. The Extremely Large Telescope. Image obtained from: "<https://www.elt-uk.org/>" on 12/01/2020, 2020. URL: <https://www.elt-uk.org/>.
- [45] P. P. Vaidyanathan. Generalizations of the sampling theorem: Seven decades after Nyquist. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 48(9):1094–1109, 2001. doi:[10.1109/81.948437](https://doi.org/10.1109/81.948437).
- [46] Wout van de Ketterij. *Blind Deconvolution of Anisoplanatic Aberrations*. Msc thesis, Delft University of Technology, 2019.
- [47] Gleb Vdovin, Oleg Soloviev, and Mikhail Loktev. Plenoptic wavefront sensor with scattering pupil. *Optics Express*, 22(8):9314, 2014. doi:[10.1364/oe.22.009314](https://doi.org/10.1364/oe.22.009314).
- [48] M Verhaegen and V Verdult. *Filtering and System Identification: A Least Squares Approach*. Cambridge University Press, 2007. URL: <https://books.google.de/books?id=6Ne76uY01VwC>.
- [49] Michel Verhaegen, Paulo Pozzi, Oleg Soloviev, Gleb Vdovin, and Dean Wilding. Control for High Resolution Imaging, 2017.

-
- [50] Wikimedia Commons. A representation of a plane wave's electric field shown from an oblique angle. *Image obtained from "https://commons.wikimedia.org/wiki/File:Plane_Wave_Oblique_View.jpg" on 08/11/2020*, 2010. URL: https://commons.wikimedia.org/wiki/File:Plane_Wave_Oblique_View.jpg.
- [51] Chensheng Wu. *The plenoptic sensor*. PhD Dissertation, University of Maryland, College Park, 2017. doi:10.13016/M20R54.
- [52] Chensheng Wu, Jonathan Ko, and Christopher C. Davis. Determining the phase and amplitude distortion of a wavefront using a plenoptic sensor. *Journal of the Optical Society of America A*, 32(5):964, 2015. doi:10.1364/josaa.32.000964.
- [53] Chensheng Wu, Jonathan Ko, and Christopher C. Davis. Complex wavefront sensing with a plenoptic sensor. *Laser Communication and Propagation through the Atmosphere and Oceans V*, 9979(September 2016):99790Y, 2016. doi:10.1117/12.2239117.
- [54] Tsai-wei Wu and Meng Cui. Numerical study of multi-conjugate large area wavefront correction for deep tissue microscopy. *Optics Express*, 23(6):7463, 2015. arXiv:arXiv:1501.02685, doi:10.1364/oe.23.007463.

Glossary

List of Acronyms

AO	Adaptive Optics
DFT	discrete Fourier transform
DM	deformable mirror
FFT	Fast Fourier Transform
FOV	field of view
FPGA	Field Programmable Gates Array
GPU	Graphics Processing Unit
MCAO	Multi-Conjugate Adaptive Optics
MLA	microlens array
MMSE	minimum mean square error
MVM	matrix-vector multiplication
MPI	multiple point source imaging
MVU	minimum-variance unbiased
MVUE	minimum-variance unbiased estimate
OBT	object-based tilt
OTF	optical transfer function
P3F	plenoptic 3F
P4F	plenoptic 4F
P4FSH	Plenoptic 4F Shack-Hartmann hybrid
PSD	power spectral density
PSF	point-spread function
PWZM	piece-wise zero-mean
RMS	root mean square
RMSE	root-mean-square
SH	Shack-Hartmann
SNR	signal-to-noise ratio
WFS	wavefront sensor