

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

MASTER THESIS

Evaluation of Optical Aberrations using Phase Diversity

Author:
Jordan VOIRIN

Supervisors:
Dr. Laurent JOLISSAINT
Dr. Jean-Paul KNEIB

*A thesis submitted in fulfillment of the requirements
for the degree of Master in Applied Physics*

in the

Astrophysics laboratory
Basic Sciences

February 26, 2018

Declaration of Authorship

I, Jordan VOIRIN, declare that this thesis titled, "Evaluation of Optical Aberrations using Phase Diversity" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

“For the sake of persons of ... different types, scientific truth should be presented in different forms, and should be regarded as equally scientific, whether it appears in the robust form and the vivid coloring of a physical illustration, or in the tenuity and paleness of a symbolic expression.”

James Clerk Maxwell

Acknowledgements

The acknowledgments and the people to thank go here, don't forget to include your project advisor...

Contents

Declaration of Authorship	iii
Acknowledgements	vii
Introduction	1
1 Theoretical background	3
1.1 Scalar Diffraction Theory	3
1.1.1 Scalar Field and Helmholtz equation	3
1.1.2 Rayleigh-Sommerfeld integral	3
1.1.3 Fresnel approximation	4
1.1.4 Fraunhofer approximation	5
1.1.5 Converging lens introduction	5
1.2 Imaging system	6
1.2.1 Impulse Response (IR)	6
1.2.2 Optical Transfer Function (OTF)	6
1.2.3 From Object to Image	8
1.3 Aberrations	9
1.3.1 Sources of aberration	9
1.3.2 Zernike polynomials	10
1.4 Phase retrieval	12
1.4.1 Shack-Hartmann	13
2 Phase Diversity	15
3 Phase Diversity Experiment	17
3.1 Experimental Setup	17
3.1.1 Light source	18
3.1.2 Entrance pupil	18
3.1.3 Pupil imaging system	18
3.1.4 Detectors	19
3.2 Data Acquisition	19
3.2.1 Ximea Camera	19
3.2.2 Shack-Hartman WFS	19
3.3 Results	19
3.3.1 Parallel plane plate	20
A Optical Component Datasheets	21
A.1 Pigtailed laser diode	22
A.1.1 Power supply modification	23
A.2 Converging lens A220TM-A, $f = 11$ mm	23
A.3 Pinhole $10\ \mu\text{m}$	24
A.4 Converging lens AL100200, $f = 200$ mm	25

A.5	Converging lens AC254-100-A, $f = 100$ mm	26
A.6	Ximea Camera, MQ013MG-E2	27
A.7	Shack-Hartmann wavefront sensor, WFS150-5C	28
B	Python Code	29
B.1	AlignementScriptXimeaCamera.py	29
B.2	AcquisAndSaveXimea.py	32
	Bibliography	35

List of Figures

1.1	Diffraction Schemas	4
1.2	Schema of a imaging instrument, (Goodman, 1988, Chapter 6.1)	6
1.3	PSF of a perfect imaging system composed by a 3.6 mm pupil and a focal length of 80 mm at a wavelength of 637.5 nm. The size, N, of the PSF is 400 and the pixel size is 5.3 μm	7
1.4	OTF of a perfect imaging system composed by a 3.6 mm pupil and a focal length of 80 mm at a wavelength of 637.5 nm.	7
1.6	Comparison of perfect PSF and PSF with aberrations of an imaging system composed by a 3.6 mm pupil and a focal length of 80 mm at a wavelength of 637.5 nm. The size, N, of the PSF is 400 and the pixel size is 5.3 μm	9
1.5	Gaussian reference sphere vs. aberrated Wavefront	9
1.7	Representation of the 21 st Zernike polynomials	11
1.8	Shack-Hartmann principle	12
3.1	Experimental Setup Schema	17
3.3	Source schema and pinhole effect on the beam.	18
3.2	Wavefront curvature	18
3.4	PSFs example of an alignment procedure	19

List of Tables

3.1 Optical Components	17
----------------------------------	----

Introduction

???

Chapter 1

Theoretical background

In this chapter, we will present the theory upon which this work is based. First, the light propagation formalism will be reminded through the scalar diffraction theory based on the Goodman (1988). Then we will describe in general an imaging system and its properties. And finally we will discuss the wavefront aberration theory.

1.1 Scalar Diffraction Theory

1.1.1 Scalar Field and Helmholtz equation

A monochromatic wave, at position P and time t , can be represented by a scalar field $u(P, t)$ written as :

$$u(P, t) = A(P) \exp[-j2\pi\nu t + j\phi(P)], \quad (1.1)$$

where $A(P)$ and $\phi(P)$ are the amplitude and phase, respectively, of the wave at position P and ν is the wave frequency.

The spatial part of eqt. (1.1), also called phasor in the literature,

$$U(P) = A(P) e^{j\phi(P)}, \quad (1.2)$$

must verify the Helmholtz equation :

$$(\nabla^2 + k^2)U = 0, \quad (1.3)$$

where k is the wave number given by

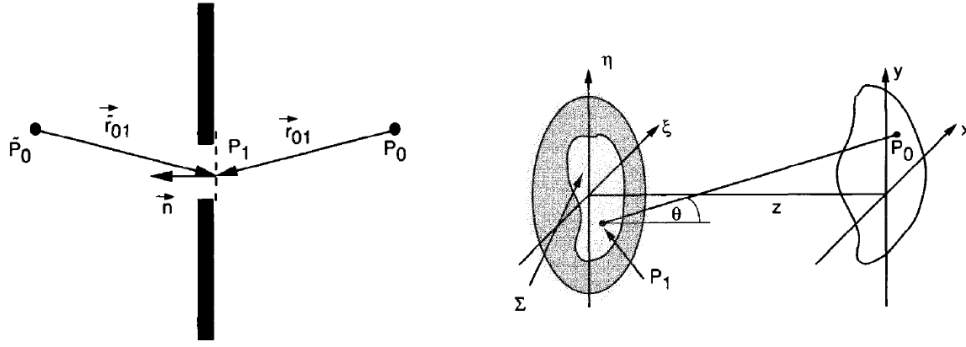
$$k = 2\pi n \frac{\nu}{c} = \frac{2\pi}{\lambda}, \quad (1.4)$$

and λ is the wavelength in the dielectric medium.

1.1.2 Rayleigh-Sommerfeld integral

Rayleigh and Sommerfeld developed a formalism using the Helmholtz equation and Green's Theorem to compute the induced diffraction by a plane screen. Let's suppose that we have a monochromatic source at \tilde{P}_0 on the left of a plane screen with aperture Σ , the Rayleigh-Sommerfeld formula allows to compute the complex amplitude at P_0 on the right of the plane screen (see Figure 1.1a).

$$U(P_0) = \frac{1}{j\lambda} \iint_{\Sigma} U'(P_1) \frac{\exp(jkr_{01})}{r_{01}} \cos(\mathbf{n}, \mathbf{r}_{01}) d\mathbf{s} \quad (1.5)$$



(A) Rayleigh-Sommerfeld formulation of diffraction by a plane screen, (Goodman, 1988, Chapter 3.5). (B) Diffraction geometry, (Goodman, 1988, Chapter 4.1).

FIGURE 1.1: Diffraction Schemas

$U'(P_1)$ is the complex amplitude on the screen, $\cos(\mathbf{n}, \mathbf{r}_{01})$ is the cosine of the angle between the aperture plane normal toward the source and the vector $\mathbf{r}_{01} = \mathbf{P}_0\mathbf{P}_1$ given by

$$r_{01} = \sqrt{z^2 + (x - \xi)^2 + (y - \eta)^2}. \quad (1.6)$$

We can rewrite eqt. (1.5) using $\cos(\mathbf{n}, \mathbf{r}_{01}) = \cos(\theta) = \frac{z}{r_{01}}$ and the coordinate systems (ξ, η) and (x, y) , see Figure 1.1b,

$$U(x, y) = \frac{z}{j\lambda} \iint_{-\infty}^{\infty} U(\xi, \eta) \frac{\exp(jkr_{01})}{r_{01}^2} d\xi d\eta. \quad (1.7)$$

We can integrate from $-\infty$ to ∞ , using $U(\xi, \eta) = P(\xi, \eta)U'(\xi, \eta)$ where $P(\xi, \eta)$ is the pupil function. The latter equals to one in the pupil and zero outside.

1.1.3 Fresnel approximation

To reduce eqt. (1.7), also known as the Huygens-Fresnel principle, one can approximate the distance r_{01} using the Taylor expansion of the square root :

$$r_{01} = z \sqrt{1 + \frac{x - \xi}{z} + \frac{y - \eta}{z}} \approx z \left[1 + \frac{1}{2} \left(\frac{x - \xi}{z} \right)^2 + \frac{1}{2} \left(\frac{y - \eta}{z} \right)^2 \right] \quad (1.8)$$

To obtain the Fresnel approximation, one has to replace r_{01} by eqt. (1.8) in eqt. (1.7). At the denominator, only the first term z is kept, since the introduced error is small, but in the exponential everything is kept. Then the final expression is given by,

$$U(x, y) = \frac{e^{jkz}}{j\lambda z} \iint_{-\infty}^{\infty} U(\xi, \eta) \exp \left\{ j \frac{k}{2z} [(x - \xi)^2 + (y - \eta)^2] \right\} d\xi d\eta. \quad (1.9)$$

In this form, the Fresnel approximation can be seen as a convolution between $U(\xi, \eta)$ and $h(x, y) = \frac{e^{jkz}}{j\lambda z} \exp \left[\frac{jk}{2z} (x^2 + y^2) \right]$.

Another form is found by developing $[(x - \xi)^2 + (y - \eta)^2]$,

$$U(x, y) = \frac{e^{jkz}}{j\lambda z} e^{j\frac{k}{2z}(x^2+y^2)} \iint_{-\infty}^{\infty} \left\{ U(\xi, \eta) e^{j\frac{k}{2z}(\xi^2+\eta^2)} \right\} e^{-j\frac{2\pi}{\lambda z}(x\xi+y\eta)} d\xi d\eta, \quad (1.10)$$

it is the Fourier transform of the complex field in the pupil multiplied by a quadratic phase exponential.

1.1.4 Fraunhofer approximation

In addition to the Fresnel approximation, we can introduce another approximation using the condition,

$$z \gg \frac{k(\xi^2 + \eta^2)_{max}}{2}. \quad (1.11)$$

If eqt. (1.11) is satisfied the Fresnel approximation simplifies, since the quadratic phase factor in (ξ, η) is approximately one on the entire pupil, as

$$U(x, y) = \frac{e^{jkz}}{j\lambda z} e^{j\frac{k}{2z}(x^2+y^2)} \iint_{-\infty}^{\infty} U(\xi, \eta) e^{-j\frac{2\pi}{\lambda z}(x\xi+y\eta)} d\xi d\eta. \quad (1.12)$$

For instance, at a wavelength of 637.5 nm and a pupil diameter of 3.6 mm the Fraunhofer approximation constrain z to be greater than 63 meters to be valid.

1.1.5 Converging lens introduction

The Fraunhofer conditions are severe as shown above, but one can reduce the distance z by observing at the focal plane of a converging lens. Indeed, using the paraxial approximation, i.e. small angles with respect with the optical axis, the lens transmission function is given by,

$$\begin{aligned} t_l(\xi, \eta) &= \exp[jkn\Delta_0] \exp\left[-jk(n-1)\frac{\xi^2 + \eta^2}{2}\left(\frac{1}{R_1} - \frac{1}{R_2}\right)\right] \\ &= \exp\left[-j\frac{k}{2f}(\xi^2 + \eta^2)\right], \end{aligned} \quad (1.13)$$

where n is the refractive index of the lens material, R_1 and R_2 are the radii of curvature of the front and back surface of the lens, respectively and f is the focal length of the lens defined as,

$$\frac{1}{f} \equiv (n-1)\left(\frac{1}{R_1} - \frac{1}{R_2}\right). \quad (1.14)$$

We can define $U_l(\xi, \eta) = U(\xi, \eta)t_l(\xi, \eta)$, which represents the complex amplitude passing through a lens. Finally, replacing $U(\xi, \eta)$ by $U_l(\xi, \eta)$ in Fresnel approximation and setting the observing distance to the focal length of the converging lens, we recover the Fraunhofer approximation,

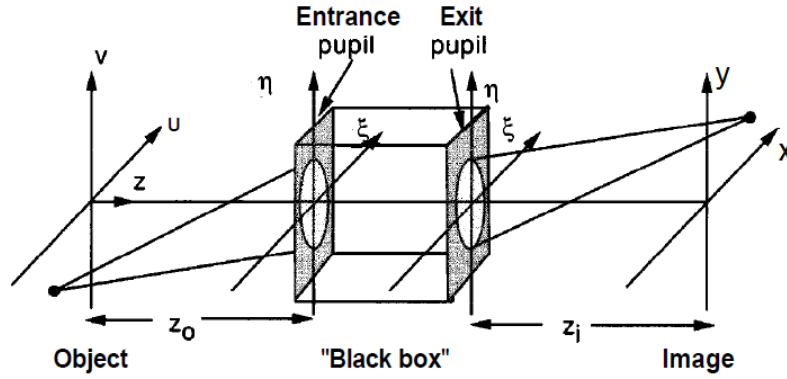


FIGURE 1.2: Schema of an imaging instrument, (Goodman, 1988, Chapter 6.1)

$$\begin{aligned}
 U(x, y) &= \frac{e^{jkz}}{j\lambda z} e^{j\frac{k}{2z}(x^2+y^2)} \mathcal{F} \left\{ U(\xi, \eta) \exp \left[j\frac{k}{2}(\xi^2 + \eta^2) \left(\frac{1}{z} - \frac{1}{f} \right) \right] \right\} \\
 &\stackrel{z=f}{=} \frac{e^{jkz}}{j\lambda z} e^{j\frac{k}{2z}(x^2+y^2)} \mathcal{F} \{ U(\xi, \eta) \}.
 \end{aligned} \tag{1.15}$$

1.2 Imaging system

An imaging system, such as a telescope, is used to acquire images of an object as perfectly as possible. An optical system, forming an instrument, is composed by lenses, mirrors, etc... and a detector (can be the human eye). A complex optical system can be reduced to a pupil, $P(\xi, \eta)$, and a focal length, f . The diffraction of the wave can be determined by the Fraunhofer approximation as long as the paraxial approximation is valid, see subsection 1.1.5. And the observed image of an incoherent object at the focal plane of the system is proportional to the square modulus of the complex amplitude $U(x, y)$,

1.2.1 Impulse Response (IR)

The impulse response or point spread function (PSF), $h(x, y; u, v)$, of an optical system is the field amplitude induced at coordinates (x, y) by a unit-amplitude point source at object coordinates (u, v) . Using the linearity of the wave propagation, we can write the imaged amplitude as the superposition integral,

$$U(x, y) = \iint_{-\infty}^{\infty} h(x, y; u, v) U(u, v) du dv \tag{1.16}$$

1.2.2 Optical Transfer Function (OTF)

The optical transfer function, OTF, is defined as the Fourier transform of the impulse response, see Figure 1.4. Using the Fourier transform properties it can also be given by the autocorrelation of the pupil function,

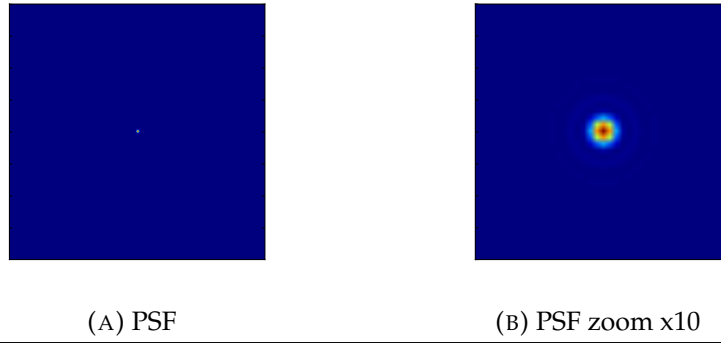


FIGURE 1.3: PSF of a perfect imaging system composed by a 3.6 mm pupil and a focal length of 80 mm at a wavelength of 637.5 nm. The size, N , of the PSF is 400 and the pixel size is $5.3 \mu\text{m}$.

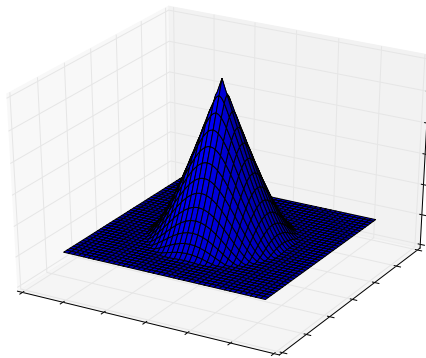


FIGURE 1.4: OTF of a perfect imaging system composed by a 3.6 mm pupil and a focal length of 80 mm at a wavelength of 637.5 nm.

$$\tilde{h}_{optical}(\xi, \eta) = \mathcal{F} \{h_{optical}(x, y)\} = (P \otimes P)(\xi, \eta), \quad (1.17)$$

where ξ and η are the conjugate variables of x and y with respect to the Fourier transform.

1.2.3 From Object to Image

A detector only senses the energy distribution produced by an electromagnetic wave. Therefore, an image is given by the square modulus of the complex amplitude at the focal plane,

$$i(x, y) = |U(x, y)|^2 = \left| \iint_{-\infty}^{\infty} h(x, y; u, v) U(u, v) du dv \right|^2. \quad (1.18)$$

This integral simplifies differently depending on the type of object we are observing. For a **coherent object**, Goodman (1988, Chapter 6.2) showed that the imaging is linear in complex amplitude, thus eqt. (1.18) becomes,

$$i(x, y) = \left| \iint_{-\infty}^{\infty} h(x - \tilde{u}, y - \tilde{v}) U(\tilde{u}, \tilde{v}) d\tilde{u} d\tilde{v} \right|^2, \quad (1.19)$$

where $(\tilde{u} = Mu, \tilde{v} = Mv)$ are the normalized object coordinates and M is the magnification of the imaging system. The image is given by the squared modulus of the convolution of the impulse response and the object complex amplitude.

For an **incoherent object**, Goodman (1988, Chapter 6.2) showed that the imaging is linear in intensity,

$$i(x, y) = \iint_{-\infty}^{\infty} |h(x - \tilde{u}, y - \tilde{v})|^2 o(\tilde{u}, \tilde{v}) d\tilde{u} d\tilde{v} = (h_{optical} \otimes o)(x, y), \quad (1.20)$$

where $o(x, y) = |U(x, y)|^2$. We can recognize the convolution of an object with an intensity impulse response, $h_{optical}(x, y) = |h(x, y)|^2$.

In this study, we will study stars radiation which are object with an incoherent emission. The stars are point source objects given there distances. The image that an optical system gives of a point source is called the point spread function, PSF, or impulse response, IR, of the system, see Figure 1.3. A point source is characterized by an infinite distance to the instrument and therefore the wave is planar, which means that the phasor is reduced to $U(\xi, \eta) = P(\xi, \eta)$. The PSF or IR is given by,

$$h_{optical}(x, y) = |\mathcal{F} \{P(\xi, \eta)\}|^2(x, y) \quad (1.21)$$

The domain where the impulse response is invariant under translation is called the **isoplanatic domain**.

In presence of aberrations, which will be discussed in the section 1.3, the wavefront is deformed with respect to the perfect planar or spheric form. The pupil function at the exit of the imaging system is modified as following,

$$\mathcal{P}(\xi, \eta) = P(\xi, \eta) e^{j\phi_{Ab}(\xi, \eta)}, \quad (1.22)$$

where $\phi_{Ab}(\xi, \eta)$ is the dephasing caused by the aberration present between the object and the image planes. Replacing the new pupil function in eqt. (1.21), we obtain the PSF of an imaging system having aberrations on the optical path.

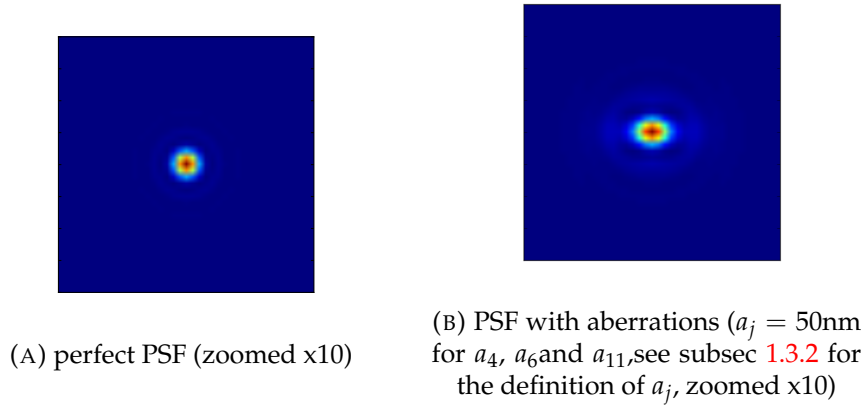


FIGURE 1.6: Comparison of perfect PSF and PSF with aberrations of an imaging system composed by a 3.6 mm pupil and a focal length of 80 mm at a wavelength of 637.5 nm. The size, N , of the PSF is 400 and the pixel size is $5.3 \mu\text{m}$.

1.3 Aberrations

The aberrations present on the optical path between the object and the image planes decrease the quality of the resulting image. Indeed, they induce fluctuations of the amplitude and phase of the wave in the pupil plane. Since the amplitude fluctuations are negligible with respect to the phase fluctuations, we focus only on the latter. In figure 1.5, one can see the effect of the aberrations on an hypothetical spherical wavefront at the exit pupil of an imaging system. And in Figure 1.6, one can see the effect of aberrations on the PSF of an imaging system. The PSF is blurred due to the defocus and we can well recognize the astigmatism introduced as the shape of the PSF tends towards the ellipse.

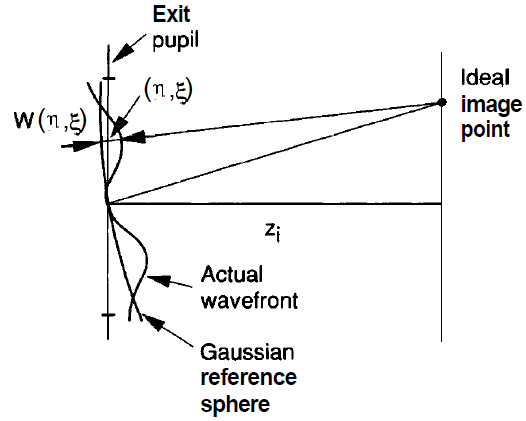


FIGURE 1.5: Gaussian reference sphere vs. aberrated Wavefront, (Goodman, 1988, Chapter 6.4). The dephasing is equal to the wave number multiplied by the aberration function,

$$\phi_{Ab}(\xi, \eta) = kW(\xi, \eta).$$

1.3.1 Sources of aberration

The phase fluctuations introduced by aberrations are due to different kind of perturbation on the optical path. In ground based astronomy, the main source of aberrations is the **atmosphere**. Indeed, the atmosphere's temperatures fluctuations coupled with turbulent airflows are at the origin of the variations of the refractive index. Those variations induce then different optical path, in other words they generate perturbations on an electromagnetic wave passing through the atmosphere. The theory of the Atmospheric Turbulence is beyond the scope of this work, but we redirect the interested reader to Obukhov (1949), V.I. Tatarski (1961), and Kolmogorov (1968).

The other source of aberrations are the **defects of the instrument** themselves. The defaults can limit the resolution and decrease the quality of an diffraction-limited imaging system. The defects can have multiple origins as described by Blanc (2002). The first one is a **default during the fabrication process**, such as mirror polishing. This kind of defect is fixed and of high spatial frequency. Perturbation of the electromagnetic wave can also be due to **misalignment of the optical components of the instrument**. For instance, during the first stages of operation stages of the Hubble telescope, the mirrors were not aligned and the resulting images were blurry. These defects are of low spatial frequency and vary slowly with time. Finally, there are defects due to **mechanical stresses of the instrument**. The mirrors have to be held in place by different mechanical components. And under the influence of the gravity, a mirror deformation can arise resulting in an aberration introduction. This kind of defect also evolves slowly through time but has a large spatial frequency domain.

In this work, we focus on a method to correct the effect of the instrument defects. This method is particularly suited since it does not require a different path than the path to the scientific instrument, which means that we can correct the aberrations on the entire optical path.

1.3.2 Zernike polynomials

In order to study the aberrations present in an imaging system, F. Zernike (1934) introduced an orthonormal basis on which we can decompose the phase on a circular pupil, such has a telescope pupil. Those polynomials are the factor of a trigonometric function and a polynomial function (Noll, 1976).

$$Z_j(\mathbf{r}) = R_n^m(r)\Theta_n^m(\theta), \quad (1.23)$$

where $r = \frac{r'}{R_{pup}}$ is the normalized radius on the unit circle and θ is the azimuthal angle on the unit circle. j correspond to the Zernike index in the Noll order, for a specific j correspond only one couple (n, m) . The values of n and m are always integral and satisfy $|m| \leq n, n - |m| = \text{even}$.

The trigonometric function is given by,

$$\Theta_n^m(\theta) = \begin{cases} \sqrt{n+1} & \text{if } m = 0, \\ \sqrt{2(n+1)}\cos(m\theta) & \text{if } m \neq 0 \text{ and } i \text{ even}, \\ \sqrt{2(n+1)}\sin(m\theta) & \text{if } m \neq 0 \text{ and } i \text{ odd}, \end{cases} \quad (1.24)$$

and the radial function is given by,

$$R_n^m(r) = \sum_{s=0}^{(n-m)/2} \frac{(-1)^s (n-s)!}{s![(n+m)/2-s]![(n-m)/2-s]!} r^{n-2s}. \quad (1.25)$$

The decomposition of the phase onto the Zernike polynomials is given by,

$$\phi(\mathbf{r}) = \sum_{i=1}^{+\infty} a_i Z_i(\mathbf{r}), \quad (1.26)$$

where the a_i 's are the Zernike coefficient. And we characterize a wavefront by its root mean squared error without the piston component (a_1), which is given by,

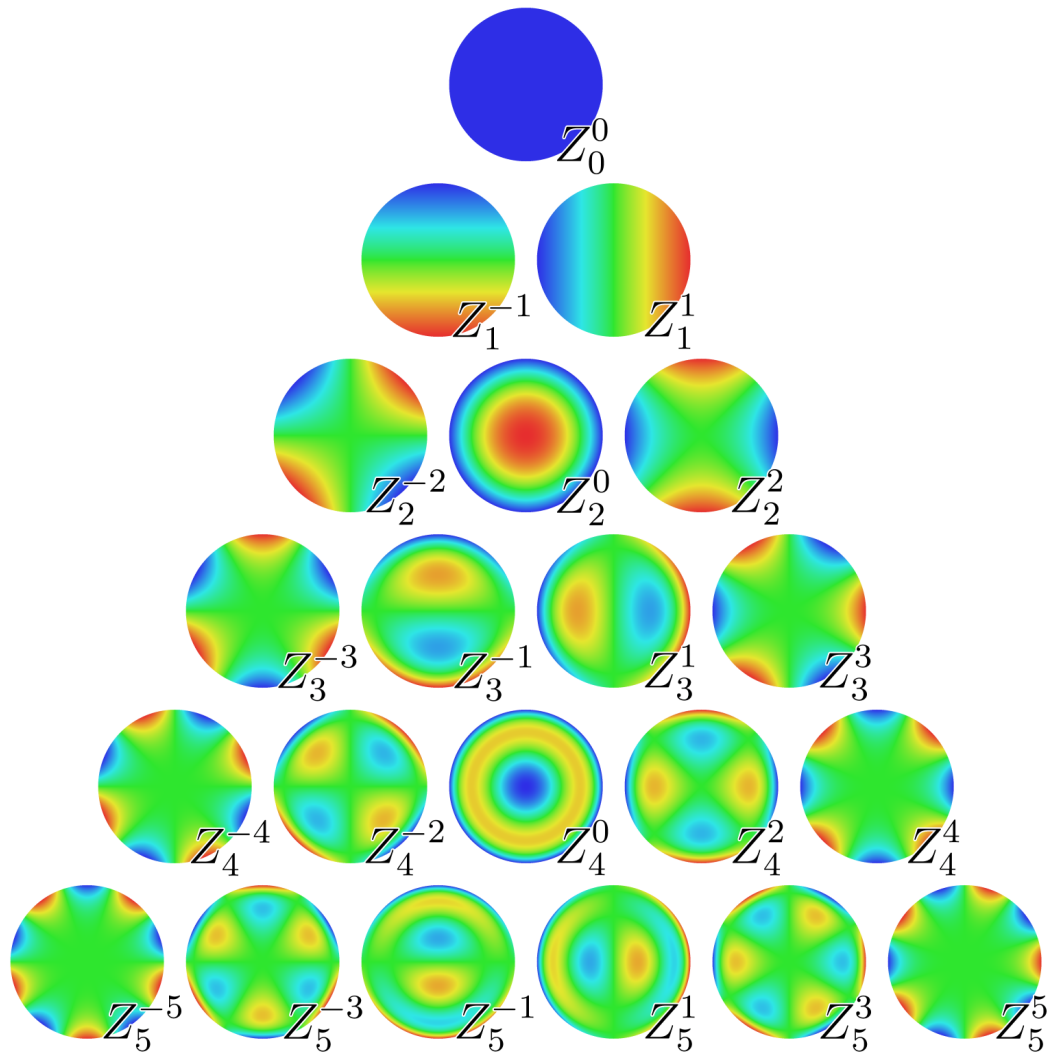


FIGURE 1.7: Representation of the 21st Zernike polynomials (Wikipedia, 2018)

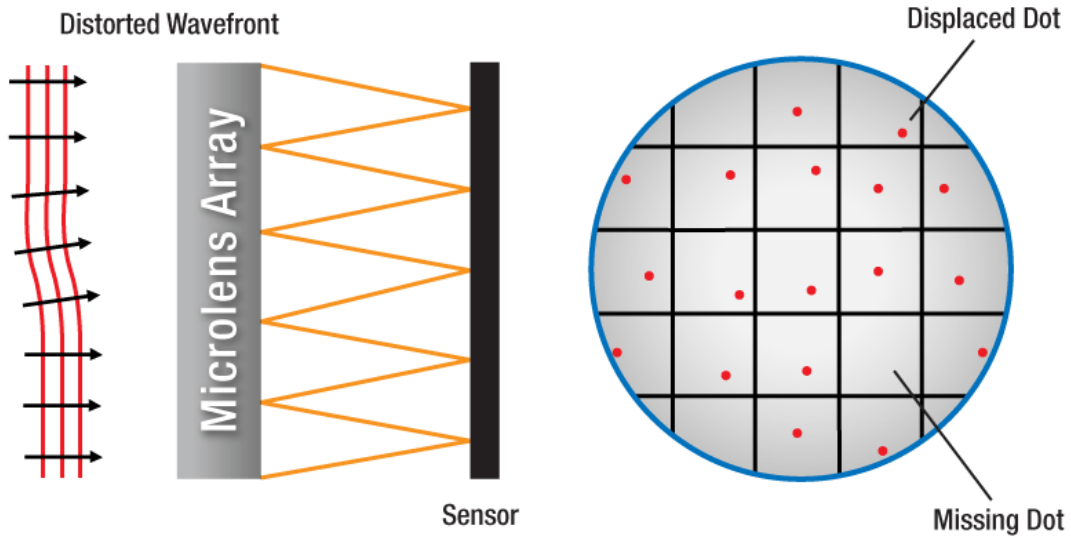


FIGURE 1.8: Shack-Hartmann principle (Thorlabs, 2018)

$$\sigma_\phi = \sqrt{\frac{1}{S} \int_S \phi^2(\mathbf{r}) d\mathbf{r}} = \sqrt{\sum_{i=2}^{+\infty} a_i^2}, \quad (1.27)$$

where S is the surface of the pupil.

1.4 Phase retrieval

Phase retrieval is a complicated process since the detectors are only sensitive to the intensity of a wave and not the wave itself, which renders impossible to have a direct relation between phase and image, thus we only have an indirect measurement of the wavefront. It is important to be able to estimate the aberrations of a system in order to correct them in real-time (Adaptive Optics systems) or in post-processing. There are a couple of ways to characterize the form of a wavefront in order to determine the amplitude of the aberrations present in an imaging system. Some use the optical geometric approximation, which says that locally the light rays are perpendicular to the wavefront. These achromatic methods measure the gradient of the wave surface, such as Shack-Hartmann (Shack and Platt, 1971), Curvature sensing (Roddier, 1988) and Pyramid sensing (Ragazzoni, 1996). Another kind of methods are called interferometric or focal plane methods, they use the interference patterns of the pupil to determine the form of the wavefront. The Phase Diversity is part of those kind of methods, it was discovered by Gerchberg and Saxton (1972) and Gonçalves (1982). The most popular method nowadays are the Shack-Hartmann, the curvature and the phase diversity. The latter will be explained in chapter 2 and we will explain the principle of a Shack-Hartmann sensor since we will use it later on.

1.4.1 Shack-Hartmann (Shack and Platt, 1971)

The Shack-Hartmann wavefront sensor measures the gradient of an aberrant phase. Figure 1.8 shows the principle. A micro-lenses array samples the wavefront passing through the pupil in a conjugated plane of the entrance pupil. Each micro-lens produces a dot on a CCD placed at the foci of the micro-lenses. The deformations of the wavefront induce a displacement of the dot with respect to a reference position obtained with a perfectly planar or spherical wavefront. By measuring this displacement Δx , it gives directly the local slope of the wavefront,

$$\frac{2\pi}{\lambda} \frac{\Delta x}{f} = \frac{1}{S} \int_S \frac{\delta\phi(x, y)}{\delta x} dx dy, \quad (1.28)$$

where S is the surface of a micro-lens and f is the focal length of the micro-lenses.

Chapter 2

Phase Diversity

Chapter 3

Phase Diversity Experiment

Here, we will describe the experiment put in place in the optical laboratory at the HEIG-VD to reconstruct wavefronts with unknown static aberrations introduced using phase screens. At first, we study the behavior of the phase diversity algorithm put in place by Mugnier, Blanc, and Idier (2006) at ONERA with respect to number of averaging images, in other words noise level, and number of Zernike coefficients retrieved. Then we test the algorithm using a known aberration introduced by a parallel plane plate in the beam comparing the result to Zemax simulation. And finally, we introduce the phase screen to have random aberrations in the pupil and try to compare the phase diversity results with the Shack Hartman wavefront sensor results.

3.1 Experimental Setup

The design of the experiment was already done by Bouxin (2017). The system is built according to her plans and specifications.

The experiment is mounted on a pressurized legs optical table. The assembly contains six main components : a light source, an entrance pupil, an imaging system, a converging lens to focus the beam on the camera, a camera and a wavefront sensor.

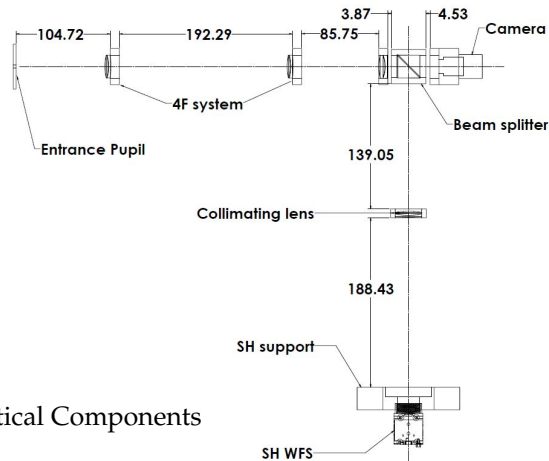


TABLE 3.1: Optical Components

#	Components	Model	Reference
1	Pigtailed laser diode	Thorlabs, LPS-635-FC	A.1
2	Converging lens, $f = 11$ mm	Thorlabs, AP20TMA	A.2
3	Pinhole, $10\ \mu\text{m}$	Thorlabs, P10S	A.3
4	Converging lens, $f = 200$ mm	Thorlabs, AL100200	A.4
5	3.2 mm Hole milled in metal sheet
6	Converging lens, $f = 100$ mm	Thorlabs, AC254-100-A	A.5
7	Converging lens, $f = 80$ mm		
8	Camera CMOS	Ximea, MQ013MG-E2	A.6
9	Converging lens, $f = 100$ mm		
10	Shack-Hartman WFS	Thorlabs, WFS150-5C	A.7

FIGURE 3.1: Experimental setup schema with the relevant distances, (Bouxin, 2017).

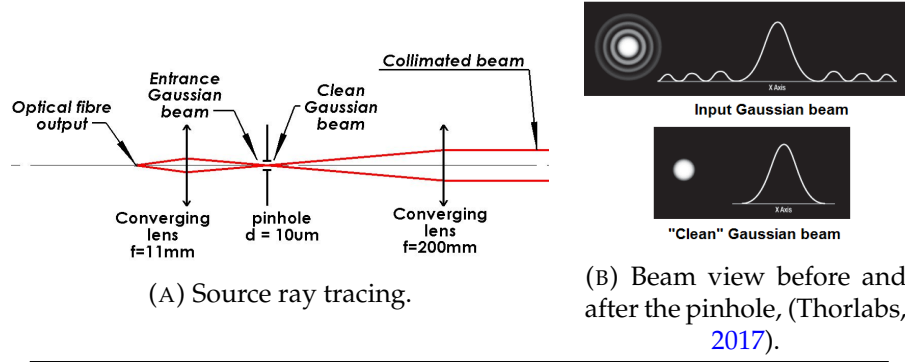
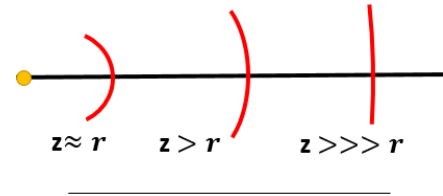


FIGURE 3.3: Source schema and pinhole effect on the beam.

3.1.1 Light source

The final application of the phase diversity will be to characterize the optical aberrations induced by the imperfect optical path to a scientific detector of a telescope. For this reason, the light source has to simulate a distant star aberration-free wavefront. A distant star wavefront is considered planar since the object distance, z , is far greater than the telescope size, r , see Fig. 3.2. The source of our experiment must then be characterized by a planar wavefront.

In order to obtain such a planar wavefront at the entrance pupil, the light source consist of a "pigtailed laser diode", a $f=11\text{mm}$ converging lens, a pinhole and a $f=200\text{ mm}$ converging lens, see Table 3.1. The pigtailed laser diode emits a Gaussian beam centred at 637.5 nm slightly diverging. The converging lens concentrates the beam at the center of the $10\mu\text{m}$ pinhole to filter the noise. The second converging lens collimates the beam, obtaining a collimated beam with a planar wavefront, see Fig. 3.3a and 3.3b.

FIGURE 3.2: Wavefront curvature for different point source's distances, z . r represents the characteristic size of the arc of interest.

3.1.2 Entrance pupil

The entrance pupil of our optical system is a circular aperture of 3.2 mm diameter placed after the collimating lens of the light source. It is milled in a metal plate and centred in his support, to avoid positioning with a XY table. The diameter is chosen in available material to fit in the different detector's surfaces.

3.1.3 Pupil imaging system

The phase diversity technique requires PSFs images as input, which means that the beam as to be focused onto the detector surface. To analyse the aberration in the pupil plane, one needs to focus an image of the beam passing through the entrance pupil. The simplest assembly to achieve this goal is the 4F system, which consist of two converging lenses of focal 100 mm . The two lenses are separated by 200 mm ,

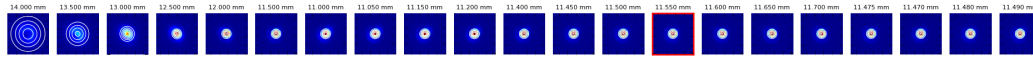


FIGURE 3.4: PSFs example of an alignment procedure

see Fig. 3.1. This places the image of the entrance pupil 100 mm after the second converging lens.

3.1.4 Detectors

The image of the entrance pupil, obtained with the 4F system, is focused onto a CMOS Ximea camera by a $f = 80$ mm converging lens to acquire the PSFs for the phase diversity wavefront retrieval. The camera has a surface composed by 1280×1024 pixels of $5.3 \mu\text{m}$, see Appendix A.6. It is mounted on sliding support in order to be able to acquire in/out-of-focus images. A beam splitter is placed in the converging beam to separate it in two. The second beam is collimated and a Shack-Hartman WFS is placed on the entrance pupil image plane, to check the results of the phase diversity wavefront retrieval. The Shack-Hartman WFS has a 39×31 lenslets grid and a CCD with a resolution of 1280×1024 pixels of $4.65 \mu\text{m}$, see Appendix A.7.

3.2 Data Acquisition

3.2.1 Ximea Camera

The ONERA algorithm takes at least one focused and one defocused PSFs, as described in section ???. The PSFs are acquired using a python script which uses an open-source library to control the ximea camera, `pyXimea`¹, available on GitHub. The acquisition is done following these steps :

1. The first step in order to acquire PSFs is to determine the position of the camera's focus point using the python script `AlignementScriptXimeaCamera.py`, see Appendix B.1. This script let's you acquire consecutively PSFs at different camera's positions and computes their FWHM. It finally returns the minimum FWHM and the camera's position, see Figure 3.4.
2. Once you have the focus point position

3.2.2 Shack-Hartman WFS

3.3 Results

This section presents the results of the phase diversity experiment, with the introduction of different sources of aberration.

¹<https://github.com/pupil-labs/pyximea>

3.3.1 Parallel plane plate

The first source of aberration studied in this work is a tilted parallel plane plate which is used as a calibrated source of astigmatism.

Appendix A

Optical Component Datasheets


A.1 Pigtailed laser diode

THORLABS

Pigtailed Laser Diode, SMF

Description

Thorlabs' Single Mode Pigtailed Laser Diodes are standard TO-packaged diodes that have been pigtailed to a 1 m long single mode fiber with an FC/PC connector. Each unit is tested before shipment. Please refer to the unit-specific test datasheet for optimal operating parameters.



Specifications

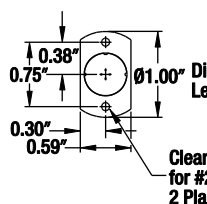
LPS-635-FC Specifications	
LD Reverse Voltage (Max)	2 V
PD Reverse Voltage (Max)	30 V
Optical Output Power	2.5 mW (Typ.) 3.5 mW (Max)
Operating Temperature	0 to 50 °C
Storage Temperature	-10 to 65 °C
Pin Code	9A
Laser Diode	HL6320G
Fiber	SM600
Connector	FC/PC

LPS-635-FC Specifications			
	Min	Typ.	Max
Wavelength	625 nm	635 nm	640 nm
Threshold Current*	20 mA	50 mA	75 mA
Slope Efficiency*	0.13 mW/mA	0.18 mW/mA	-
Operating Current @ P ₀ = 2.5 mW*	-	70 mA	95 mA
Operating Voltage @ P ₀ = 2.5 mW*	-	2.2 V	2.7 V
Monitor Current @ P ₀ = 2.5 mW*	0.05 mA	0.17 mA	0.3 mA

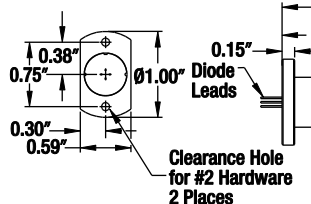
*Temperature = 25 °C

Drawing

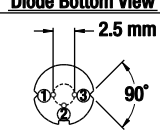
Pigtail Bottom View



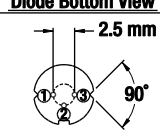
Side View




Diode Bottom View



Pin Diagram





NOTICE

To avoid equipment damage from electrostatic discharge: Wear ESD wriststrap when handling this device.

US, Canada, & South America: +1-973-300-3000 | France: +33 (0) 970 444 844 | Europe: +49 (0) 8131-5956-0 | UK & Ireland: +44 (0)1353-654440
 Brazil: +55-16-3413 7062 | Scandinavia: +46-31-733-30-00 | Japan & Asia: +81-3-5979-8889 | China: +86 (0)21-60561122

June 6, 2013
5175-S01, Rev B

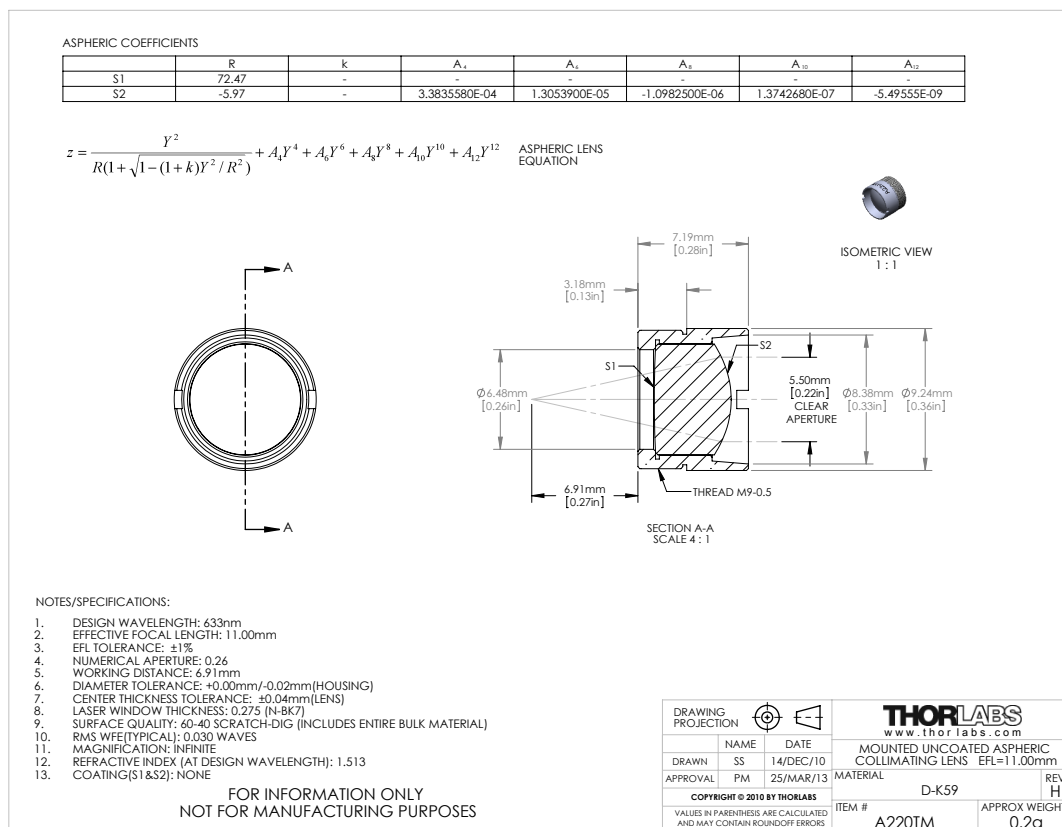
www.thorlabs.com

A.1.1 Power supply modification

The former student, ??? its name ???, mounted a diode driver card to power it. Unfortunately, for the phase diversity experiment, chapter , the power was too high and the Ximea camera was always saturated. So I modified the driver circuit and added two resistances to lower the current so that the detector do not reach the saturation.

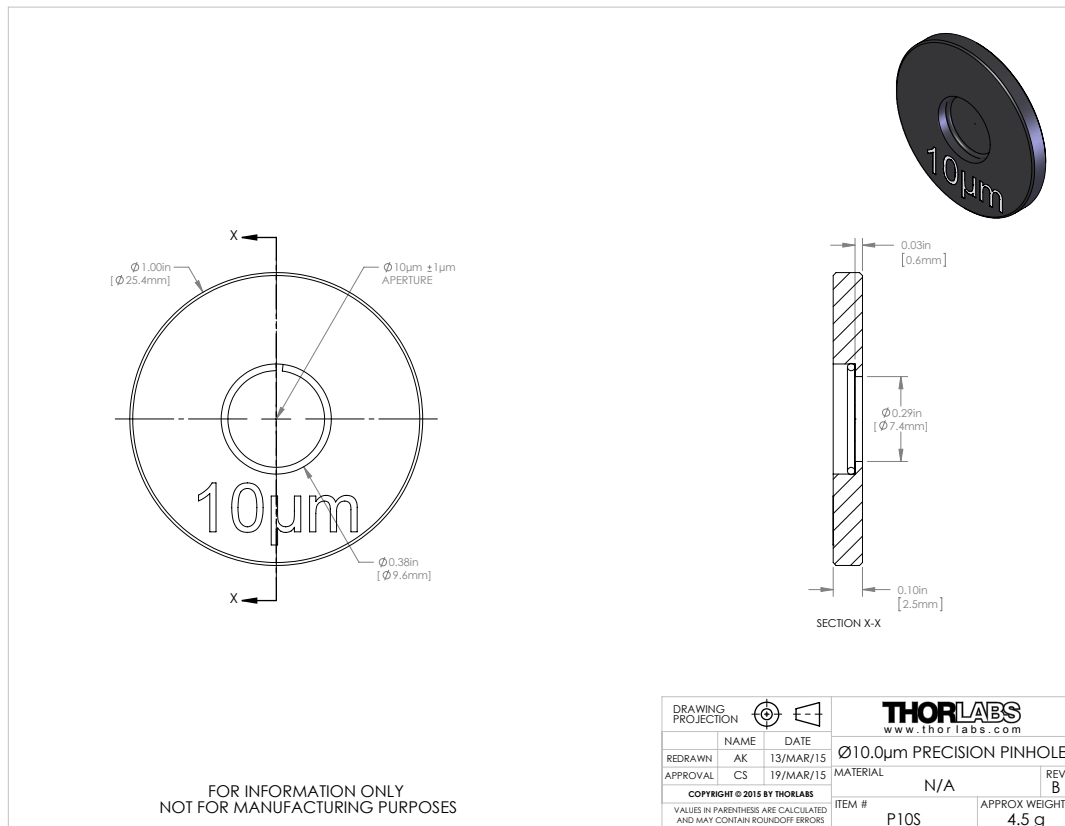
!!! mettre la photo du driver et de la modif !!!

A.2 Converging lens A220TM-A, $f = 11 \text{ mm}$



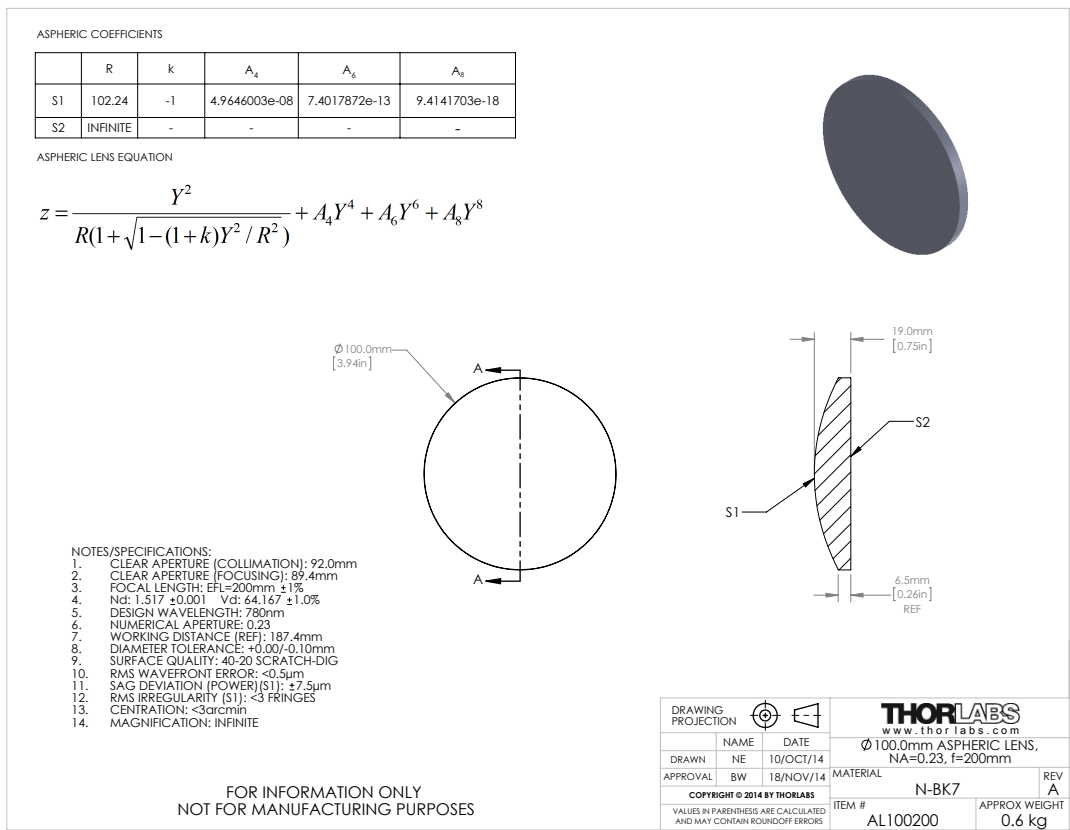
Source : www.thorlabs.com

A.3 Pinhole 10 μm



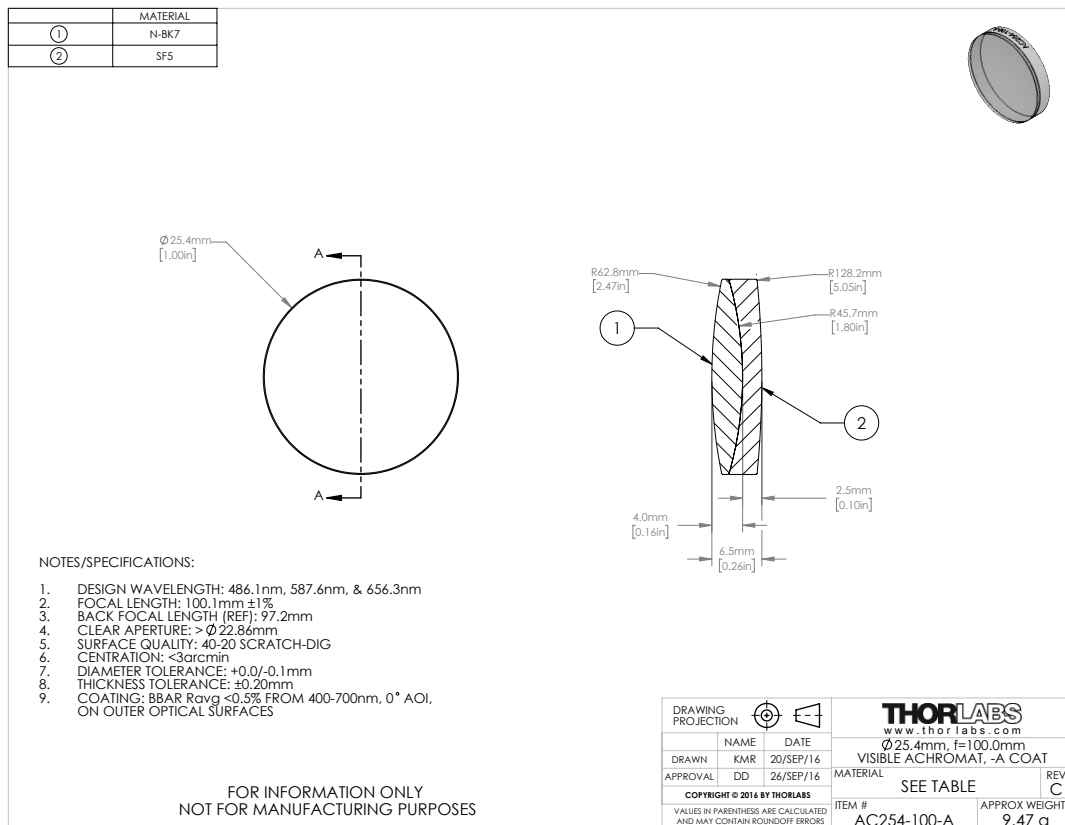
Source : www.thorlabs.com

A.4 Converging lens AL100200, f = 200 mm



Source : www.thorlabs.com

A.5 Converging lens AC254-100-A, $f = 100$ mm



Source : www.thorlabs.com

A.6 Ximea Camera, MQ013MG-E2



Specifications:

Resolution:	1.3 MP 1280 × 1024 pixels
Sensor type:	CMOS Matrix B/W
Sensor model:	e2V EV76C560 ABT-EQV
Sensor size:	1/1.8"
Sensor active area:	6.9 × 5.5 mm
Pixel size:	5.3 μm
Bits per pixel:	8, 10
Dynamic range:	60 dB
Frame rates:	60 fps
On-chip binning:	1x1, 2x2
Image data interface:	USB 3.0
Data I/O:	GPIO IN, OUT
Power requirements:	0.9 Watt
Lens mount:	C or CS Mount
Weight:	26 grams
Dimensions WxHxD:	26 x 26 x 26 mm
Operating environment:	50 °C
Customs tariff code:	8525.80 30 (EU) / 8525.80 40 (USA)
ECCN:	EAR99

Source : www.ximea.com/en/products/usb3-vision-cameras-xiq-line/mq013mg-e2

A.7 Shack-Hartmann wavefront sensor, WFS150-5C

8 Appendix

8.1 Technical Data

8.1.1 WFS150/300

Item #	WFS150-5C	WFS150-7AR	WFS300-14AR
Microlenses			
Microlens Array	MLA150M-5C	MLA150M-7AR	MLA300M-14AR
Substrate Material	Fused Silica (Quartz)		
Number of Active Lenslets	Software Selectable		
Max. Number of Lenslets	39 x 31		19 x 15
Camera			
Sensor Type	CCD		
Resolution	max. 1280 x 1024 pixels, Software Selectable		
Aperture Size	5.95 mm x 4.76 mm		
Pixel Size	4.65 μm x 4.65 μm		
Shutter	Global		
Exposure Range	79 μs - 65 ms		
Frame Rate	max. 15 Hz		
Image Digitization	8 bit		
Wavefront Measurement			
Wavefront Accuracy ¹⁾	$\lambda/15$ rms @ 633 nm		$\lambda/50$ rms @ 633 nm
Wavefront Sensitivity ²⁾	$\lambda/50$ rms @ 633 nm		$\lambda/150$ rms @ 633 nm
Wavefront Dynamic Range ³⁾	> 100 λ @ 633 nm		> 50 λ @ 633 nm
Local Wavefront Curvature ⁴⁾	> 7.4 mm	> 10.0 mm	> 40.0 mm
External Trigger Input			
Save Static Voltage level	0 to 30 V DC		
LOW Level	0.0 V to 2.0 V		
HIGH Level	5.0 V to 24 V		
Input current	> 10 mA		
Min Pulse Width	100 μs		
Min. Slew Rate	35 V / msec		
Common Specifications			
Optical Input	C-Mount		
Power Supply	<1.5 W, via USB		
Operating Temperature Range ⁵⁾	+5 to +35 °C		
Storage Temperature Range	-40 to 70 °C		
Warm-Up Time for Rated Accuracy	15 min		
Dimensions (W x H x D)	32.0 mm x 40.4 mm x 45.5 mm		
Weight	0.1 kg		

¹⁾ Absolute accuracy using internal reference. Measured for spherical wavefronts of known RoC.

²⁾ Typical relative accuracy. Achievable after, and with respect to a user calibration, 10 image averages

³⁾ Over entire aperture of wavefront sensor

⁴⁾ Radius of wavefront curvature over single lenslet aperture

⁵⁾ non-condensing

All technical data are valid at 23 \pm 5°C and 45 \pm 15% rel. humidity (non condensing)

Appendix B

Python Code

B.1 AlignementScriptXimeaCamera.py

```

1  ##Script to compute the FWHM of the beam on the camera averaging
2  #over "nbrImgAveraging" images and see which position minimizes it.
3
4  from ximea import xiapi
5  import numpy as np
6  from matplotlib import pyplot as plt
7  import scipy.optimize as opt
8  import datetime
9  import functionsXimea as fX
10 import seaborn as sns
11 import os
12 sns.set()
13 ### instantiation
14
15 dataFolderPath =
16     'C:/Users/Jojo/Desktop/PdM-HEIG/Science/data/PD/phaseScreen/alignement/'
17 plotFolderPath =
18     'C:/Users/Jojo/Desktop/PdM-HEIG/Science/fig/PD/phaseScreen/alignement/'
19 #create the matrix grid of the detector CCD
20 x = np.linspace(0,1280,1280)
21 y = np.linspace(0,1024,1024)
22 x, y = np.meshgrid(x, y)
23
24 #initial guess for the fit depending on the position of the beam in the CCD
25 initial_guess = [250, 481, 706, 3, 3]
26
27 #number of image to average
28 nbrImgAveraging = 10
29
30 ###data acquisition and treatment
31
32 #create instance for first connected camera
33 cam = xiapi.Camera()
34 #start communication
35 print('Opening camera...')
36 cam.open_device()
37 #settings
38 cam.set_imgdataformat('XI_MONO8') #XIMEA format 8 bits per pixel
39 cam.set_gain(0)
40 #create instance of Image to store image data and metadata
41 img = xiapi.Image()
42 #start data acquisition

```

```

41 print('Starting data acquisition...')
42 if cam.get_acquisition_status() == 'XI_OFF':
43     cam.start_acquisition()
44
45 cam.set_exposure(fX.determineUnsaturatedExposureTime(cam,img,[60,10000],1))
46
47 #instanciation for the while loop
48 answer = 'y'
49 i=0
50 relativePos = []
51 data = []
52 data_fitted = []
53 FWHMx = []
54 FWHMy = []
55 x0 = []
56 y0 = []
57 sigmaX0 = []
58 sigmaY0 = []
59
60 while answer == 'y':
61
62     try:
63         relativePos.append(float(raw_input('What is the position on the
64             screw [mm] ? ')))
65     except ValueError:
66         print('Not a float number')
67
68     [tmpdata,stdData] = fX.acquireImg(cam,img,nbrImgAveraging)
69     data.append(tmpdata)
70     #Fit the img data on the 2D Gaussian to compute the FWHM
71     print('Fitting 2D Gaussian...')
72     popt, pcov = opt.curve_fit(fX.TwoDGaussian, (x,y), data[i].ravel(), p0 =
73         initial_guess)
74     print('Fitting done')
75
76     FWHMx.append(2*np.sqrt(2*np.log(2))*popt[3])
77     FWHMy.append(2*np.sqrt(2*np.log(2))*popt[4])
78     x0.append(popt[2])
79     sigmaX0.append(popt[4])
80     y0.append(popt[1])
81     sigmaY0.append(popt[3])
82
83     print 'Fig %d : (x,y) = (%3.2f,%3.2f), FWHM x = %3.2f, FWHM y = %3.2f'
84         %(i,x0[i],y0[i],FWHMx[i],FWHMy[i])
85
86     data_fitted.append(fX.TwoDGaussian((x, y),
87         popt[0],popt[1],popt[2],popt[3],popt[4]).reshape(1024, 1280))
88
89     #plot the beamspot
90     fig, ax = plt.subplots(1, 1)
91     ax.imshow(data[i], cmap=plt.cm.jet,origin='bottom',
92         extent=(x.min(), x.max(), y.min(), y.max()))
93     ax.contour(x, y, data_fitted[i], 5, colors='w',linewidths=0.8)
94     plt.xlim( (popt[2]-4*popt[4], popt[2]+4*popt[4]) )
95     plt.ylim( (popt[1]-4*popt[3], popt[1]+4*popt[3]) )
96     plt.show()
97

```

```

94     #ask if the person wants to acquire a new image to improve the alignment
95     pressedkey = raw_input('Do you want to acquire an other image [y (yes)
        or n (no)]: ')
96     if (pressedkey == 'n'):
97         answer = pressedkey
98     #increase i
99     i+=1
100
101 #stop data acquisition
102 print('Stopping acquisition...')
103 cam.stop_acquisition()
104
105 #stop communication
106 cam.close_device()
107
108 #convert list to np.array
109 relativePos = np.array(relativePos)
110 data = np.array(data)
111 FWHMx = np.array(FWHMx)
112 FWHMy = np.array(FWHMy)
113 x0 = np.array(x0)
114 y0 = np.array(y0)
115 sigmaX0 = np.array(sigmaX0)
116 sigmaY0 = np.array(sigmaY0)
117
118 #plot the FWHM vs. relPos
119 fig, ax = plt.subplots(1,1)
120 ind = np.argsort(relativePos)
121 ax.plot(relativePos[ind], (np.sqrt(FWHMx**2+FWHMy**2))[ind])
122 ax.set_xlabel('Position [mm]')
123 ax.set_ylabel('FWHM [px]')
124 ax.grid()
125 date = datetime.datetime.today()
126 if not os.path.isdir(plotFolderPath):
127     os.makedirs(plotFolderPath)
128 plt.savefig(plotFolderPath+date.strftime('%Y%m%d%H%M%S')+ 'FWHM_pos.pdf')
129 plt.savefig(plotFolderPath+date.strftime('%Y%m%d%H%M%S')+ 'FWHM_pos.png')
130
131
132 indOfMinFWHM = np.argmin(np.sqrt(FWHMx**2+FWHMy**2))
133
134 fig, axarr = plt.subplots(1,np.size(data,0))
135 #plot all the images besides each other
136 for iImg in ind:
137     axarr[iImg].imshow(data[iImg], cmap=plt.cm.jet,origin='bottom',
138         extent=(x.min(), x.max(), y.min(), y.max()))
139     # axarr[iImg].contour(x, y, data_fitted[iImg], 5,
140         colors='w',linewidths=0.8)
141     axarr[iImg].set_xlim( (x0[iImg]-12, x0[iImg]+12) )
142     axarr[iImg].set_ylim( (y0[iImg]-12, y0[iImg]+12) )
143     axarr[iImg].set_yticklabels(' ',visible=False)
144     axarr[iImg].set_xticklabels(' ',visible=False)
145     axarr[iImg].set_title('%5.3f mm'%relativePos[iImg],fontsize=8)
146     if iImg == indOfMinFWHM:
147         axarr[iImg].set_frame_on(True)
148         for pos in ['top', 'bottom', 'right', 'left']:
149             axarr[iImg].spines[pos].set_edgecolor('r')

```

```

149         axarr[iImg].spines[pos].set_linewidth(2)
150     else:
151         axarr[iImg].set_frame_on(False)
152 plt.show()
153 date = datetime.datetime.today()
154
155 plt.savefig(plotFolderPath+date.strftime('%Y%m%d%H%M%S')+'ImgPSF.pdf')
156 plt.savefig(plotFolderPath+date.strftime('%Y%m%d%H%M%S')+'ImgPSF.png')
157
158
159 #save data
160 if not os.path.isdir(dataFolderPath):
161     os.makedirs(dataFolderPath)
162 date = datetime.datetime.today()
163 np.save(dataFolderPath+date.strftime('%Y%m%d%H%M%S')+'data.npy',data)
164 np.save(dataFolderPath+date.strftime('%Y%m%d%H%M%S')+'relativePos.npy',relativePos)

```

B.2 AcquisAndSaveXimea.py

```

1  %%% Script to acquire images average over nbrImgAveraging images and save
   them into fits file
2
3  from ximea import xiapi
4  import datetime
5  import functionsXimea as fX
6  import winsound
7  import numpy as np
8
9  %%%instanciation
   -----
10 #number of image to average
11 nbrImgAveraging = 5000
12 numberOfFinalImages = 1
13
14 #Cropping information
15 sizeImg = 256
16
17 #Parameter of camera and saving
18 folderPathCropped =
   '...../data/PD/astigmatism/angle_study_3/wth/cropped/20/'
19 darkFolderPathCropped =
   '...../data/dark/astigmatism/angle_study_3/wth/cropped/20/'
20 folderPathFull = '...../data/PD/astigmatism/angle_study_3/wth/full/20/'
21 darkFolderPathFull =
   '...../data/dark/astigmatism/angle_study_3/wth/full/20/'
22 nameCamera = 'Ximea'
23 focusPos = 11.63
24
25 #Sound
26 duration = 1000 # millisecond
27 freq = 2000 # Hz
28
29 #initial guess for the fit depending on the position of the beam in the CCD
30 initial_guess = [250, 468, 954, 3, 3]
31

```

```

32 #-----
33 ### data acquisition
34 -----
35
36 #Opening the connection to the camera
37 cam = xiapi.Camera()
38 cam.open_device()
39 cam.set_imgdataformat('XI_MONO8') #XIMEA format 8 bits per pixel
40 cam.set_gain(0)
41
42 img = xiapi.Image()
43 if cam.get_acquisition_status() == 'XI_OFF':
44     cam.start_acquisition()
45 ### exposition
46 cond = 1
47 while bool(cond):
48     source = ''
49     winsound.Beep(freq, duration)
50     source = int(raw_input('Is the source turned on and at focus point
51                           (usually %5.3f mm) (yes = 1) ? '%focusPos))
52     if source == 1:
53         cond = 0
54     else:
55         print 'Please turn on the source and place the camera on the focus
56               point (%5.3f mm) '%focusPos
57
58 if bool(source):
59     #Set exposure time
60     cam.set_exposure(fX.determineUnsaturatedExposureTime(cam,img,[1,10000],1))
61     #get centroid
62     centroid = fX.acquirePSFCentroid(cam,img,initial_guess)
63     print 'centroid at (%d, %d)' %(centroid[0],centroid[1])
64
65 ###Acquire images at different camera position
66
67 acquire = 1
68 while bool(acquire):
69     cond = 1
70     while bool(cond):
71         dark = ''
72         winsound.Beep(freq, duration)
73         dark = int(raw_input('Is the source turned off (yes = 1) ? '))
74         if dark == 1:
75             cond = 0
76         else:
77             print 'Please shut down the source.'
78
79     winsound.Beep(freq, duration)
80     pos = float(raw_input('What is the position of the camera in mm focused
81                           (%5.3f mm) dephase 2Pi (pos+ = %5.3f mm, pos- = %5.3f) ?
82                          '%(focusPos,focusPos+3.19,focusPos-3.19)))
83
84 if bool(dark):
85     print 'Acquiring dark image...'
86     # Acquire dark images
87     [darkData,stdDarkData] = fX.acquireImg(cam,img,nbrImgAveraging)
88     print 'Cropping'

```

```

84     [darkdataCropped, stddarkDataCropped] =
        fX.cropAroundPSF(darkData, stdDarkData, centroid, sizeImg, sizeImg)
85     print 'saving'
86     fX.saveImg2Fits(datetime.datetime.today(), darkFolderPathCropped, nameCamera, darkdataCropped, stddarkDataCropped)
87     fX.saveImg2Fits(datetime.datetime.today(), darkFolderPathFull, nameCamera, darkData, stdDarkData)
88
89     #Acquire images -----
90     cond = 1
91     while bool(cond):
92         source = ''
93         winsound.Beep(freq, duration)
94         source = int(raw_input('Is the source turned on (yes = 1) ? '))
95         if source == 1:
96             cond = 0
97         else:
98             print 'Please place turn on the camera'
99
100    if bool(source):
101        print 'Acquiring images...'
102        # Acquire focused images
103        for iImg in range(numberOfFinalImages):
104            imgNumber = iImg+1
105            print 'Acquiring Image %d'%imgNumber
106            [data, stdData] = fX.acquireImg(cam, img, nbrImgAveraging)
107            print 'Cropping'
108            [dataCropped, stdDataCropped] =
                fX.cropAndCenterPSF(data-darkData, stdData+stdDarkData, sizeImg, initial_guess)
109            print 'Saving'
110            fX.saveImg2Fits(datetime.datetime.today(), folderPathCropped, nameCamera, dataCropped, stdDataCropped)
111            fX.saveImg2Fits(datetime.datetime.today(), folderPathFull, nameCamera, data-darkData, stdDarkData+stdDataCropped)
112
113    cond = 1
114    while bool(cond):
115        acquire = ''
116        winsound.Beep(freq, duration)
117        acquire = int(raw_input('Do you want to acquire at an other camera
            position (yes = 1, no = 0) ? '))
118        if acquire == 1:
119            cond = 0
120        elif acquire == 0:
121            cond = 0
122        else:
123            print 'please answer with 0 or 1 for no or yes, respectively'
124
125    ##Stop the acquisition
126    cam.stop_acquisition()
127    cam.close_device()
128
129    print 'Acquisition finished'
130

```

Bibliography

- Blanc, Amandine (2002). "Identification de réponse impulsionnelle et restauration d'images: apports de la diversité de phase". PhD thesis. Université Paris XI UFR Scientifique d'Orsay.
- Bouxin, A. (2017). "Phasor diversity to measure the static aberrations of an optical system". MA thesis. HEIG-VD.
- F. Zernike, von (1934). "Beugungstheorie des schneidenverfahrens und seiner verbesserten form, der phasenkontrastmethode". In: *Physica* 1.7, pp. 689–704. ISSN: 0031-8914. DOI: [https://doi.org/10.1016/S0031-8914\(34\)80259-5](https://doi.org/10.1016/S0031-8914(34)80259-5). URL: <http://www.sciencedirect.com/science/article/pii/S0031891434802595>.
- Gerchberg, R. W. and W. O. Saxton (1972). "A practical algorithm for the determination of phase from image and diffraction plane pictures". In: *Optik* 35.2, 237–250.
- Gonsalves, Robert A. (1982). "Phase Retrieval And Diversity In Adaptive Optics". In: *Optical Engineering* 21, pp. 21–21–4. DOI: [10.1117/12.7972989](https://doi.org/10.1117/12.7972989). URL: <http://dx.doi.org/10.1117/12.7972989>.
- Goodman, Joseph W. (1988). *Introduction to Fourier Optics*. Ed. by L. Cox and John M. Morris. 2nd. McGraw-Hill Companies, Inc.
- Kolmogorov, A N (1968). "Local Structure of Turbulence in an Incompressible Viscous Fluid at Very High Reynolds numbers". In: *Soviet Physics Uspekhi* 10.6, p. 734. URL: <http://stacks.iop.org/0038-5670/10/i=6/a=R02>.
- Mugnier, Laurent M., Amandine Blanc, and Jérôme Idier (2006). "Phase Diversity: A Technique for Wave-Front Sensing and for Diffraction-Limited Imaging". In: ed. by Peter Hawkes. Vol. 141. *Advances in Imaging and Electron Physics*. Elsevier, pp. 1–76. DOI: [https://doi.org/10.1016/S1076-5670\(05\)41001-0](https://doi.org/10.1016/S1076-5670(05)41001-0). URL: <http://www.sciencedirect.com/science/article/pii/S1076567005410010>.
- Noll, Robert J. (1976). "Zernike Polynomials and Atmospheric Turbulence". In: *J. Opt. Soc. Am.* 66.3, pp. 207–211. DOI: [10.1364/JOSA.66.000207](https://doi.org/10.1364/JOSA.66.000207). URL: <http://www.osapublishing.org/abstract.cfm?URI=josa-66-3-207>.
- Obukhov, A. M. (1949). "Structure of the temperature field in turbulent flow". In: *Ser. Geograf. Geofiz.* 13.1, pp. 58–69.
- Ragazzoni, Roberto (1996). "Pupil plane wavefront sensing with an oscillating prism". In: *Journal of modern Optics* 43.2, pp. 289–293.
- Rodier, François (1988). "Curvature sensing and compensation: a new concept in adaptive optics". In: *Appl. Opt.* 27, pp. 1223–1225.
- Shack, R. V. and B. C. Platt (1971). "Production and use of a lenticular Hartmann screen". In: *Spring Meeting of the Optical Society of America*. Vol. 61, 656–660.
- Thorlabs (2017). *Principles of Spatial Filters*. Thorlabs. URL: https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=1400.
- (2018). *Shack-Hartmann Wavefront Sensor*. Thorlabs. URL: https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=2946.
- V.I. Tatarski, Richard A. Silverman (1961). *Wave Propagation in a Turbulent Medium*. Ed. by McGraw-Hill. McGraw-Hill.
- Wikipedia (2018). *Zernike polynomials*. Wikipedia. URL: https://en.wikipedia.org/wiki/Zernike_polynomials.