



# **Evaluation of Phase Retrieval in a Compressive Computational Ghost Imaging Setup**

**Juan Andrés Urrea Niño**

Undergraduate Monograph  
Physics

Director:  
Alejandra Valencia, PhD

Los Andes University  
Sciences Faculty, Physics' Department  
Bogotá D.C., Colombia  
June 1, 2017

## Acknowledgements

First of all, I would like to thank my parents and sister for their constant, unconditional support. Without them, all of this would not have been possible.

Thanks to my friends, without whom these 4 years would not have been as enjoyable.

Special thanks to my director Alejandra Valencia for many useful conversations regarding this work and much more, and to the entire Quantum Optics group for their help inside and outside the laboratory.

---

*"But nature is always more subtle, more intricate, more elegant than what we are able to imagine."*

Carl Sagan, *The Demon-Haunted World: Science as a Candle in the Dark*.

---

## Abstract

Ghost imaging has been developed into different setups to use the correlation properties of light to obtain the image of an object. Its computational version has been combined with Compressive Sensing algorithms to recover images with far less data than the Nyquist limit dictates. However, the retrieval in the image of certain properties of the object, such as its Fourier phase spectrum, has not been studied using ghost imaging setups. This monograph precisely presents this type of studies. Specifically, a new compressive computational ghost imaging setup is introduced, and implemented in the optical table, obtaining high quality images. Additionally, a procedure to indirectly measure and quantify the phase retrieval is proposed, explained and implemented. This procedure allows to confirm the successful recovery of phase spectrum information through compressive sensing methods in computational ghost imaging.

## Resumen

El experimento de Ghost Imaging se ha desarrollado en diferentes montajes para usar las propiedades de correlación de la luz para obtener la imagen de un objeto. La versión computacional se ha combinado con algoritmos de muestreo compresivo para recuperar imágenes con cantidades de datos mucho menores a la requerida por el límite de Nyquist. Sin embargo la recuperación en la imagen de ciertas propiedades del objeto, como su espectro de fase de Fourier, no se ha estudiado usando montajes de Ghost Imaging. Esta monografía presenta este tipo de estudio. Específicamente, un nuevo montaje para Ghost Imaging computacional compresivo se presenta, y se implementa en la mesa óptica, obteniendo imágenes de alta calidad. Adicionalmente se propone, explica e implementa un procedimiento para medir indirectamente y cuantificar la recuperación de la fase. Este procedimiento permite confirmar la recuperación exitosa de la información del espectro de fase a través de métodos de muestreo compresivo en un montaje de Ghost Imaging computacional.

---

# Contents

<b>Chapter 1</b>	<b>Introduction</b>	<b>7</b>
1.1	About this Work, the State-of-the-art and its Importance . . . . .	7
<b>Chapter 2</b>	<b>Theoretical Background</b>	<b>9</b>
2.1	Importance of the Fourier Phase Spectrum in signal processing . . . . .	9
2.2	Standard Ghost Imaging . . . . .	12
2.2.1	Cross-Correlation Method . . . . .	13
2.2.2	Modulation Method . . . . .	14
2.3	Computational Ghost Imaging . . . . .	15
2.4	Compressive Alternative Computational Ghost Imaging . . . . .	16
2.5	Compressive Sensing . . . . .	18
2.5.1	Basis Pursuit Algorithm . . . . .	19
2.5.2	Total Variation Algorithm . . . . .	19
2.6	Compressive Sensing in Ghost Imaging . . . . .	20
2.6.1	Basis Pursuit Algorithm Implementation . . . . .	20
2.6.2	Total Variation Algorithm Implementation . . . . .	23
2.7	Image Comparison Techniques . . . . .	24
2.8	Phase Information Comparison Technique . . . . .	25
<b>Chapter 3</b>	<b>Experimental Setup and Results Obtained</b>	<b>27</b>
3.1	Standard Ghost Imaging Setup . . . . .	27
3.1.1	Experimental Results: Cross-Correlation Method . . . . .	29
3.1.2	Cross-Correlation Method with Simulated Object Path . . . . .	30
3.2	Alternative Computational Ghost Imaging . . . . .	31
3.2.1	Experimental Results: Cross-Correlation Method . . . . .	32
3.2.2	Cross Correlation Method with Simulated Object Path . . . . .	33
3.2.3	Experimental Results: Modulation Method . . . . .	34
3.3	Compressive Alternative Computational Ghost Imaging . . . . .	35
3.3.1	Experimental Results: Modulation Method with TV algorithm . . . . .	35
3.3.2	Application of Image Comparison Methods . . . . .	37
3.3.3	Phase Analysis . . . . .	38
3.3.4	Experimental Results: Modulation Method with BP algorithm . . . . .	40
3.3.5	Application of Image Comparison Methods . . . . .	41

3.3.6	Phase Analysis . . . . .	43
<b>Chapter 4</b>	<b>Conclusions and further discussion</b>	<b>47</b>
4.0.1	Conclusions . . . . .	47
4.0.2	Further Discussions . . . . .	49
<b>Chapter 5</b>	<b>Appendices</b>	<b>51</b>
5.1	SLM as Amplitude Modulator . . . . .	51

---

# Chapter 1

## Introduction

### 1.1 About this Work, the State-of-the-art and its Importance

Treatment of large amounts of data has always been a major issue, specially in signal sampling processing. The processing and analysis of such data requires significant time and equipment. This limitation comes mainly from the Nyquist theorem, which states the minimum amount of data required to successfully recover a sampled signal [1]. It was not until ten years ago that ways to achieve successful signal recovery, with far less data than the Nyquist limit, were discovered by Candès, Tao, Romberg and Donoho, and the area of Compressive Sensing was born [2, 3]. This new area soon caused a controversy as it appeared to violate the previously mentioned sampling theorem. However, such violation is a simple misunderstanding since compressive sensing works only in some restrictive cases in which there is actually no violation of Nyquist theorem. Several algorithms of Compressive Sensing are nowadays widely studied and used in different applications such as holography, microscopy, image processing and Terahertz imaging. [4, 5, 6]. Between the most efficient algorithms of compressive sensing, there are the *Total Variation algorithm (TV)* [7] and the *Basis pursuit algorithm (BP)*[3]. These two methods have a remarkable image recovery capability for binary images. In particular, the TV is able to achieve successful image processing with as few as 20% of the required amount of data imposed by Nyquist theorem [8].

Imaging is, in its core, a signal sampling process. In a typical imaging experiment, the image of an object is obtained. The object and the image can be represented by a discrete 2-dimensional real valued signal. However, while the image has a definite resolution, given by the pixels of the camera that is used to perform the process of imaging, the object can have an arbitrary resolution. It is possible to obtain the Fourier transform of both of these signals, obtaining the corresponding magnitude and phase spectrum. This Fourier phase spectrum contains vital information regarding the signals. As several works have proven, this spectrum has a huge impact on our visual perception.[9, 10].

A successful recovery of the object's phase spectrum can be done by sampling according to the Nyquist theorem. Compressive sensing algorithms have been successful for the formation of an image. However, this does not imply that compressive sensing is useful for phase retrieval. Therefore, it becomes interesting to

test if Compressive sensing algorithms can do such a thing. The main objective of the present monograph is precisely to test the capabilities of compressive sensing for phase spectrum retrieval. In particular, we are interested in proving Compressive Sensing in a Ghost Imaging (GI) setup. GI consists of imaging an object without strictly looking at it. This means, focusing not on the object itself and its properties, but on the properties of the light that is used to observe it. In particular, in the spatial second order correlations of the light. GI experiments were first implemented in the context of quantum mechanics using pairs of entangled photons i.e, light with quantum correlations [11]. Years later, a GI implementation with pairs of classically correlated light started a rich discussion about the role of entanglement in this type of experiments [12, 13]. This discussion came to an end with an experiment that did not use pairs of photons at all and therefore ruled out completely the necessity of entanglement in GI experiments [14].

Compressive Sensing algorithms have also been applied to GI, reducing both amount of data [15] and, in its computational version, the amount of equipment [16, 17]. Recent results demonstrated that compressive sensing algorithms can be applied to obtain high-quality images with practical uses in 3D scenery imaging[18] and moving-component imaging [19], to mention just some of them. However, to the best of our knowledge no study of the recovery of phase spectrum in a GI setup using compressive sensing has been done. The goal of this thesis is precisely to provide an experimental study to determine the applicability of Compressive sensing in GI to retrieve the phase spectrum of an object via its image's phase spectrum. This type of study is relevant to characterize the use of Compressive Sensing methods and its range of application.

This document is organized as follows: Chapter 2 presents a theoretical background regarding the amplitude and phase of a signal, ghost imaging, its computational version and compressive sensing. In chapter 3 the experimental implementations and the results are presented. The conclusions and a final discussion are presented in chapter 4, and chapter 5 contains an appendix relevant to this work.



## Chapter 2

# Theoretical Background

This chapter focuses on giving an insight to the basic concepts in the Fourier analysis of finite discrete real-valued signals and its importance regarding the information about the signal it stores in the magnitude and phase spectra it generates. After showing the notable importance of the phase spectrum in signal recovery, compared to the magnitude spectrum, basic concepts of signal sparsity and compressive sensing are explored, showing how they relate in the study done in this work. For sparsity, a statistical study is done to see the appearance of this characteristic in the Fourier Transform magnitude of typical images, while showing that the phase spectrum most often does not present it. The basics of Ghost Imaging (GI) in its different variations are explored and explained, together with a new alternative that, to the best of our knowledge, has not been implemented until now.

### 2.1 Importance of the Fourier Phase Spectrum in signal processing

Fourier analysis is a currently used technique for signal processing which allows to take the original signal, normally considered to exist in the Time Domain, to its transform in the corresponding Frequency Domain. Considering time instants as the discrete variable  $n = 1, 2, 3, \dots, N$ , experiments generally work with signals defined as N-tuples of the form  $x_n = \{x_1, x_2, \dots, x_N\}$  such that  $x_n \in \mathbb{R}^N$  is a discrete finite real-valued one dimensional signal. One dimensional signals that can be processed computationally, such as basic audio, are of this kind. For these signals, the discrete Fourier Transform (DFT) is defined as:

$$X_k = \sum_{n=1}^N x_n e^{-i \frac{2\pi k n}{N}}, \quad (2.1)$$

each element of  $X_k \in \mathbb{C}^N$  may be written in polar representation, displaying its magnitude and phase:

$$X_k = \{|X_1|e^{i\theta_1}, |X_2|e^{i\theta_2}, \dots, |X_N|e^{i\theta_N}\} \quad (2.2)$$

Eq. 2.2 shows that, from the DFT, two discrete finite real valued signals can be obtained; one containing the magnitudes  $|X_k|$  and another containing the phases  $\theta_k$ . Those are, respectively, the *magnitude spectrum* and the *phase spectrum* of the original signal  $x_n$ .

A similar process allows to find these spectra from a two-dimensional signal, such as an image,  $x_{m,n} \in$

$Mat_{M \times N}(\mathbb{R})$ , where  $Mat_{M \times N}(\mathbb{R})$  denotes the set of  $M \times N$  matrices of real-valued entries. Considering  $m$  and  $n$  as discrete position variables that satisfy  $m = 1, \dots, M$  and  $n = 1, \dots, N$ , the two dimensional DFT will be in terms of the corresponding spatial frequency discrete variables  $p$  and  $q$ , which run over the same intervals as their position counterparts. The magnitude and phase spectra would now be  $|X_{p,q}|, \theta_{p,q} \in Mat_{M \times N}(\mathbb{R})$ .

Under these definitions, all the information from any signal is stored in both spectra. The question on which of the two generated spectra contains more information about the signal is answered in [9, 10, 20], where it is said that the phase spectrum of a signal contains the majority of its vital information. Such a conclusion might have been imagined, as the magnitude spectrum only shows the "weight" of a given frequency in the signal, so all the other characteristics of the signal must be in the phase spectrum. To support this statement, computational experiments performed by [20] will be replicated here as a practical demonstration that the phase spectrum is what influences most of our perception, mainly because it contains the spatial distributions of each spatial frequency in the image. It must be noted that, for computer calculations, the DFT operation is calculated via the fast Fourier Transform (FFT), with its corresponding Inverse Fast Fourier Transform (IFFT).

The experiments consist in taking a grayscale image and obtain its magnitude and phase spectrum. To identify which spectrum contains the most information relevant to visual perception, the image will be reconstructed from these spectra in two different ways. One way, denoted as Magnitude Conservation (MC), consists of taking the original magnitude spectrum and replacing the original phase spectrum with one consisting of a constant matrix whose value is the average of the original one, generating an image matrix via the IFFT. Another way, denoted as Phase Conservation (PC), consists of taking the original phase spectrum and replacing the original magnitude spectrum with one consisting of a constant matrix whose value is the average of the original one, generating an image via the IFFT. Comparing the images obtained by both MC and PC methods allows to determine which spectrum generates the best reconstruction in terms of human visual perception and ability to recognize the reconstructed image. A graphic explanation of this procedure can be seen in figures 2.1 and 2.2, similar to how it is explained in [20].

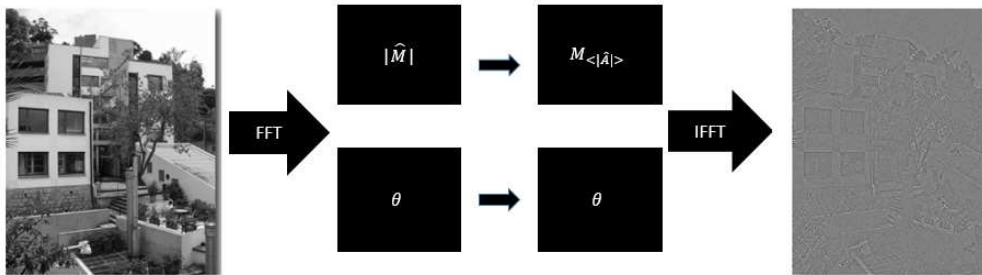


Figure 2.1: Graphical description of the PC method. Both spectra are obtained by applying the FFT. The magnitude spectrum matrix  $|\hat{M}|$  is replaced by a matrix of constant elements all equal to the average of  $|\hat{M}|$ , denoted  $M_{\langle |\hat{A}| \rangle}$ . The phase spectrum matrix  $\theta$  remains the same. A complex valued matrix is formed with the matrices and the recovered image is generated from it by applying the IFFT. Image courtesy of the Physics Department in Universidad de los Andes.

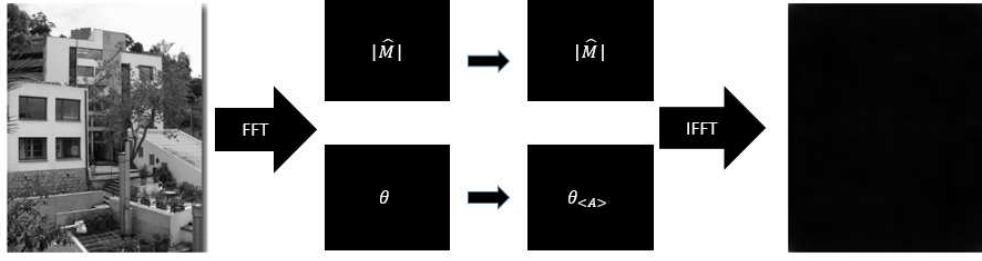


Figure 2.2: Graphical description of the MC method. The phase spectrum matrix  $\theta$  is replaced by a matrix of constant elements all equal to the average of  $\theta$ , denoted  $\theta_{\langle \hat{A} \rangle}$ . The magnitude spectrum matrix  $\theta$  remains the same. A complex valued matrix is formed with the matrices and the recovered image is generated from it by means of pplying the IFFT.

Figure 2.1 shows that keeping the original phase spectrum intact, while almost completely ignoring the information in the original magnitude spectrum, allows to obtain a reconstructed image that resembles the original one enough for visual perception to identify. Keeping only the information contained in the magnitude spectrum fails to generate a reconstructed image that bears any resemblance to the original one, as it can be seen in figure 2.2.

A similar procedure can be done now involving two different images A and B. Each one generates two spectra and they are combined; the magnitude spectrum of A and phase spectrum of B form a new complex valued matrix, while the magnitude spectrum of B and phase spectrum of A form another. Applying the IFFT, two matrices in the position variables  $m$  and  $n$  are obtained. Figure 2.3 displays a graphic explanation of this procedure, similar to how it is explained in [20]. The results displayed in figure 2.3 allows us to see that only the image that contributed with its phase spectrum remains distinguishable.

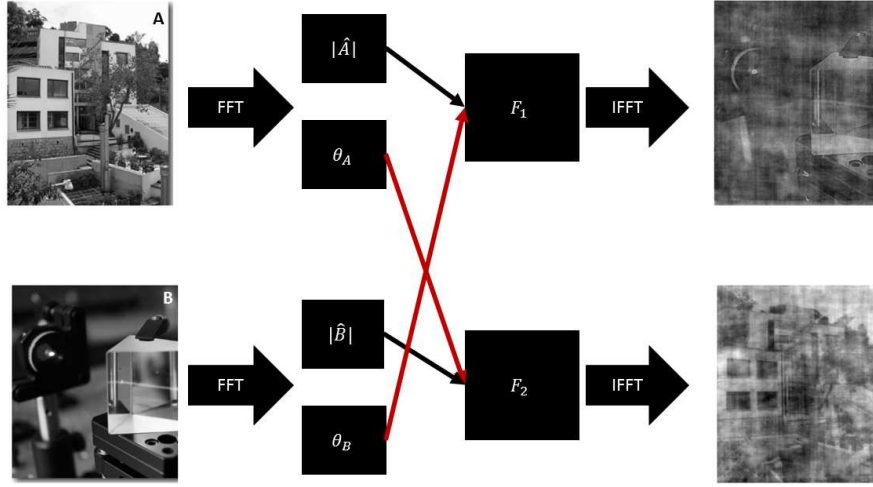


Figure 2.3: Procedure done to combine magnitude and phase spectrum of two different images. Each combined pair of spectra become a complex valued matrix as each magnitude and phase make up a complex number. These resulting complex matrices are denoted as  $F_1$  and  $F_2$ . The IFFT is applied to each one, allowing to obtain 2 new images from the combination of spectra. Image A generates magnitude spectrum  $|\hat{A}|$  and phase spectrum  $\theta_A$ , while image B generates magnitude spectrum  $|\hat{B}|$  and phase spectrum  $\theta_B$ . Images courtesy of the Physics Department in Universidad de los Andes.

The results displayed in figures 2.1 and 2.3, similar to the ones obtained by [9, 20], show the rather notable importance of the phase spectrum in image reconstruction, due to the amount of information it stores about the image and how our perception focuses on the phase spectrum. In this sense, information contained in the phase spectrum should always remain a priority when seeking to analyse or recover a signal. Specifically for this work, the use of a compressive sensing algorithm in a computational ghost imaging setup will be tested for the recovery of the phase spectrum information of an intensity mask object.

## 2.2 Standard Ghost Imaging

Ghost Imaging (GI) is an optical technique developed in the quantum realm that has been given practical application in recent years. Its importance resides in its intriguing nature, which allows to obtain an image not like in standard imaging, but by measuring the correlation between light beams or the modulation of a known light beam by an object. In standard lensless GI, shown in figure 2.4, light is separated into two beams and sent through two different paths. One beam is sent through an object and detected by a bucket detector avoiding the reconstruction of the image only by looking at it. This path is called the Object Arm. The other beam passes through what is called the reference arm in which there is no object and it is finally detected. This path is called the Reference Arm. The intensity measurements obtained in the object path and the light beam patterns obtained in the reference path can be used to generate an image of the object by either of the methods described above, i.e measurement of correlation or light modulation.

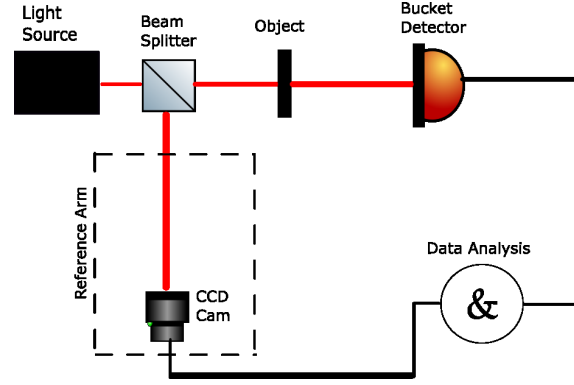


Figure 2.4: A standard lens-less GI setup. Object path and reference path can be seen.

For this work, the light used is pseudo-thermal, generated by a laser hitting on a Rotating Ground Glass (RGG). This setup, displayed in figure 2.5, allows the creation of a dispersed beam which then divides into two arms with propagating speckle-fields. Previous experiments have proved the usefulness of this kind of light for GI [21, 15, 13].

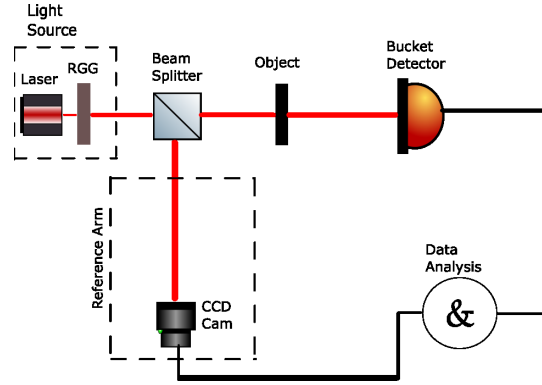
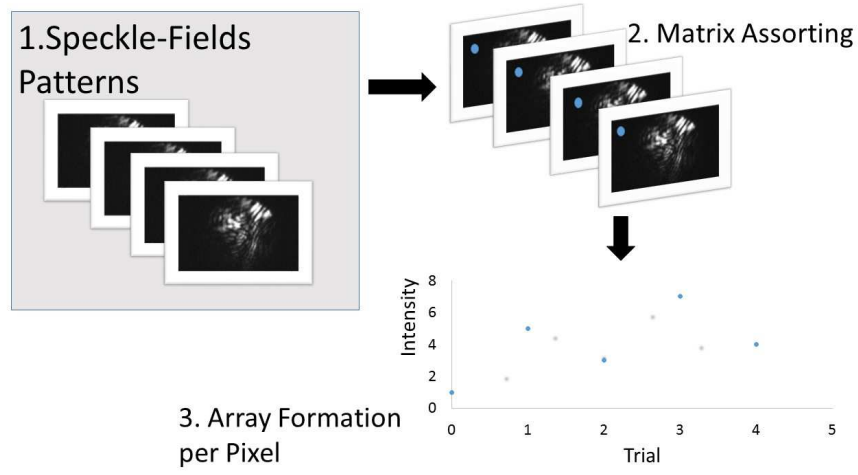


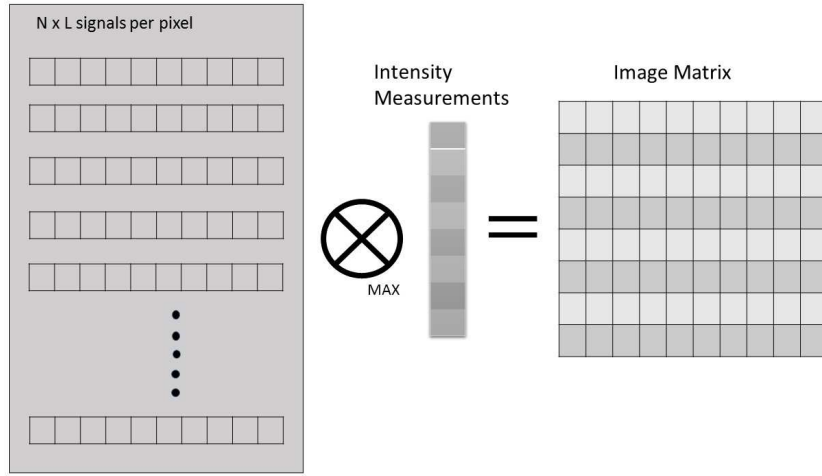
Figure 2.5: A GI setup using pseudo-thermal light. Object path and reference path can be seen.

### 2.2.1 Cross-Correlation Method

In this method,  $M$  different photos of the speckle-fields in the reference arm are aligned in an  $N \times L \times M$  array, where  $N \times L$  is the size of each photo. Each pixel position now has an intensity vs trial number signal associated with it of size  $1 \times M$ . Each of these signals are cross-correlated with the intensity signal measured by the bucket detector, size  $1 \times M$ , by using a normalized cross-correlation operation and obtaining the maximum correlation coefficient. Each of these coefficients is associated with the corresponding pixel position in a  $N \times L$  matrix and displayed as an image. The process of array formation per pixel is depicted in figure 2.6a and the cross-correlation and image formation method is displayed in figure 2.6b.



(a)



(b)

Figure 2.6: Cross-correlation method. **(a)** Array formation per pixel process. The photos of the speckle-fields are captured in step 1. They are arranged in a three dimensional array in step 2 and an intensity vs trial array per pixel is formed in step 3. **(b)** Cross-correlation and image formation process. For each pixel there exists an intensity vs trial signal which is cross correlated with the intensity measurements of the bucket detector. The maximum correlation coefficient is then assigned to the corresponding pixel in the new image matrix.

### 2.2.2 Modulation Method

In this method, since the CCD camera displayed in figure 2.5 is located at the same distance from the beam-splitter as the object, the speckle-fields that hit upon these two elements are the same. This known speckle-field is then modulated by the object and the resulting modulated light is collected by the bucket detector. This process is represented by:

$$\sum_{i,j} S_{ij} \circ T_{ij} = \sum_{i,j} M_{ij} = I, \quad (2.3)$$

where  $S_{ij}$  is the  $N \times L$  matrix of the speckle-field obtained by the CCD,  $T_{ij}$  is the  $N \times L$  binary matrix of the object,  $M_{ij}$  is the modulated speckle-field matrix and  $\circ$  denotes element-wise multiplication, i.e modulation. This corresponds to a single measurement in the experiment. For each measurement done in the experiment  $T_{ij}$  remains the same, while the speckle-field  $S_{ij}$  and the measured intensity  $I$  change. Following [15], each speckle-field matrix  $S_{ij}^k$ , with  $k = 1, 2, \dots, M$ , can be flattened into a single  $1 \times NL$  row vector,  $T_{ij}$  can be flattened into a  $NL \times 1$  column vector and each measured intensity  $I^k$  can become an element in a  $NL \times 1$  column vector. To create a fully determined equation system that can be solved by matrix inversion,  $N \times L$  measurements would be necessary such that each independent row vector  $S^k$  can be placed in a  $NL \times NL$  matrix and the system can be written as:

$$\begin{pmatrix} S_1^1 & \dots & S_{NL}^1 \\ \vdots & \ddots & \vdots \\ S_1^{NL} & \dots & S_{NL}^{NL} \end{pmatrix} \begin{pmatrix} T_1 \\ \vdots \\ T_{NL} \end{pmatrix} = \begin{pmatrix} I^1 \\ \vdots \\ I^{NL} \end{pmatrix} \quad (2.4)$$

This system can be solved to find the column vector  $\vec{T} = (T_1, T_2, \dots, T_{NL})$  via matrix inversion, as the different speckle-fields are assumed to be independent. If the matrix of speckle-fields is denoted  $S$  and the vector  $\vec{I}$  denotes the vector of measured intensities, the system can be written as:

$$S\vec{T} = \vec{I}, \quad (2.5)$$

and the solution to  $\vec{T}$  is obtained as:

$$\vec{T} = S^{-1}\vec{I}. \quad (2.6)$$

A graphical depiction of this process is shown in figure 2.7.

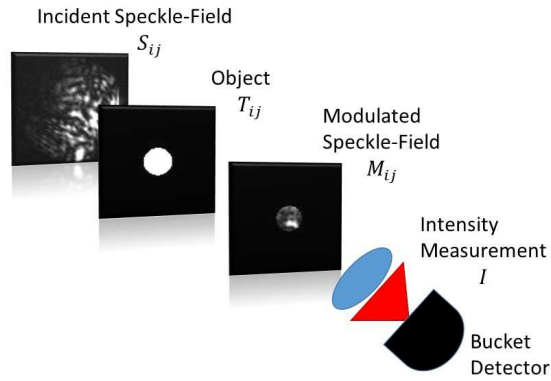


Figure 2.7: Modulation method. A single measurement of intensity via the modulation method. This is repeated  $M$  times in the experiment.

## 2.3 Computational Ghost Imaging

It can be noted that the reference arm for the standard GI setup, shown in figure 2.8, consists of a light beam propagating through air, a classical phenomenon that can be simulated by a computer using Fresnel's

propagation theory. Replacing the reference arm for a simulation is the setup known as Computational Ghost Imaging (CGI) [14], shown in figure 2.8. CGI allows the simulated electric field data to be obtained before, during or after the data from the object arm is generated, eliminating the need of the Reference Arm. Similarly, less opto-electronical elements are required on the optical table, simplifying the original setup and reducing to half the amount of experimental data generated. The pseudo-thermal light is now generated by a laser reflected on a Spatial Light Modulator (SLM). The resulting beam comes out with the phases given by the SLM, which allows to have initial phase conditions for the field to be simulated. To generate an image in this setup the same modulation and correlation methods explained before can be used, since again the speckle-field that hits upon the object is known thanks to a simulated reference arm.

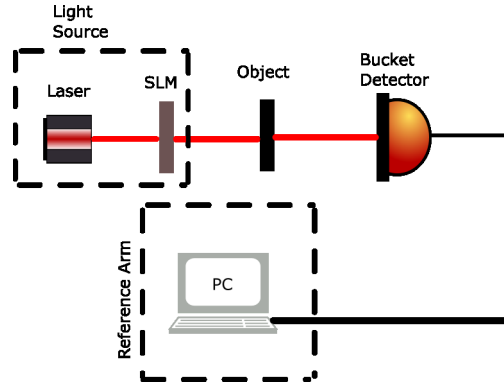


Figure 2.8: A CGI setup using pseudo-thermal light. Reference path is replaced by a computer simulation.

## 2.4 Compressive Alternative Computational Ghost Imaging

The setup displayed in figure 2.8 can be modified, as indicated by [14], by replacing the light source with a laser that passes through a polarizer P1 and a beam expander (BE) before hitting on an SLM. The beam reflected from the SLM then passes through a second polarizer P2 and through a convergent lens F of focal length  $f$  a distance  $2f$  from the SLM. The light beam is focused to a 50/50 beam splitter located a distance  $f$  from the lens. On one arm coming out of the BS the object is placed a distance  $f$  from the BS, followed by a bucket detector, forming the Object Arm. On the other arm a CCD camera is placed a distance  $f$  from the BS, forming the Reference Arm. Thanks to the lens positioning, an "SLM-Lens" imaging system is formed, displaying the image of the SLM's screen on both the object and the CCD camera, where the SLM is now acting as an amplitude modulator thanks to the presence of polarizers P1 and P2 (See Annex 1), displaying previously prepared pseudo-random binary matrixes. The resulting setup is displayed in figure 2.9.



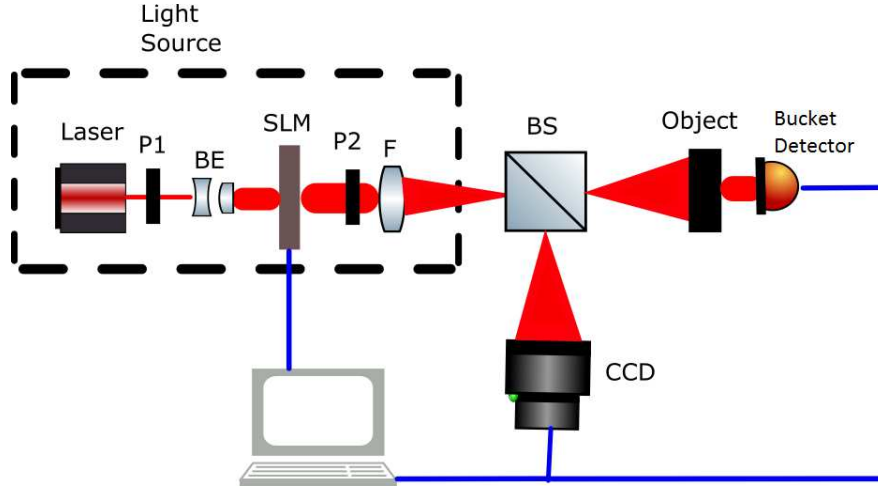


Figure 2.9: The SLM Ghost Imaging setup, described by [14], modified in this work.

The computational counterpart of this setup can now be constructed. To do this, the reference arm must be removed from the setup. Contrary to traditional CGI [14], where the reference arm is replaced by the computational simulation of Fresnel propagation of light through air, no simulations are needed in this alternative. This is due to the fact that the image captured by the CCD in the reference arm corresponds to the matrix displayed on the SLM's screen. This matrix is known and therefore there is actually no need for a reference arm. The setup then becomes the one depicted in figure 2.10.

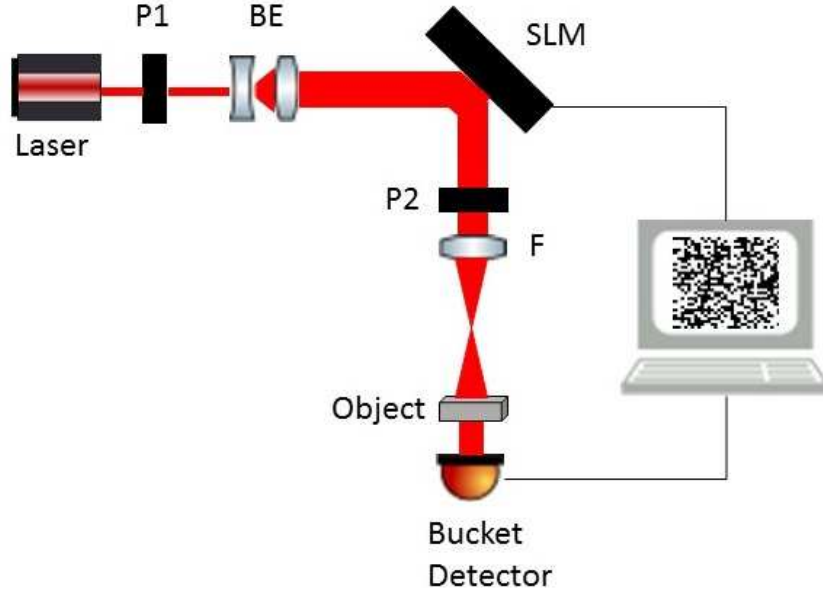


Figure 2.10: Alternative computational ghost imaging proposed in this work.

For this case, the correlation methods can be employed to generate an image of the object. This setup will, however, include an error associated with the difference between the image of the SLM's screen that is displayed on the object and the actual matrix that is sent to the SLM for display. This limits the use of the modulation method, which assumes the light field of the reference arm is exactly what hits upon the object,

which is not the case here. To deal with this problem and be able to apply the modulation method, the setup in figure 2.10 is modified. This modification consists in replacing the object and bucket detector with a CCD camera. The resulting setup is displayed in figure 2.11. Here, the CCD captures the image that comes from the SLM-Lens system. The object used in figure 2.10 is replaced by a virtual object, i.e a binary matrix with arbitrary shape. Each photo is then modulated by the virtual object and summed over all its elements to obtain an equivalent intensity measurement. This process guarantees that the modulation method can be used to generate an image of the virtual object.

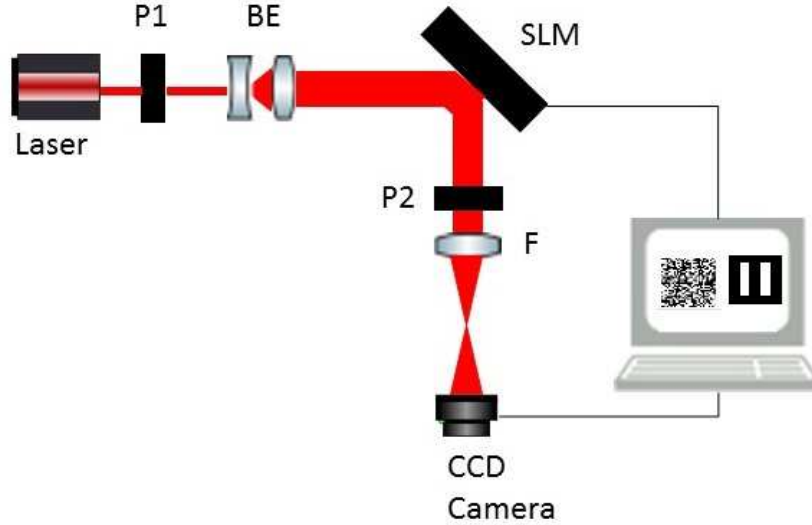


Figure 2.11: Alternative computational ghost imaging using a virtual object.

In the case where the pseudo-random patterns sent to the SLM are binary, with pixel values of 0 or 1, the values of  $\theta$  are, accordingly,  $0^\circ$  and  $90^\circ$  (See Appendix 1). Therefore, the observed intensity hitting on the object has a value of 0 when the pixel has a value of 0 and an approximately constant value when the pixel has value of 1. This is the ideal case, when  $E(x, y)$  is considered constant on the SLM's screen. This way the pseudo-random pattern sent to the SLM is displayed on the object up to a proportionality constant of value  $|T_1 R_{SLM} T_2 E_H|^2$  (See Appendix 1).

## 2.5 Compressive Sensing

When working in signal sampling and recovery, the Nyquist theorem plays an essential role assuring that the signal is appropriately recovered. All details covered in this section will be for a one dimensional continuous time function  $x(t)$  for simplicity, yet all properties can be taken to the domain of n-dimensional finite discrete functions, as is the case of 2 dimensional images as matrices. If  $x(t)$  has a fourier transform  $X(\omega)$  that satisfies  $X(\omega) = 0 \forall |\omega| > \frac{\Omega}{2}$ , where  $\Omega$  is a frequency, then  $x(t)$  can be recovered perfectly if it is sampled at a rate of  $\frac{\Omega}{2\pi}$  [1]. Given the size of most signals used in sampling experiments, satisfying the Nyquist theorem generally implies having to acquire a huge amount of data. For example, when dealing with a one-dimensional discrete

signal with duration of  $N$  seconds,  $M = N \times \frac{\Omega}{2\pi}$  samples are required.

Compressive Sensing (CS) refers to any sampling method that allows a perfect reconstruction of a signal from a number of samples that is significantly smaller than the ones required by the Nyquist theorem. As presented by [2, 3, 22], this reconstruction requires certain conditions to be satisfied by the signal that will be recovered, such as sparsity in a given basis. Sparsity denotes the property of a signal to have relatively few non-zero elements in a specific basis. This sparse representation in some basis may be achieved in several ways; for example, by changing base via a matrix operation or by applying a defined operator on the original signal. The main problem can be expressed in the following way: a vector  $\vec{y} \in \mathbb{R}^M$  of samples was obtained from an original signal  $\vec{x} \in \mathbb{R}^N$  by means of a measurement matrix  $A \in \text{Mat}_{M \times N}(\mathbb{R})$  with  $M \ll N$ . The under-determined system that results is shown below:

$$A\vec{x} = \vec{y} \quad (2.7)$$

and it is desired to find  $\vec{x}$  given  $A$ , a matrix that contains the information on how the measurement was done, and  $\vec{y}$ , the vector of experimentally measured data.

### 2.5.1 Basis Pursuit Algorithm

In order to find a unique solution to recover  $\vec{x}$ , its sparsity in a given basis is a key feature that is used in several CS algorithms. The *Basis Pursuit* (BP) algorithm [3, 22] is an example that uses basis change to achieve sparsity of the studied signal  $x$ . A basis  $\{\Phi_1, \Phi_2, \dots, \Phi_N\}$  where  $x$  has a sparse representation is chosen, in such a way that:

$$\vec{x} = \Phi^{-1}\vec{s}, \quad (2.8)$$

where  $\vec{s}$  is a sparse representation of  $\vec{x}$ . Plugging eq. 2.8 into eq. 2.7 yields:

$$A\Phi^{-1}\vec{s} = \vec{y} \quad (2.9)$$

Eq 2.9 allows to find  $\vec{s}$  and therefore  $\vec{x}$ . However, this equation has no unique solution. To choose one, [2, 3, 22] apply an  $l_1$  minimization condition on the solution  $\vec{s}$ , defined as:

$$\|\vec{s}\|_1 = \sum_{i=1}^N |s_i|, \quad (2.10)$$

which results on a  $\vec{s}$  that becomes a  $\vec{x}$  corresponding to the original sampled signal.

### 2.5.2 Total Variation Algorithm

For the case of applying an operator on  $\vec{x}$  to find a sparse representation, the *Total Variation* (TV) algorithm [7, 8] consists on solving eq. 2.7 and choosing a solution  $\vec{x}$  by applying an  $l_1$  minimization on the gradient of  $\vec{x}$ , defined in [7] as:

$$(\nabla x)_{i,j} = \sqrt{(x_{i,j} - x_{i,j-1})^2 + (x_{i,j} - x_{i-1,j})^2} \quad (2.11)$$

The solution  $\vec{x}$  which satisfies this minimization corresponds to the original signal  $\vec{x}$ .

## 2.6 Compressive Sensing in Ghost Imaging

Starting with the system described by eq. 2.13, the goal is to recover  $\vec{T}$  when the number of measurements is  $M \ll N \times L$  and the system is sub-determined.

Setting a number of measurements  $M \ll N \times L$  to make the acquisition process more feasible results in a sub-determined system as follows:

$$\begin{pmatrix} S_1^1 & \dots & S_{N \times L}^1 \\ \vdots & \ddots & \vdots \\ S_1^M & \dots & S_{N \times L}^M \end{pmatrix} \begin{pmatrix} T_1 \\ \vdots \\ T_{N \times L} \end{pmatrix} = \begin{pmatrix} I^1 \\ \vdots \\ I^M \end{pmatrix} \quad (2.12)$$

This matrix equation can be written in short as follows:

$$S\vec{T} = \vec{I} \quad (2.13)$$

With  $S$  the matrix composed of all the  $M$  speckle-fields measured as rows,  $\vec{T}$  the column vector obtained by flattening the object's matrix and  $\vec{I}$  the column vector of measured intensities. Such system can be solved by applying either *BS* or *TV* algorithms.

### 2.6.1 Basis Pursuit Algorithm Implementation

To apply the *BS* algorithm, several conditions must be satisfied by both  $S$  and  $\vec{T}$ . For the case of  $S$ , it must satisfy two properties; the *Restricted Isometry Property* (RIP) and *Incoherence* with the basis  $\Phi$  that will be used for the algorithm. On one side, the RIP demands a conservation of  $l_2$  norm in the form:

$$1 - \epsilon \geq \frac{|\Theta \vec{u}|_2}{|\vec{u}|_2} \leq 1 + \epsilon, \quad (2.14)$$

where  $\epsilon > 0$  is a constant,  $\Theta = S\Phi^{-1}$  is the sampling matrix and  $\vec{u}$  is any vector that has the same non-zero entries as  $\vec{s}$  [22]. On the other side, *Incoherence* demands that the rows of  $S$  are not a sparse basis for the columns of  $\Phi^{-1}$  [22]. While these two conditions can be computationally verified for the case of  $S$ , an easier way consists on acknowledging that the speckle-fields that compose  $S$  are pseudo-random, therefore  $S$  can be considered a pseudo-random matrix. According to [22] this guarantees with high probability that  $S$  will satisfy the RIP and *Incoherence* properties.

For the case of  $\vec{T}$ , a basis where it is sparse must be found. Given that the objects used in GI are binary, it is complicated to intuitively find a basis where they satisfy sparsity. One way to approach this problem is to check if sparse basis that work for natural images, i.e common views such as landscapes, work also for binary images.

Following [23], it is known that the DFT magnitude spectrum of a natural image is approximately sparse, presenting an exponential decay. However, the phase spectrum is not sparse at all. This can be clearly seen when a collection of 20 natural images, courtesy of [24], are used to obtain their average magnitude and phase spectrum. Figure 2.12a shows an example of a natural image from the used set of images. Figure 2.12b shows the logarithm of the average magnitude spectrum of the set of images, while figure 2.12c shows the average phase spectrum of the whole set and figure 2.12d shows the phase spectrum of figure 2.12a. The expected

exponential decay can be observed in figure 2.12a. The average phase spectrum displayed in figure 2.12c shows a rather sparse composition, expected due to the apparently random behaviour of the phase spectrum of each image used, shown in 2.12d.

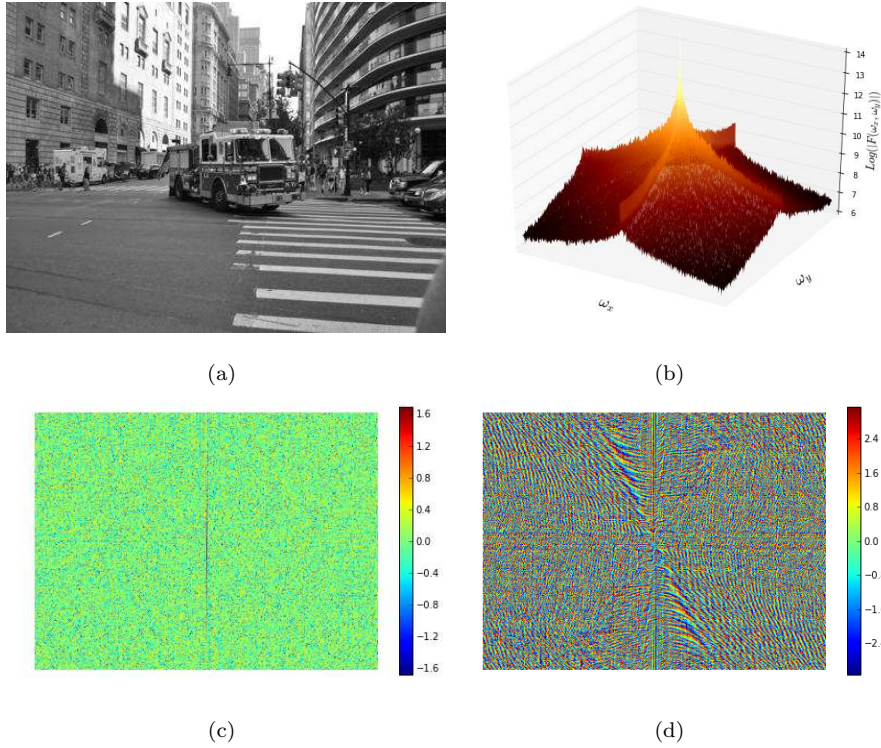


Figure 2.12: **(a)** Example of natural image. **(b)** Logarithm of the average magnitude spectrum of the image set. **(c)** Average phase spectrum of the set of images. **(d)** Phase spectrum of (a).

Another common transform for image treatment is the Discrete Cosine Transform (DCT). This transform only yields a magnitude spectrum, as the DCT transform of a real-valued signal such as an image is a real valued signal as well. Applying this transform to the same set of natural images as before allows to see that a higher level of sparsity is achieved in the DCT basis. This can be seen by comparing figures 2.12b and 2.13

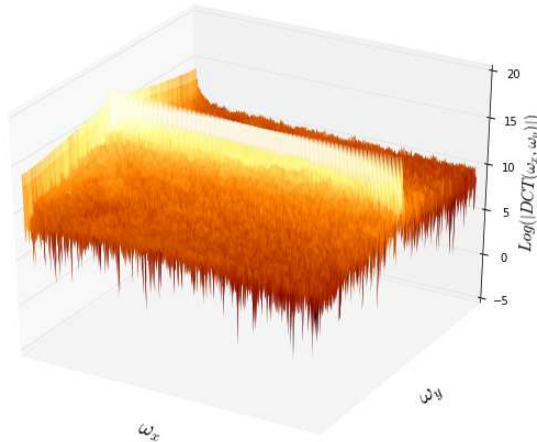


Figure 2.13: Logarithmic plot of the average DCT transform of the set of natural images used.

This property of being more sparse in the DCT basis than in the DFT basis is also satisfied by a binary image. An illustration of this is depicted in figure 2.14. Figure 2.14a shows the binary representation of a circular aperture. Figures 2.14b and 2.14c depict the magnitude spectrum of the DFT and the DCT, respectively. A quantification of sparsity is done by determining the percentage of the signal that is zero, ie. being lower than 0.01 to consider numerical precision errors, and the percentage of the signal that is different than zero. Figure 2.14(d-f) show this sparsity calculation for the binary object, the DFT magnitude and the DCT respectively.

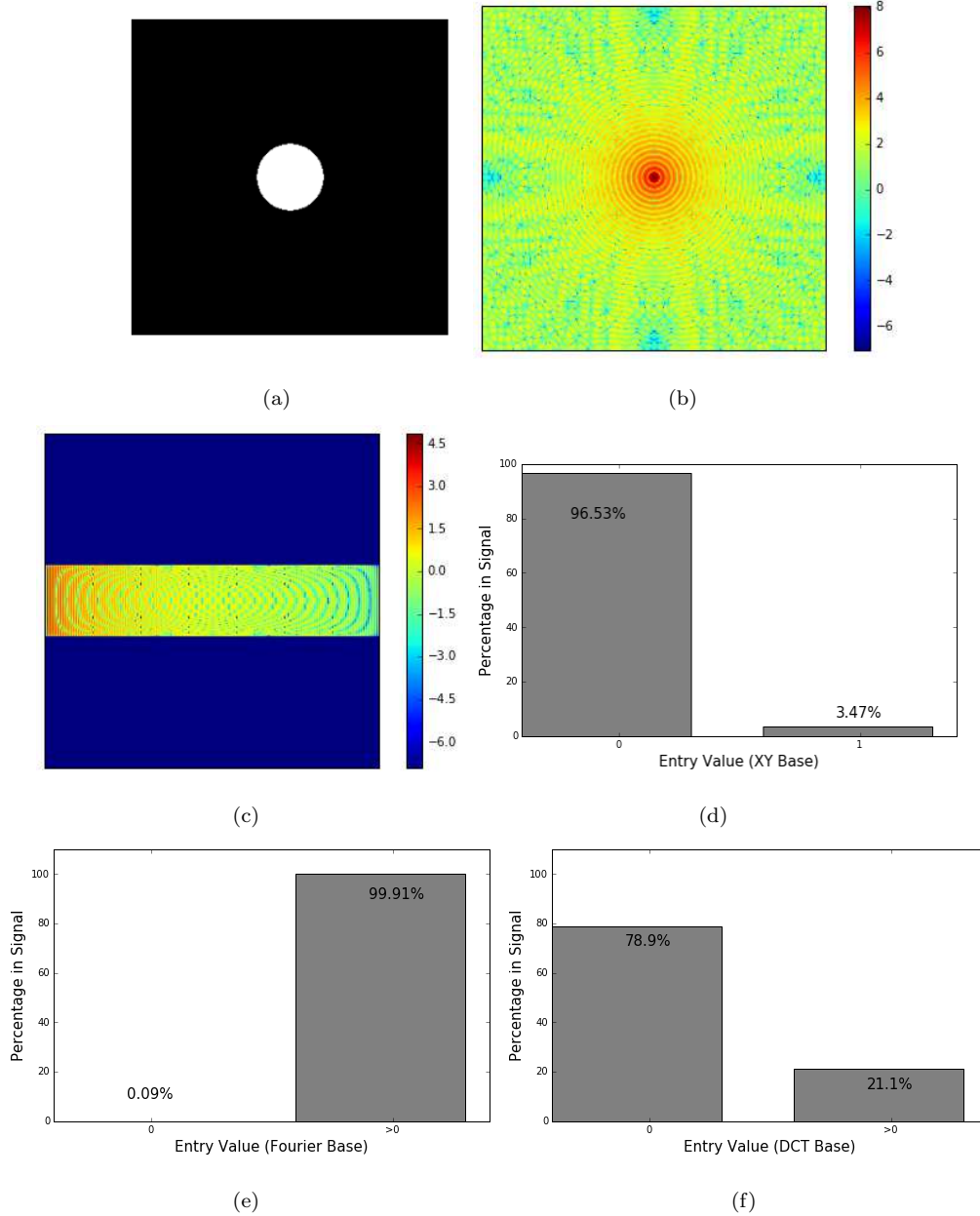


Figure 2.14: **(a)** Example of binary circular aperture. **(b)** Plot presenting the logarithm of the DFT magnitude spectrum of (a). **(c)** Plot presenting the logarithm of the DCT of (a). **(d)** Sparsity calculation for (a). **(e)** Sparsity calculation for (b). **(f)** Sparsity calculation for (c).

It is clear that the DFT is not a sparse basis and therefore is not a useful basis for BP. On the other hand,

the DCT presents a relatively high degree of sparsity and therefore will be used for the BP algorithm in this work. Another way of noticing the futility of the DFT is by means of the *Uniform Uncertainty Principle* (UUP) [25]. This principle states that if a signal is sparse in any basis, its DFT is not sparse.

### 2.6.2 Total Variation Algorithm Implementation

The *Total Variation* algorithm can also be applied to CGI, provided that the gradient of a binary image is sparse. Such condition can be seen in figure 2.15. Figure 2.15(a,b,c) depict three examples of binary images; a circular aperture, a double slit aperture and a  $\theta$  aperture, respectively. Figure 2.15(d,e,f) depicts the corresponding representations in the gradient basis. For a binary image sparsity in the gradient basis is easier to imagine than in the DCT basis. This can be understood since equation 2.11 depends only on the value difference between consecutive pixels. A binary aperture, in the gradient basis, will become even more sparse than in the original, since only the borders between 0 and 1 values will have a non-zero value. The sparsity calculation displayed in figure 2.15g shows the expected higher sparsity. These results, together with the ones obtained in image recovery by [7, 8] and in Pseudo-Thermal Ghost Imaging by [15], guarantee that this algorithm will be useful to generate the desired images in this work.

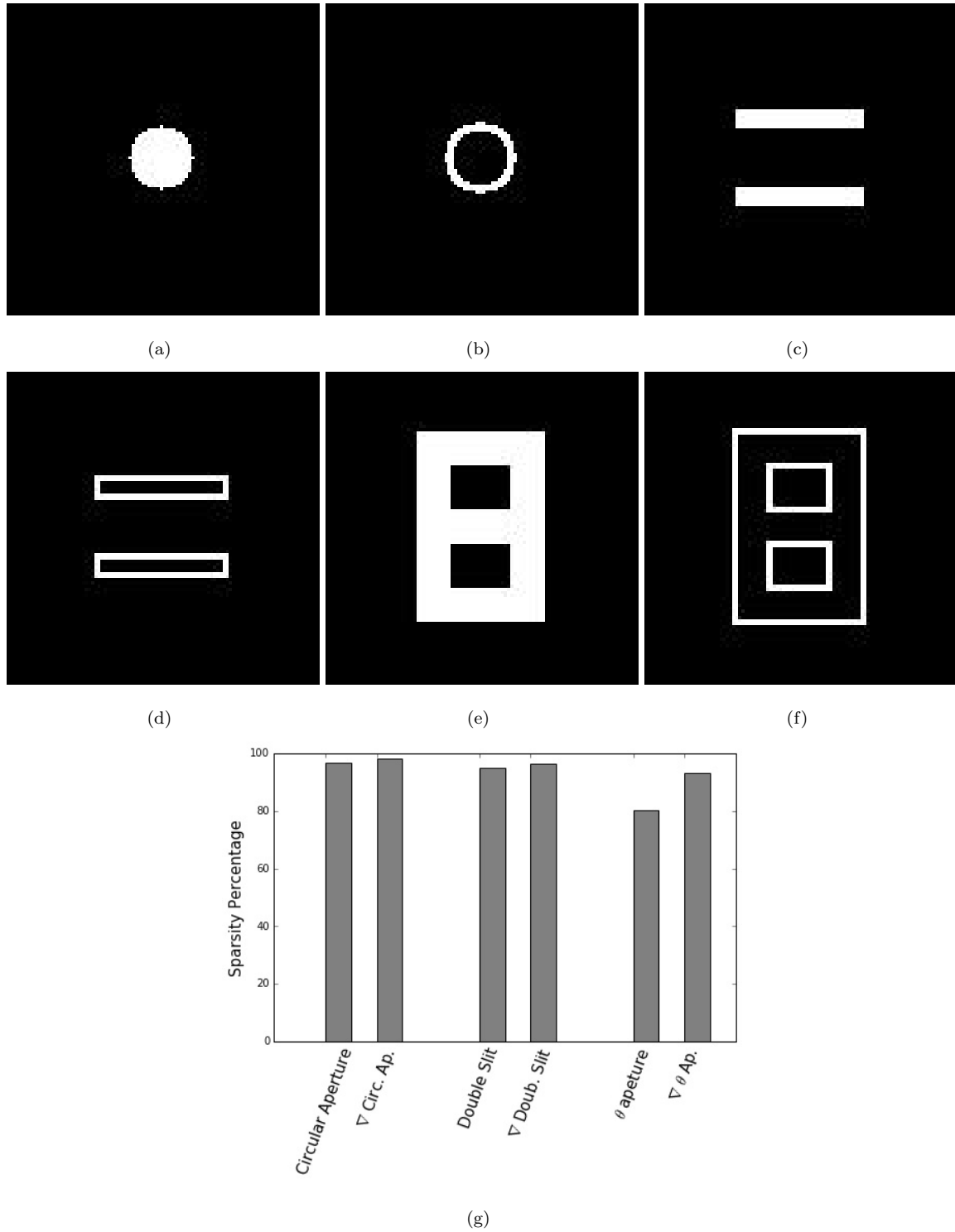


Figure 2.15: (a) Circular aperture. (b) Gradient magnitude representation of (a). (c) Double slit. (d) Gradient magnitude representation of (c). (e) Theta aperture. (f) Gradient magnitude representation of (e). (g) Sparsity calculation of each binary aperture and its corresponding gradient magnitude representation.

## 2.7 Image Comparison Techniques

Given that the obtained images will be compared with the matrix representation of the objects, the quantitative comparison methods used in this monograph to determine the success of the reconstruction are the



following:

- Signal-To-Noise Ratio (SNR)
- Structural Similarity Index Method (SSIM)

The *Signal-To-Noise Ratio* (SNR) is a widely known method for signal comparison. The noise  $X_{noise}$  between a given image  $X$  and a reference image  $X_R$  can be calculated as follows:

$$X_{noise} = X_R - X. \quad (2.15)$$

The SNR between  $X$  and  $X_R$  is calculated following the Matlab built-in function, defined as:

$$SNR = 10 * \log_{10} \left( \frac{\sum_{i,j} X_{ij}^2}{\sum_{i,j} ((X_{noise})_{ij})^2} \right) \quad (2.16)$$

The *SSIM* [26] is a novel method designed to perform image comparison imitating human visual perception in terms of image structure, luminance and contrast, displaying results more according to human visual perception than famous comparison techniques such as SNR and Mean Square Error [27]. The SSIM is defined as a similarity coefficient between images A and B as follows:

$$SSIM(A, B) = \frac{(2\mu_A\mu_B + C_1)(2\sigma_{AB} + C_2)}{(\mu_A^2 + \mu_B^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (2.17)$$

In equation 2.17  $\mu_i$  is the mean of the image  $i$ ,  $\sigma_i$  is the standard deviation of image  $i$ ,  $\sigma_{ij}$  is the correlation coefficient of images  $i$  and  $j$  and  $C_i$  are stability constants. This analysis is generally done as a mobile window over images A and B, obtaining a SSIM value per pixel and averaging these values to obtain a SSIM for the whole image.

## 2.8 Phase Information Comparison Technique

Since the goal of this work is to evaluate the retrieval of phase spectrum information, it is necessary to find a method that allows to quantitatively measure the similarity between a reference phase spectrum's information and an obtained phase spectrum's information. For this purpose, an indirect way to do this is proposed. A binary object, represented by a matrix  $O$ , is imaged. This generates an image matrix  $O'$  with the same size as  $O$ . Via the FFT magnitude and phase spectra are obtained for both  $O$  and  $O'$ , denoted  $|M|_O$ ,  $\Phi_O$  for  $O$  and  $|M|_{O'}$ ,  $\Phi_{O'}$  for  $O'$ . A complex matrix  $C$  is created by taking  $|M|_O$  and  $\Phi_{O'}$ . By applying the IFFT to  $C$ , a new image is obtained, denoted as  $I_{O'}$ ; a *Phase Image*.  $I_{O'}$  has the magnitude spectrum as  $O$  and the phase spectrum of  $O'$ , therefore any difference between  $I_{O'}$  and  $O$  is due to the difference between the information contained in  $\Phi_O$  and  $\Phi_{O'}$ . The difference between  $O$  and  $I_{O'}$  are quantified by means of the SNR and SSIM as to evaluate the phase information retrieval. The process to generate  $I_{O'}$  is depicted in figure 2.16, where a double slit object is used as example.

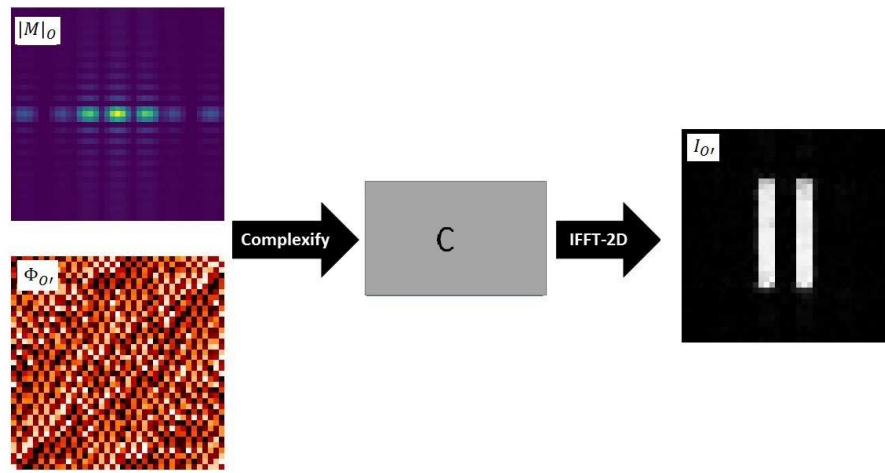


Figure 2.16: Image construction process to determine the retrieval of phase spectrum information. A double aperture object is used as example.

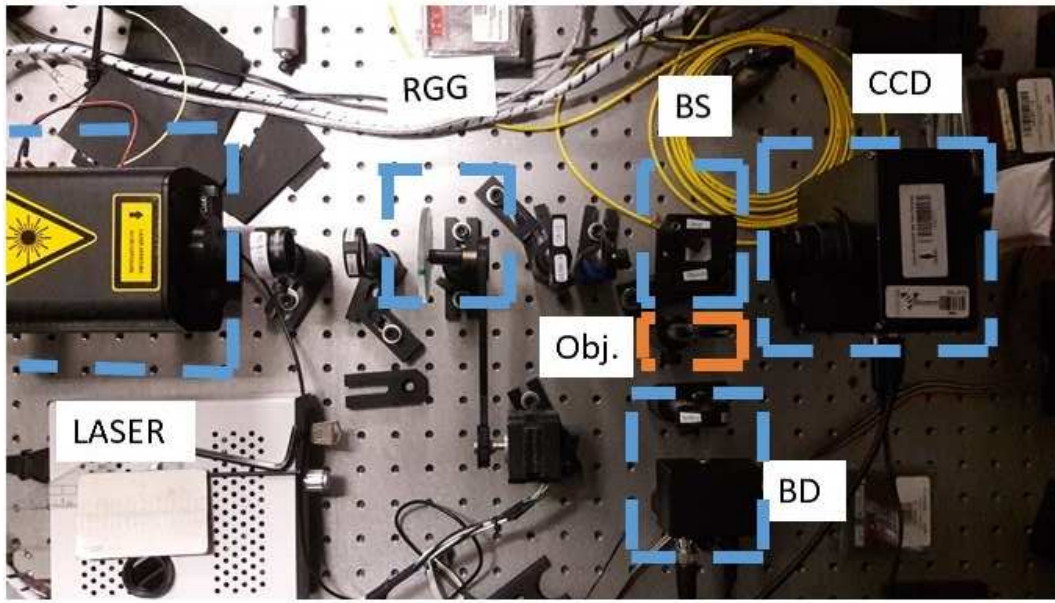
## Chapter 3

# Experimental Setup and Results Obtained

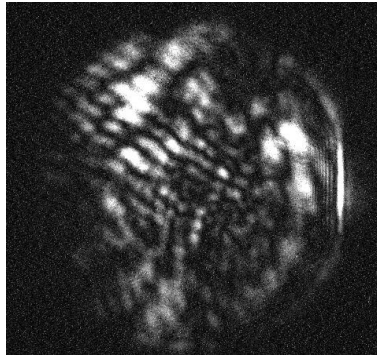
In this chapter the physical implementations of the standard pseudo-thermal ghost imaging and the proposed alternative computational ghost imaging setups will be displayed. The obtained images from the used objects will be shown, together with their corresponding analysis regarding their SNR, SSIM and phase information retrieval. The latter will be done by means of an indirect measurement, proposed in the theoretical background, and the SNR between the object's phase spectrum and the obtained images' phase spectra.

### 3.1 Standard Ghost Imaging Setup

This stage of the experiment consists in building and implementing the setup displayed in figure 2.5. The actual setup in the optical table is displayed in figure 3.1a. The laser hits on the rotating round glass (RGG), generating a speckle-field that reaches a 50/50 beam-splitter (BS). One arm the light reaches the charge coupled device camera (CCD) while in the other it reaches the object (Obj) and the bucket detector (BD). This BD is made up of a converging lens of focal length  $f = 35mm$  and a photodiode. An example of a speckle-field as captured by the CCD can be seen in figure 3.1b. The whole setup, except the computer used for data processing, was placed in a dark room in order to avoid error-inducing effects of any light present in the environment. The CCD, photodiode and DC motor were controlled manually from a computer, reason for which a considerably big amount of data could not be achieved, yet the results prove that this was not a major issue.



(a)



(b)

Figure 3.1: (a) Setup used for standard GI. (b) Sample speckle-field generated by the RGG.

To obtain the data it is necessary to generate different speckle-fields. This is done by rotating the RGG at different angles. To capture the photos of the speckle-field generated by the RGG in the reference arm, the CCD was used with an exposure time of 1 second and the resulting photo was resized from a size of  $765 \times 512$  pixels to a size of  $100 \times 100$  pixels to allow the correlation process to be computationally feasible. The object used is a circular aperture of radius  $R \approx 1\text{cm}$ , formed by a variable radius iris. The speckle-fields that reach the CCD have a round transverse section with a radius of approximately  $3\text{cm}$ .

Thanks to the fact that the size of the aperture is just slightly smaller than transverse size of the speckle-field received by the camera, the chosen resizing still allows to generate an image, though it introduces a smaller resolution compared to the original size. It must be noted that the speckle-field had a round transverse shape, so the rectangular effective area of the CCD cam, bigger than the speckle-field's area, received no light in a section of each corner. This problem was addressed by selecting a square area that fitted the entire round area minimizing the remaining dark spots.

To obtain the intensity measurements on the object arm, the intensity measured by the photodiode was acquired at the same time in which the CCD was exposed to the speckle-field. The initial values of the photodiode during each measurement were ignored, as peaks appeared due to the change of light received whenever the RGG moved and stability was achieved in the following values. Conversion of the voltage output of the photodiode to intensity was done according to its technical specifications. Each time the RGG rotated a small angular step, the procedure described above was repeated. The total amount of measurements was 398. All the collected data was saved in .txt files for the latter processing.

### 3.1.1 Experimental Results: Cross-Correlation Method

In this case  $M = 398$ ,  $N = 100$  and  $L = 100$ . There are  $1 \times 10^5$  pixel intensity value vs trial number signals which are correlated with the photodiode intensity signal via the normalized cross-correlation. The circular aperture used and the resulting image are displayed in figures 3.2a and 3.2b.

The shape of the imaged object can be clearly identified, which evidences the success of the experiment. Figures 3.2c and 3.2d display a horizontal and vertical transverse cut of the resulting image and the circular aperture, respectively. Several aspects can be noted in figure 3.2. First of all, a round, dark silhouette can be seen in figures 3.2b containing the image. This marks the round transverse shape of the speckle-fields used, and the regions beyond this silhouette should not be taken into consideration. Second of all, figures 3.2c and 3.2d display different multiple peaks in a non-symmetrical manner. The presence of these peaks is due to the fact that a relatively low amount of data was used to generate the image and therefore the expected gaussian-like curve was not obtained. The lack of symmetry is due to the fact that the speckle-fields used are not symmetrical in their light distribution.

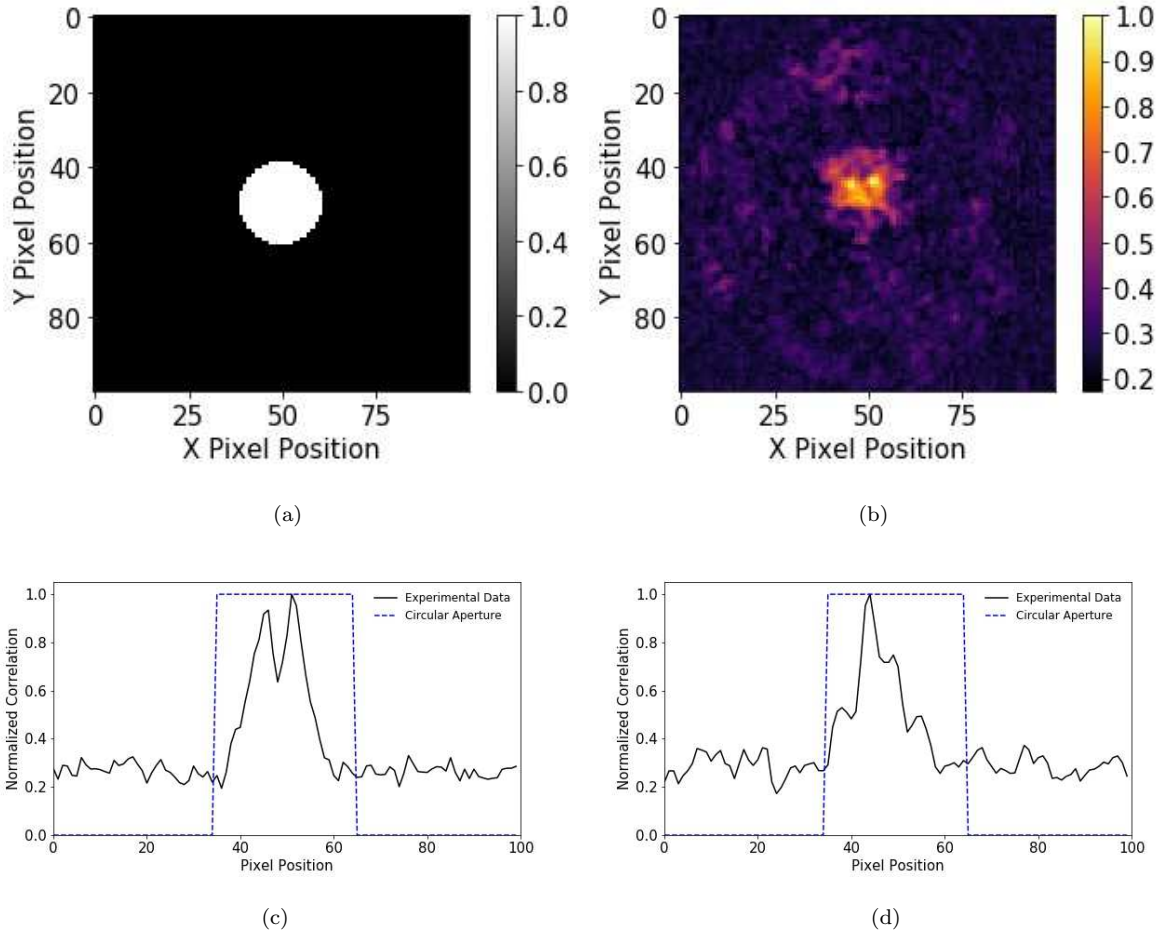


Figure 3.2: (a) Circular aperture used in the setup. (b) Obtained image via the correlation method. (c) Horizontal transverse cut of (b). (d) vertical transverse cut of (b).

### 3.1.2 Cross-Correlation Method with Simulated Object Path

To establish a superior quality boundary to the reconstructed image, i.e the best possible reconstruction from the speckle-fields used, the object path was simulated by using a Matlab R2016a in a computer. The speckle-field photos obtained by the camera were used to calculate the theoretical intensity that the photodiode should be measuring in each step of the RGG. To determine the value of this measurement, each speckle-field photo matrix is modulated by a binary matrix that simulates the aperture object. The resulting matrix is then summed over all of its terms and the result is the intensity measurement that should be detected by a photodiode. Just as in the experimental procedure, the pixel value intensity vs trial signal is cross-correlated with the simulated intensity measurements and an image is generated. This image is displayed in figure 3.3a. This algorithm took an average of 5.3 seconds to generate the image.

Once again, the image of the circular aperture can be clearly identified. Given that any measurement errors induced by the photodiode's sensibility are eliminated by simulating the object path, the resulting image resembles the circular aperture more than 3.2b. Figures 3.3b and 3.3b display the horizontal and transverse cut of the obtained image. An improved behaviour can be seen, compared to figures 3.2c and 3.2d. Again, the presence of multiple pics is attributed to the low amount of data used while the dark, round silhouette is

attributed to the size of the used speckle-fields.

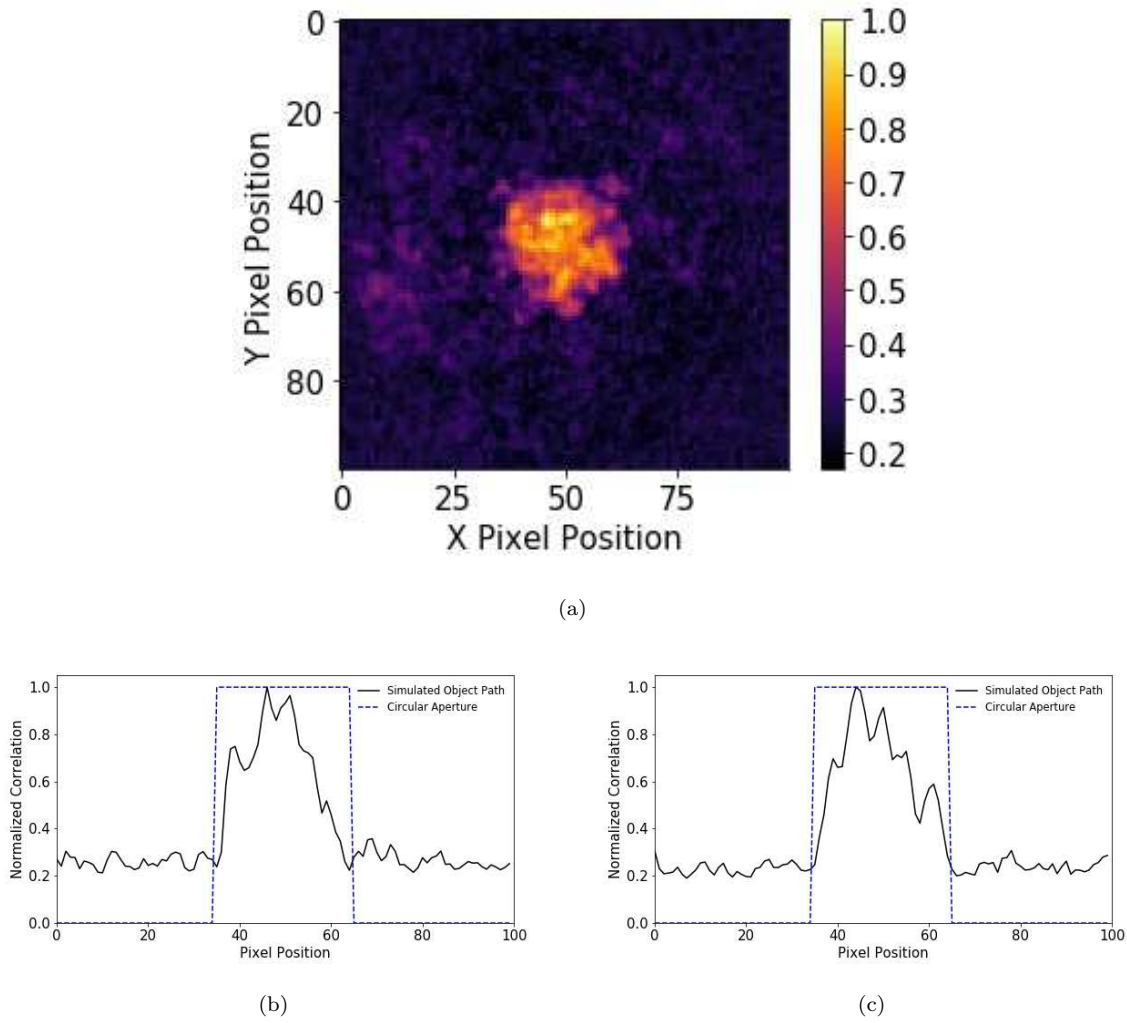


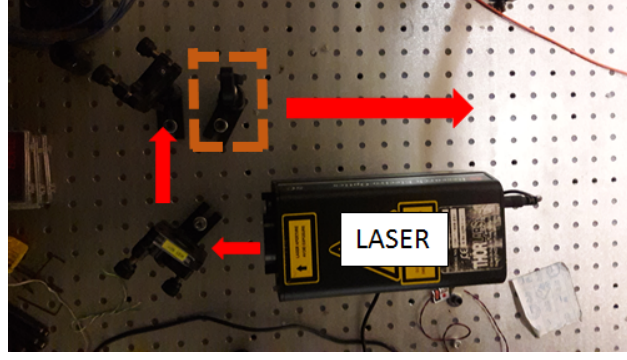
Figure 3.3: **(a)** Image obtained implementing the correlation method using a simulated object path. **(b)** Horizontal transverse cut of (a) and figure 3.2a. **(c)** Vertical transverse cut of (a) and figure 3.2a.

## 3.2 Alternative Computational Ghost Imaging

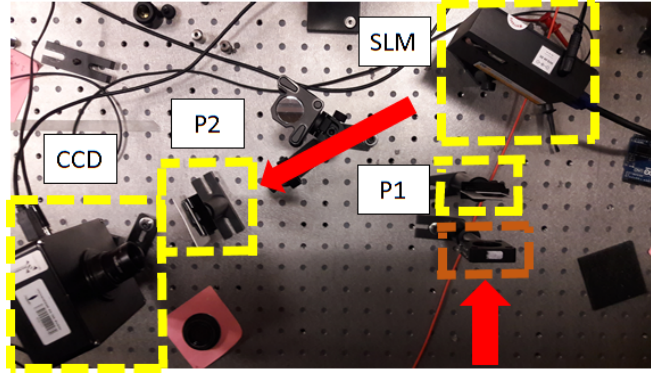
For the CGI setup displayed in figure 2.11, a Cambridge Correlators SDE1024 SLM of  $768 \times 1024$  pixels is struck by an expanded beam of a He-Ne laser (633nm). The pixel pitch is  $9\mu m$ . The reflected light reaches a lens ( $f = 100\text{mm}$ ) that performs the "SLM-Lens" imaging system, such that the pattern displayed in the SLM is the same that hits upon the CCD. All of this happens in a dark room to avoid the effects of any random light. The physical setup on the optical table is displayed in figures 3.4a and 3.4b, where the red arrows depict the path of the light beam. The lenses inside the brown dashed lines correspond to the implemented galilean beam expander with focal distances of  $-75\text{mm}$  and  $500\text{mm}$ . The speckle-fields are generated by sending pseudo-random binary matrixes made up of  $24 \times 32$  super-pixels (each super pixel is made up of  $32 \times 32$  pixels of the SLM). A copy of these captured speckle-fields are then resized to a size of  $24 \times 32$  pixels. An example of one of the matrixes sent to the SLM and the corresponding captured speckle-field are shown in figure 3.5a and 3.5b, respectively. The virtual object used in this case corresponds to a double slit aperture



of total size  $24 \times 32$ . Each slit has a width of 3 pixels and a length of 12 pixels. The slit separation is 3 pixels.

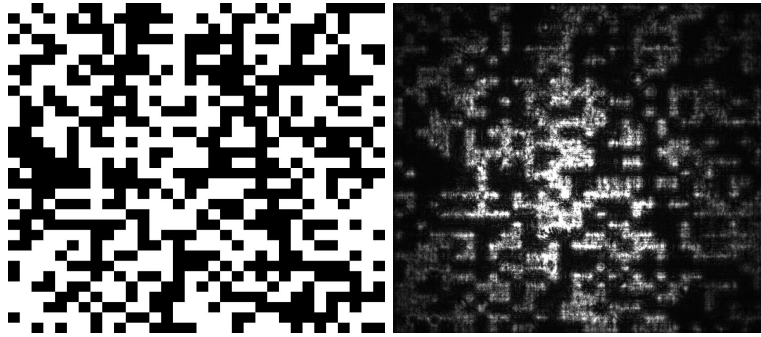


(a)



(b)

Figure 3.4: **(a)** Laser, set of mirrors and first lens of the implemented galilean beam expander. **(b)** Second lens of the beam expander, polarizers P1 and P2, SLM and CCD camera.



(a)

(b)

Figure 3.5: **(a)** One of the pseudo-random binary patterns sent to the SLM as a matrix. **(b)** Resulting speckle-field that hits on the CCD screen in a setup such as the one in figure 2.11.

### 3.2.1 Experimental Results: Cross-Correlation Method

For this experimental procedure, 1536 speckle-field patterns were captured using the CCD with an exposure time of 5 seconds. Since the CCD camera could not be automatized, the process was done manually, making the data collection process take a considerably long time. The element-wise modulation and overall addition



process to obtain the intensity measurements was done in Matlab R2016a. The virtual double slit object used is shown in figure 3.6 and the resulting image from the cross-correlation method is shown in figure 3.6b.

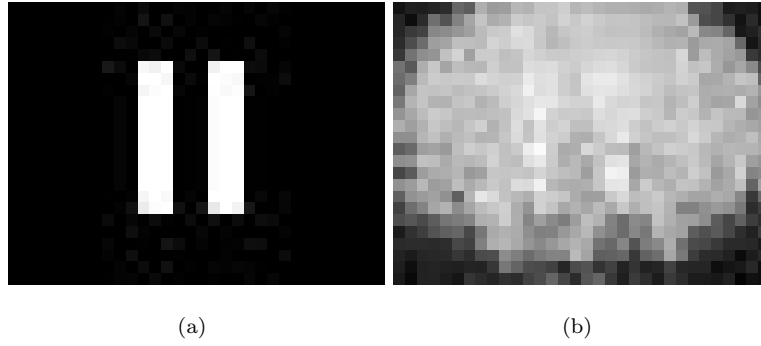


Figure 3.6: **(a)** Implemented aperture matrix of a double slit. **(b)** Obtained image using 1536 speckle-field patterns in the correlation method. The noticeable contour depicts the area that is illuminated by the most intense transverse section of the expanded laser beam.

Figure 3.6b depicts no identifiable image of the double slit object. This result is expected, since the total number of speckles is  $24 \times 32 = 768$ . This means the Nyquist limit for this image is 768, and the amount of data used is twice this value. Since the used speckle-fields may overlap or have similar regions, [17] specifies that an amount of data much bigger than the Nyquist limit must be used in order to obtain an identifiable image with high SNR.

### 3.2.2 Cross Correlation Method with Simulated Object Path

For the simulations required in this section, the whole experimental setup is modelled in Matlab R2016a. The pseudo-random patterns are created as  $24 \times 32$  matrixes, the double slit aperture is simulated as a binary matrix and the bucket detector is simulated as a sum over all the elements of the matrix resulting from the element-wise modulation between each random pattern and the aperture. Applying the cross-correlation algorithm to the pseudo-random patterns and the measured intensities allows to generate the image of the double slit aperture. However, this process assumes that the pattern sent to the SLM is the same that hits on the object. This is not always the case, since the SLM presents a dead space between pixels, along with any other possible aberrations from fabrication. These errors will modify the light pattern that reaches the object, and therefore the intensity measured by the photodiode. To simulate such condition, white gaussian noise is added to each pattern  $M_{ij}$  to obtain a modified one,  $M'_{ij}$ , that satisfies having a certain Signal-To-Noise Ratio with respect to  $M_{ij}$ , denoted  $SNR_R$ . This  $M'_{ij}$  is modulated and the measured intensities are obtained from it.

This way, the quality of the obtained image by this method depends on two factors; the amount of data obtained and the SNR that models the noise given by the SLM. When no noise is added, the SNR is considered to have a value of  $\infty$ . Figure 3.7 depicts the obtained SNRs vs amount of data obtained for different levels of noise added to the pseudo-random patterns, together with the theoretical  $\sqrt{N}$  behavior that has been already reported by [16].

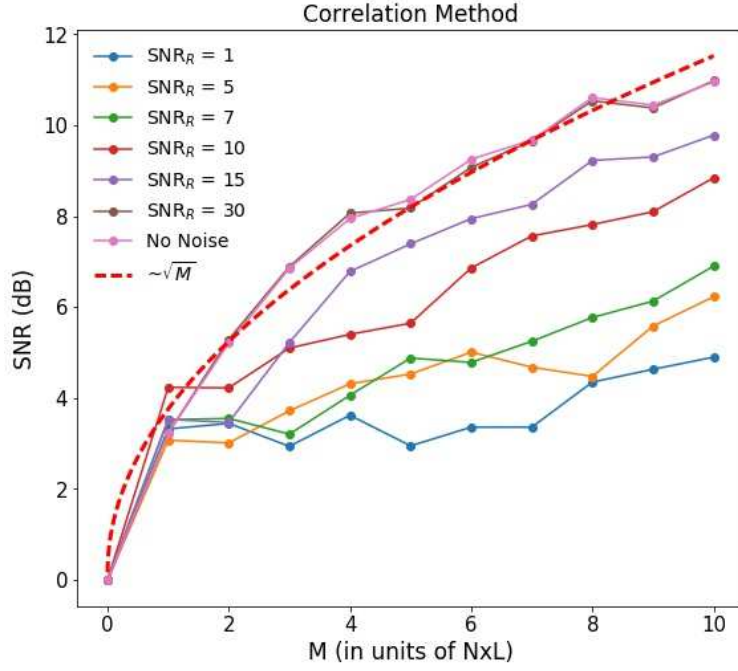


Figure 3.7: SNR vs amount of data used  $M$ , given in units of  $N \times L$ , for different  $SNR_R$ .

### 3.2.3 Experimental Results: Modulation Method

For this experimental procedure, only 768 of the total 1536 speckle-field patterns are used, since the size of the generated image is  $24 \times 32$ . This way the matrix  $S$  has a size of  $768 \times 768$ . Given the pseudo-random nature of each speckle-field pattern generated,  $S^{-1}$  can be found and therefore  $\vec{T}$  can be found, following equation 2.6. Figure 3.8a displays an example of the resized speckle-fields used and figure 3.8 shows the image obtained after finding  $S^{-1}$  and therefore finding  $\vec{T}$ .

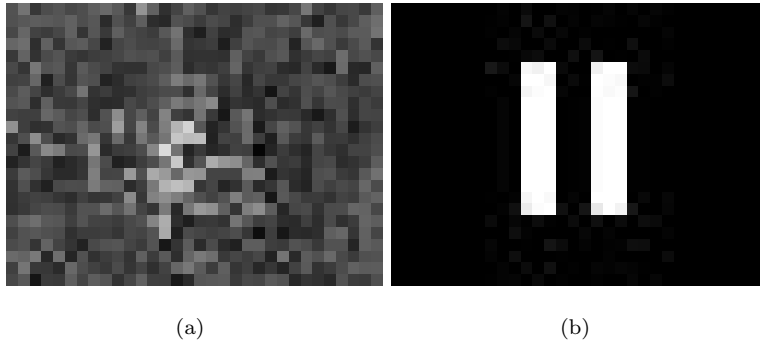


Figure 3.8: (a) The example compressed speckle-field, corresponding to the one shown in figure 3.5.(b) Obtained image using 768 speckle-field patterns in the modulation method.

Figure 3.8 shows the perfect reconstruction achieved by the implementation of the proposed alternative setup to obtain an image of the double slit virtual object. This is, avoiding any noise by taking the speckle-field patterns generated in the image plain. If this same method is applied relating the binary matrixes sent to the SLM and the measured intensities, the results would clearly never depict the desired image. In the following section the CS algorithms will be applied in the same manner, where an identifiable result will

allow to witness the advantages of the proposed setup when the SLM noise cannot be characterized precisely enough.

### 3.3 Compressive Alternative Computational Ghost Imaging

In this section, the CS algorithms of BP and TV will be applied to the modulation method to generate the image of a double slit aperture object. The obtained images will be displayed, along with the study of their SNR, SSIM and phase information retrieval success.

#### 3.3.1 Experimental Results: Modulation Method with TV algorithm

To generate an image using the TV algorithm, different amount of data, expressed as a proportion of the number of pixels  $N \times L$ , are selected from the total of 1536. This number of data  $M < N \times L$  defines the number of rows taken in the speckle matrix  $S$  and the size of the vector of intensities  $\vec{I}$ . These matrix and vector are used in the Matlab algorithm TVAL3, developed by [28], which finds the solution to equation 2.13 which satisfies the highest sparsity in the gradient basis.

TVAL3 allows to implement different kinds of solutions for the TV algorithm via the specification of different input parameters. For this work, the only setup parameter relevant is the non-negativity of the solution, since it is known that the image cannot have negative values in its pixels. All other parameter have default values which the authors suggest not to modify. Apart from this, the program takes only the incomplete speckle matrix  $S$ , the intensities vector as column  $\vec{I}$  and the resolution of the resulting image. As to increase the resolution of the generated image as far as the amount of data taken allows, a copy of the original captured speckle-fields are compressed to a size of  $39 \times 39$  pixels, the virtual object having the same resolution. To check the computational efficiency of the program, which works on iterations until finding the desired solution, figure 3.9 depicts the number of iterations vs  $M$  (the proportion of data used), in units of  $N \times L = 1521$ , together with the mean number of iterations, around which the values stay stable, and the predetermined maximum number of iterations defined in the program.

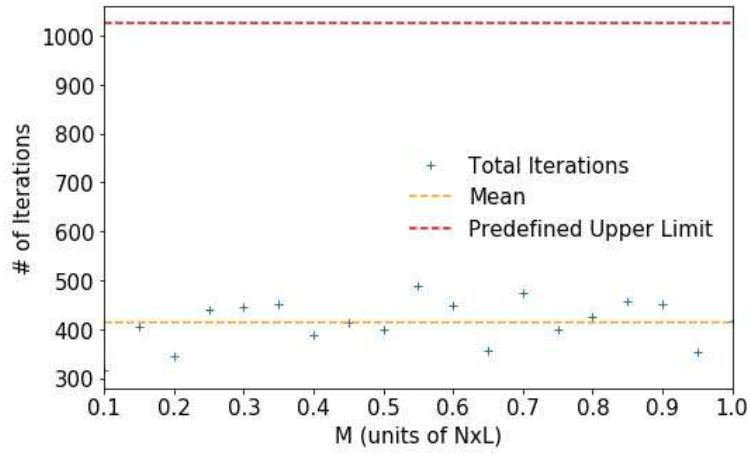


Figure 3.9: Number of iterations vs proportion of the total data used in the TV algorithm. As the iterations realized stay well beneath the upper limit predetermined in the program, it can be assumed that convergence is easily obtained and the method itself works well with the provided data in terms of functionality.

Running TVAL3 with the matrix  $S$  and the vector  $\vec{I}$  for different values of  $M$  allows to obtain an image for each value of  $M$ . The  $39 \times 39$  pixels double slit virtual object and the images for  $M = 0.2$ ,  $M = 0.3$ ,  $M = 0.8$  and  $M = 0.9$  are shown in figure 3.10a, 3.10b, 3.10c, 3.10d and 3.10e, respectively.

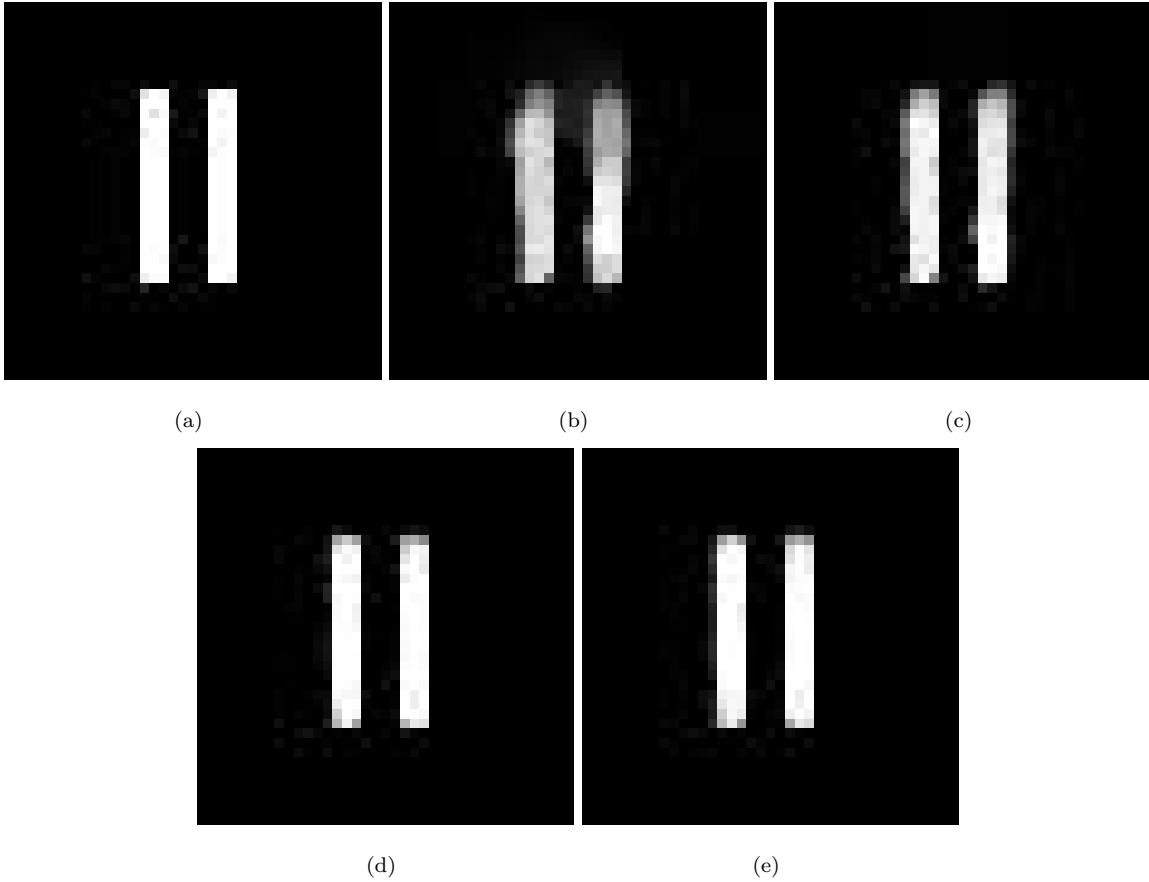


Figure 3.10: (a) Double slit virtual object. Images are depicted for values of  $M$  equal to (b) 0.2, (c) 0.3, (d) 0.8 and (e) 0.9.

### 3.3.2 Application of Image Comparison Methods

To analyze the obtained images it is first important to check that the SNR obtained continues to behave as  $\sqrt{M}$ , given how it is expected that the image resembles the original aperture as more data are taken into account. This can be seen in figure 3.11, together with the approximately expected behaviour. Certain points present a relatively big deviation from the proposed  $\sqrt{M}$  fit, yet this is explained as different noises can produce the same SNR with respect to a reference image, so it is possible that while the white sections become more like the expected image, the black areas acquire a distributed noise, presenting an ever higher noise than one with poorly defined white areas but with near to zero values in the black areas. The  $\sqrt{M}$  behaviour is thus obeyed in a global growth, deviations being accepted in certain points due to the reason previously shown.

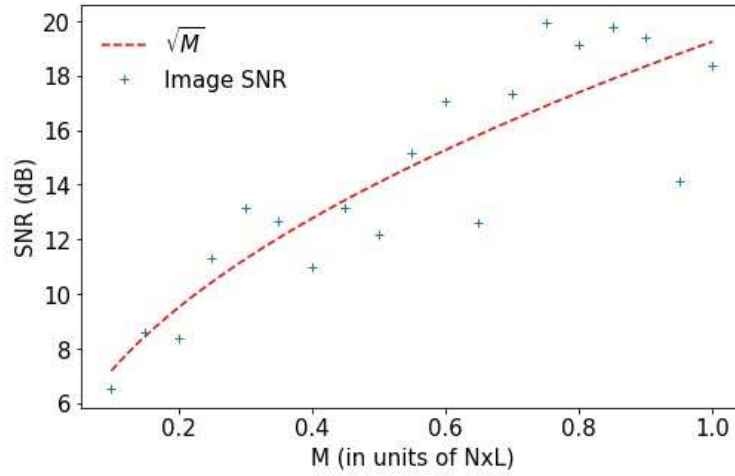


Figure 3.11: Calculated SNR vs M, amount of data used in units of  $N \times L$ .

As to justify said error tolerance in the SNR characterization, figures 3.10d and 3.10e can be observed, where both present a near optimal reconstruction of the original object, yet their corresponding SNR values display a notable difference. With this in mind, it can be asserted that the SNR is not a definite criterion to evaluate the quality of the obtained images.

The SSIM index can be calculated for each obtained image with reference to the used aperture. These results are depicted in figure 3.12, where the values get progressively close to 1 revealing the high performance of the TV algorithm in recovering the double slit aperture structure in the obtained image. This growth reaches a value very close to 1 rather quickly and becomes smoother towards it, given the fact that the double slit structure is clearly identified even from very low values of data proportions up to the final value of 0.95. The increasing behaviour of the SSIM values as the amount of data used increases depicts the expected image improvement, yet it presents similar variations as the SNR data, indicating that neither the SSIM or SNR are definite criteria for image recovery.

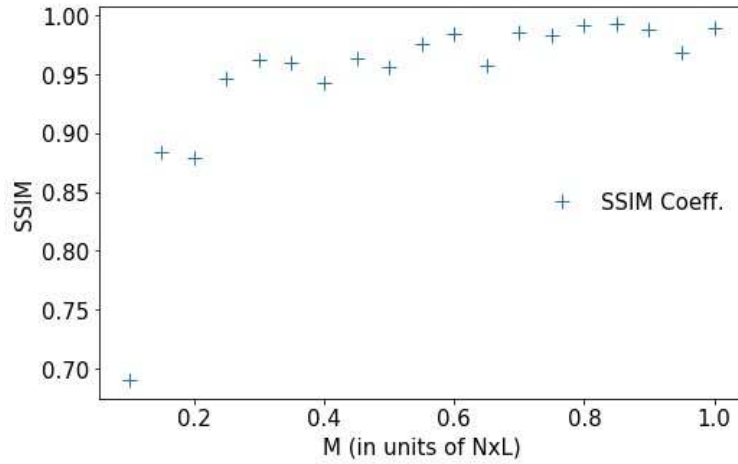


Figure 3.12: Calculated SSIM coefficients for the generated images using the double slit aperture as reference.

### 3.3.3 Phase Analysis

By obtaining a *Phase Image* for each value of  $M$  it is possible to determine a SNR value with respect to the used aperture, which quantifies the difference of information contained in the recovered phase spectrum with respect to the double aperture object as the magnitude is left unchanged. Figure 3.13 depicts the obtained results (Blue dots) together with the previously calculated SNR values for the images obtained via the TV algorithm, *TV Images*, (Red triangles) as to evidence the similar  $\sqrt{M}$  behaviour. It can be noted that, for a fixed value of  $M$ , the SNR values for the phase images are mostly higher than the values corresponding to the TV images. This occurs due to the fact that the phase images share the magnitude spectrum with the object, while the TV images have different magnitude and phase spectra. The  $\sqrt{M}$ -like growth displayed by the data (Blue dashed for Phase images and red dashed for TV images) shows the increase of information recovery as the amount of data used increases.

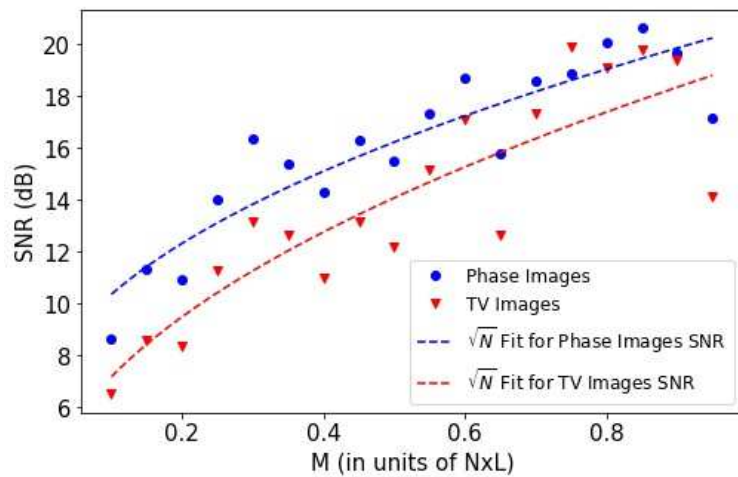


Figure 3.13: SNR vs proportion of data for the images generated using the TV algorithm (TV Images) and the images build from their phase spectrum (Phase Images).

In a similar way as before, the SSIM index value is calculated for each value of  $M$ . The results are shown

in figure 3.14, together with the ones corresponding to the TV images. In this case, the sets of data display different behaviours. While the SSIM values corresponding to the TV images depict the approximately close to 1 behaviour explained previously, the results corresponding to the Phase images depict a linear-like growth starting in low values and reaching a maximum value of approximately 0.8. This is due to the fact that, while the double slit structure is identifiable in all the phase images, there exists noise outside the double slits, in what may be called the *black background*, decreasing the value of the calculated SSIM index as there are now secondary structures given by non-zero values where there should not be any. The approximately linear growth of the SSIM index as the value of  $M$  increases evidences the decrease of said noise in the black background, acquiring higher similarity to the double slit object. Just as before, this is only a partial criterion to assess the quality of the obtained images.

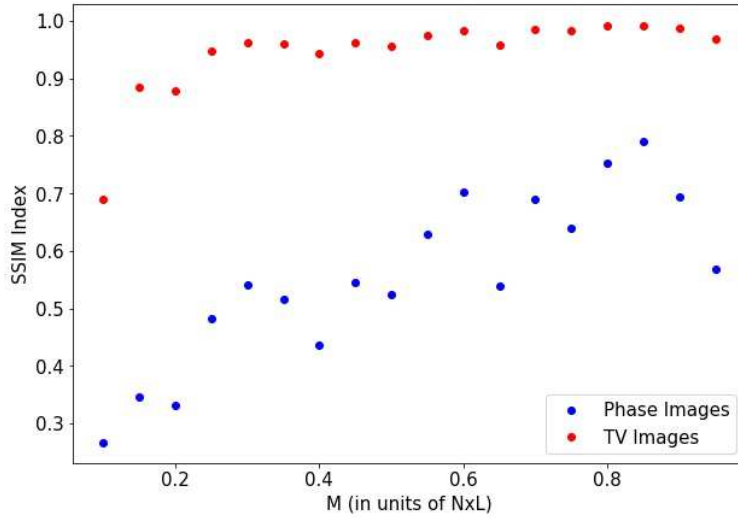


Figure 3.14: SSIM index vs data proportion for both TV images and Phase images.

Bearing the previous analysis in mind, the SNR results allow to see that, starting from remarkably low values of data proportion, the phase spectrum information from the imaged object is conserved enough to assure pixel-wise similarity between the object and its image. As for the calculated SSIM index values, the obtained behaviour shows how noise starts to appear when the total phase spectrum is not conserved. As the data proportion increases, this noise reduces and the SSIM value increases. Remarkably enough, although the SSIM values evidence these differences, they are not notable enough for the double slit aperture structure to be lost, rendering the image identifiable enough to human vision as corresponding to the imaged object.

It must be noted that even though the phase spectrum information is being well retrieved, this is done in a rather interesting way. Figure 3.15a depicts the obtained values of SNR between each  $\phi_M$  and the phase spectrum of the double slit object. The negative values indicate poor similarity in a pixel-wise comparison between the two phase spectra. This idea is stressed by visually noticing that Fig. Figure 3.15b and figure 3.15c that depict  $\phi_M$  with  $M = 0.95$  and the original object phase spectrum, respectively, do not resemble one another. Even with this lack of similarity in the phase spectrum, images that well resemble the object have been obtained (see figures 3.10(b-d)). Therefore, since the phase spectrum information contributes

to the quality of an image, one may conclude that this phase information cannot be stored in a pixel-wise similarity. This leads to consider that the phase information is coded in a different aspect, such as general structure of the spectrum or local sections that satisfy unknown conditions.

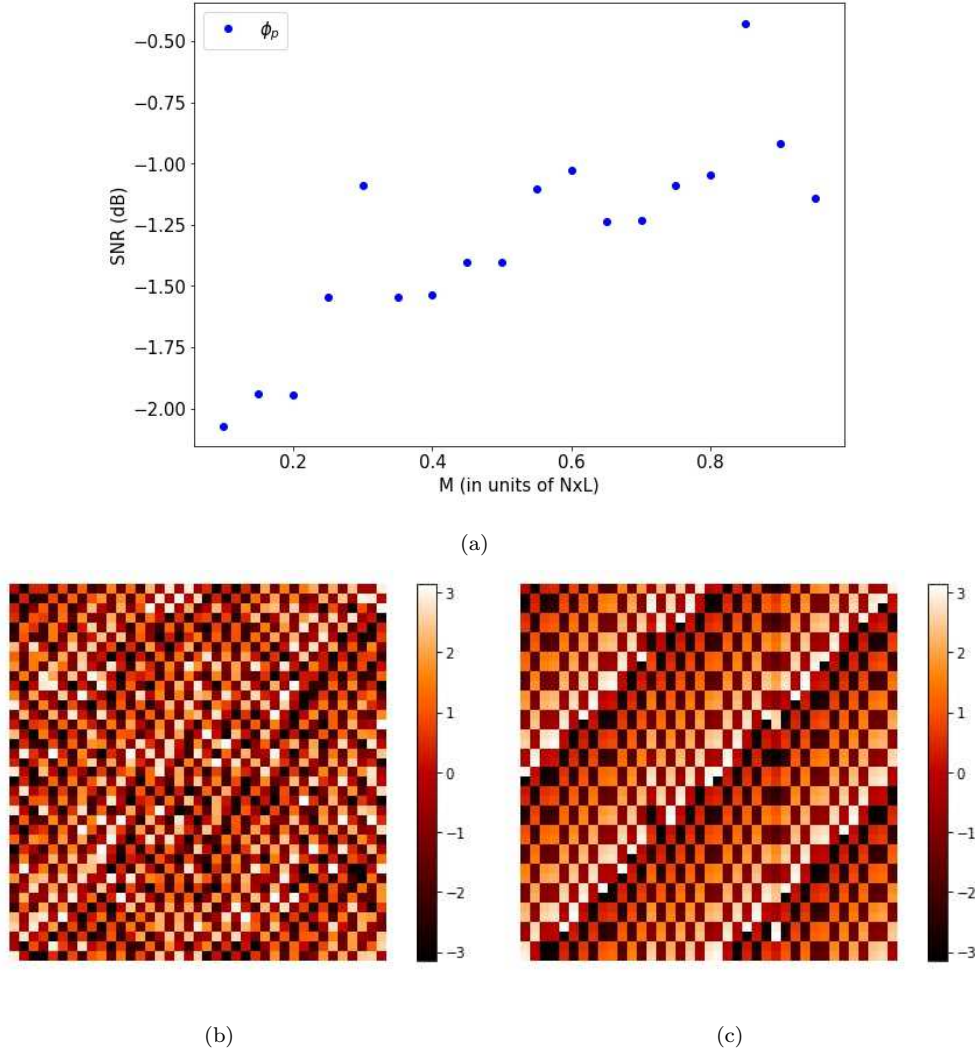


Figure 3.15: (a) SNR vs M for the different phase spectra obtained. (b) Phase spectrum for  $M = 0.95$ . (c) Original phase spectrum for the double slit object. Notice no visual similarity between (b) and (c).

### 3.3.4 Experimental Results: Modulation Method with BP algorithm

To generate an image using the BP algorithm the same amounts of values M are used as with the TV algorithm. The matrix  $S$  and intensity vector  $\vec{I}$  for each value of M are used in the Matlab program Active Set Pursuit (ASP), developed by [29], which finds the solution to equation 2.13 by applying the BP algorithm.

ASP allows to freely choose a basis  $\Phi$  for the BP algorithm. Since the DCT basis is chosen for this work, the sampling matrix given to the ASP program corresponds to  $S\Phi^{-1}$ , where  $\Phi^{-1}$  is the inverse DCT matrix, and the solution vector corresponds to  $\vec{I}$ . This solves equation 2.9 with  $A = S$  and  $\vec{y} = \vec{I}$ . The solution  $\vec{s}$  given by the ASP allows to find  $\vec{T}$ , via applying the inverse DCT matrix to it. Apart from these two parameters, others can be modified, yet the authors recommend using the predetermined configuration. Just as in the



TV implementation, the speckle-fields have a size of  $39 \times 39$  pixels and so does the double slit virtual object.

Running ASP for the same values as in the TVAL3 implementation allows to generate an image for each value of  $M$ . Figures 3.16(a-d) depict the obtained images for different values of  $M$ , allowing to see the high quality reconstruction achieved by the implementation of ASP.

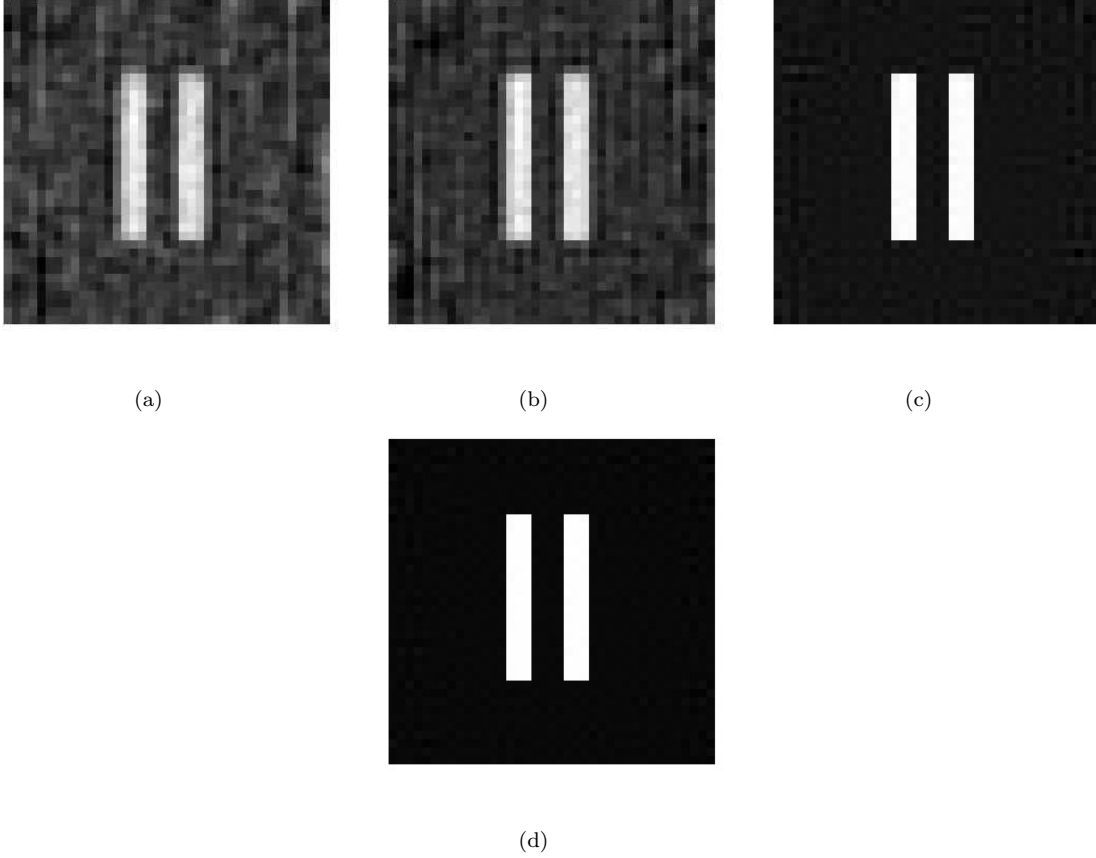


Figure 3.16: Obtained images for values of  $M$  equal to (a) 0.2, (b) 0.3, (c) 0.8 and (d) 0.9.

### 3.3.5 Application of Image Comparison Methods

To analyze the obtained images, the SNR is calculated for each value of  $M$  to check the general behaviour it follows. The obtained results for this calculation are shown in figure 3.17. A notorious aspect is that the behaviour is no longer that of  $\sqrt{M}$ . The rate of growth seems to be approximately constant up to  $M = 0.5$  and then it becomes notably marked towards high values of  $M$ , reflecting the high quality of the obtained images. Similarly, the achieved values of SNR are much bigger than in the TV implementation case. This can be attributed to the fact that the double slit image is clearly and neatly identified from very low values of  $M$  and what seems to be gradually disappearing is the noise in the black background.

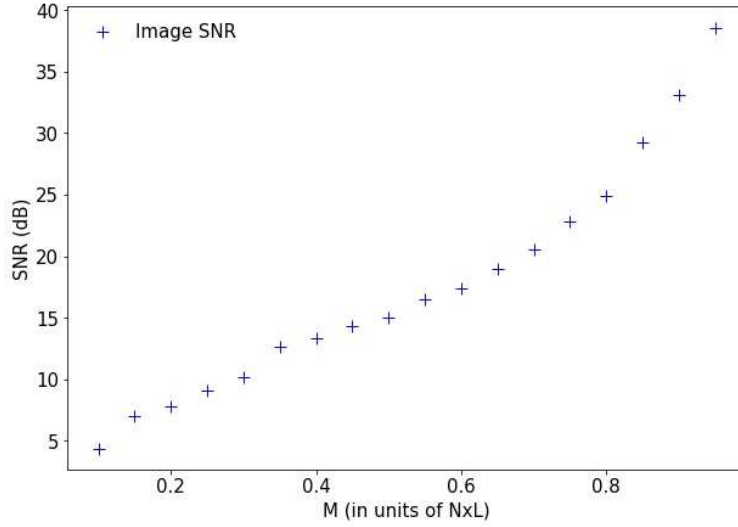


Figure 3.17: SNR vs  $M$  for the images obtained via de BP algorithm.

It is now possible to calculate the SSIM values for the different values of  $M$ , just as in the TV implementation case. The results of this calculation are shown in figure 3.18. In this case, contrary to the TV implementation, the SSIM coefficient displays an approximately lineal behaviour as  $M$  increases. As it happened before in this work, this could be due to the fact that the double slit structure is identifiable from very small values of  $M$ . As  $M$  increases, the most notable changes occur in the diminishing of the background noise that is perceived as a secondary structure, i.e non-zero values where there should be only zero values. Remarkably enough in  $M = 0.6$ , where the SNR seems to acquire a fast growing behaviour, the SSIM appears to have an inflection point, and the growth starts to diminish. From this point on, the double slit aperture becomes the most notable in comparison to any background noise, which explains why the SSIM tends to decrease its growth while the SNR starts growing more rapidly.

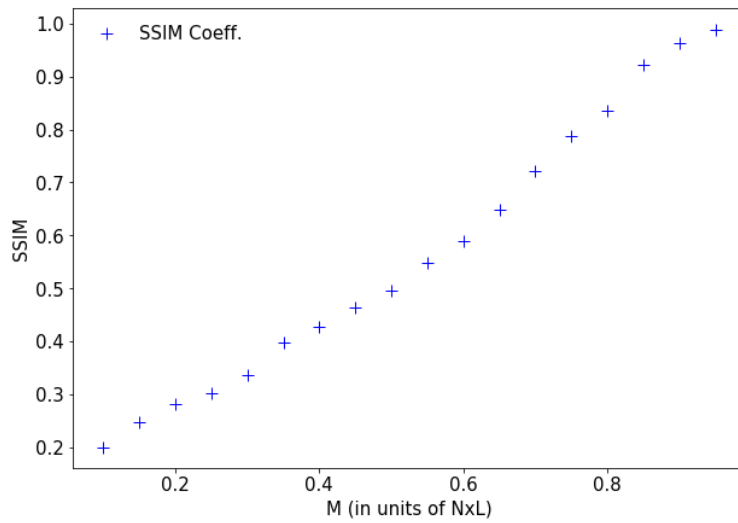


Figure 3.18: SSIM vs  $M$  for the images obtained via de BP algorithm.

### 3.3.6 Phase Analysis

To study the phase information retrieval achieved by implementing the BP algorithm, a *Phase Image* is generated for every value of  $M$ . The SNR results for these images allow to quantify the success of this retrieval. Figure 3.19 displays the obtained values of SNR vs  $M$  for each Phase Image, together with the calculated values for the original images obtained via the BP algorithm; i.e BP Images. Just as in the TV case, the phase images present a higher SNR than the original ones, thanks to the fact that they share the same magnitude spectrum with the virtual object. Apart from this, both sets of data display a similar behaviour in their growth.

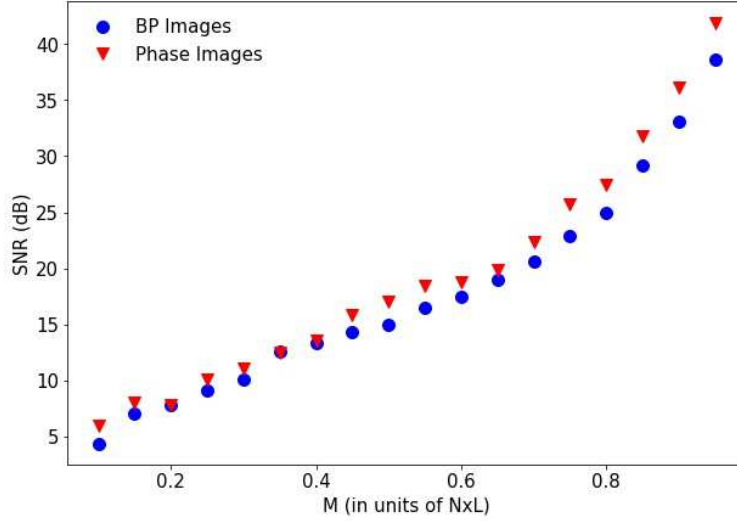


Figure 3.19: SNR vs  $M$  for the images obtained via de BP algorithm and their corresponding Phase Images.

The SSIM can also be calculated for these phase images. The results can be seen in figure 3.20, together with the values corresponding to the BP images. As in the TV case, the phase images present a lower value of SSIM for the different values of  $M$ . In this case, however, both sets of data converge to approximately the same value for high  $M$ . For lower values, on the other side, the phase images present an slowly increasing SSIM behaviour.

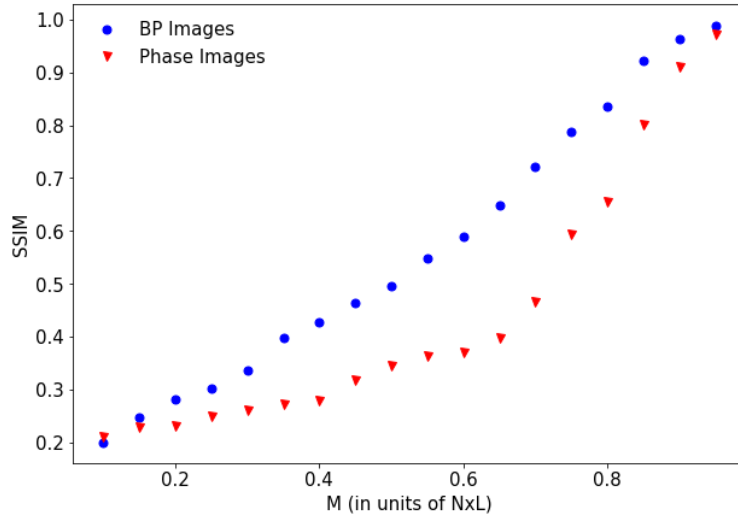
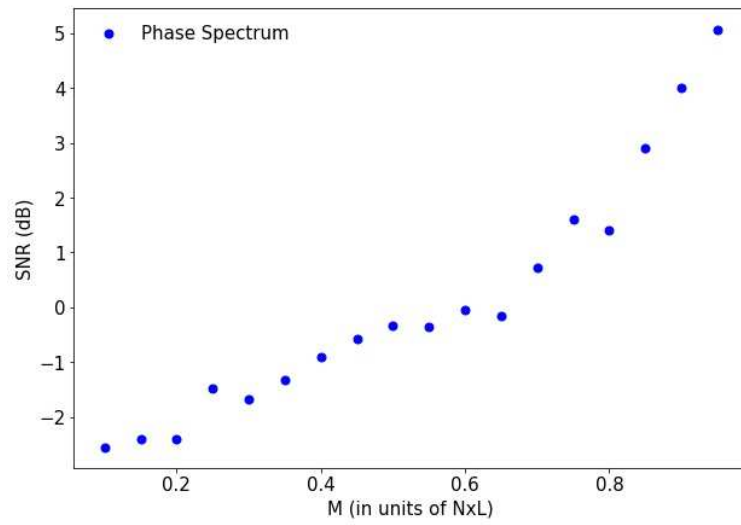
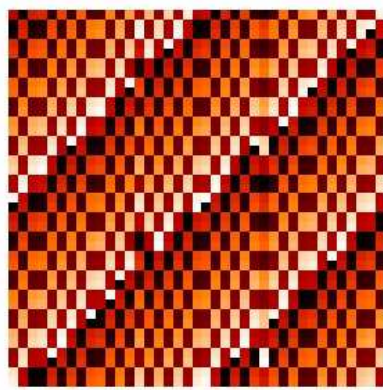


Figure 3.20: SSIM vs  $M$  for the images obtained via de BP algorithm and their corresponding Phase Images.

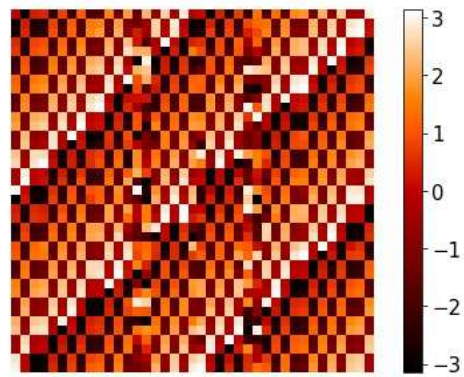
As it can be seen, the SNR and SSIM values display a more telling behaviour; unlike in the TV case, where major deviations from the global behaviour could be seen, in this case both SNR and SSIM tend to present local and global growth. Together with the obtained images, displayed in figures 3.16(a-d), these criteria show how the BP algorithm, specifically as implemented in the ASP program, allows to obtain a better image reconstruction and phase information retrieval. This superior performance is evidenced not only in the indirect measurement of phase information retrieval, but also in a direct comparison between the phase spectrum of the used object and the phase spectrum of the image obtained for every value of  $M$ . This can be seen in figure 3.21a, where SNR values greater than zero can be found for values of  $M$  greater than 0.65. An example can be seen in figure 3.21b and 3.21c, where the phase spectrum of the object and the phase spectrum of  $M = 0.95$  are displayed, respectively. Remarkable visual similarity can be observed between these two, accounting for the notorious success in image reconstruction the ASP program has presented. Remarkably enough, a soft, linear like growth can be seen in figure 3.21a for values of  $M$  from 0.1 to 0.6 and in this point, similarly to how changes of behaviour occurred in previous figures (3.19, 3.20), the rate of growth increases to a steep linear like growth.



(a)



(b)



(c)

Figure 3.21: (a) SNR vs  $M$  for the phase spectrum of each image obtained corresponding to a value of  $M$ . (b) Phase spectrum of the object. (c) Phase spectrum corresponding to  $M = 0.95$ .



## Chapter 4

# Conclusions and further discussion

### 4.0.1 Conclusions

In this monograph, several achievements were accomplished. From a theoretical aspect, an alternative ghost imaging setup and its computational version were proposed and their functionality was demonstrated. Additionally, a method that allowed to directly test the effectiveness of compressive sensing methods in ghost imaging for phase retrieval was proposed, accomplishing the main objective of the present monograph.

From an experimental aspect, a standard ghost imaging setup, the alternative computational ghost imaging and its compressive alternative were implemented on the optical table. For all the implementations it is possible to use real physical objects that are placed on the optical table or virtual objects that are modeled on a computer. For the case of standard ghost imaging we used a real object. On the other hand, for all the other implementations a virtual object was used. The implementation of the alternative setup was achieved with the available elements in the laboratory and proved to be as easy to build as the original computational ghost imaging. The data taken from the different setups was analyzed using the cross-correlation and the modulation methods. The compressive sensing algorithms used in this work were the total variation (TV) and the basis pursuit (BP). Each of the implemented setups, except the alternative computational ghost imaging analyzed via correlation method, allowed to obtain images where the imaged object could be clearly identified. The exception was an expected situation since the amount of data taken was not enough for the correlation method to work well. The acquisition of more data in future experiments should bear a high quality image.

We applied two image comparison methods to determine the quality of the image with respect to the object. Namely, the signal-to-noise-ratio (SNR) and the structural similarity index measure (SSIM). For the alternative computational GI setup, with compressive sensing using the TV algorithms, in the modulation method, the SNR showed a growth with  $\sqrt{M}$  where  $M$  is the amount of data taken. This behavior agrees with what has been reported in the literature for computational ghost imaging [16]. Demonstration of this

are the images shown in Fig 3.10(a-e) where the high quality of the image occurs even for amounts of collected data that were low compared with the Nyquist limit. This fact confirms the applicability of TV algorithms for GI setups.

The method we proposed to directly test the effectiveness of compressive sensing methods in ghost imaging for phase retrieval is based on the generation of a *phase image* that is constructed merging the magnitude spectrum of the object and the phase spectrum of the obtained image. The SNR of the experimentally obtained phase images also displayed a growth with  $\sqrt{M}$  for the alternative computational GI setup, with compressive sensing using the TV algorithm. This fact evidenced a notable phase retrieval in terms of image quality in comparison to the object. The SSIM of these phase images shows a global growth with  $M$  indicating that the quality of the image and of the phase retrieval improves with  $M$ . On the other hand, both SNR and SSIM show global improvement with  $M$  but they exhibit local deviations. This local deviations can be explain taking into account that neither SNR nor SSIM are definite criteria for image quality assessment. Nevertheless, the overall success of the TV algorithm to retrieve the phase information was clearly observed thanks to the global growth of both criteria.

For the obtained images using the BP algorithm a high quality was also obtained starting from very low values of  $M$ , specially regarding an identifiable double slit structure amongst a noisy background. The SNR and SSIM results of both obtained images and phase images did not display a  $\sqrt{M}$  behavior, and the nature of this growth is yet to be studied. Notable behaviors of inflection points and increased growth when  $M = 0.6$  could be observed and could shed some light on the nature of this behavior. However, the global growth display the increase of quality in the image generation and phase retrieval as the values of  $M$  increase. No local deviations from the global growth behavior are present for this algorithm, which displays the higher performance that BP presented over the TV for the parameters used in this experiment.

Concerning the obtained results of phase retrieval, both BP and TV algorithms successfully achieve it. However, the nature of this retrieval remains highly unknown. For the case of the TV algorithm, it was seen that the phase retrieval is not done in pixel-wise similarity, as the phase spectra of each for each  $M$  greatly differ from the object's spectrum yet the phase images highly resemble it. The method of retrieval to achieve this remains to be found. For the case of the BP algorithm, a higher pixel-wise similarity between the phase spectrum for each  $M$  and the object's is seen, attributed to the better performance this algorithm presented. This leads to conclude that phase information retrieval can be done also in pixel-wise similarity, yet it is not the only way in which this is done. This successful phase retrieval in compressive sensing algorithms via unknown methods is the main result of this work. The success of the phase retrieval was clearly evidenced and studied for different compression rates, yet the nature of the retrieval itself is left as an open question.



### 4.0.2 Further Discussions

From an experimental point of view, a characterization of the used spatial light modulator (SLM) could be done in order to be able to model the pixel-wise similarity and SNR that exists between the matrix sent to the SLM and the resulting speckle-field it generates when used as amplitude modulator. This could directly complement the alternative computational ghost imaging setup proposed in this work, where it is necessary to correlate the used matrixes with the resulting measured intensities without directly observing the resulting speckle-fields.

A more detailed and insightful theoretical and experimental study of the phase retrieval achieved via the TV and BP algorithms could also be performed. Even though this work already demonstrates a successful phase retrieval, the means by which this is achieved remains unknown. A study of the mathematics behind the algorithms and the nature of the binary objects themselves might shed light on this matter. A good starting point might be considering how the phase directly contributes to the information contained by every complex number that represents the discrete Fourier transform (DFT) of an image and how this contribution is not damaged by pixel-wise differences, as shown in this work.



# Chapter 5

## Appendices

### 5.1 SLM as Amplitude Modulator

To demonstrate the functionality of the SLM as amplitude modulator the following procedure is presented, based on the setup displayed on figure 2.11. The initial expanded electric field coming from the laser can be expressed in its Jones vector:

$$E(x, y) = \begin{pmatrix} E_H(x, y) \\ E_V(x, y) \end{pmatrix}. \quad (5.1)$$

After passing through the polarizer P1, with transmittance  $T_1$  and horizontal orientation, the resulting field is obtained as:

$$E_1(x, y) = T_1 \begin{pmatrix} E_H(x, y) \\ 0 \end{pmatrix}. \quad (5.2)$$

Upon hitting on the SLM, the reflected field is obtained by multiplying  $E_1$  by the transmission function of the SLM  $T(x, y)$ , defined as:

$$T(x, y) = R_{SLM} \times e^{i\phi(x, y)} R(\theta(x, y)) \quad (5.3)$$

where  $\phi$  is the phase shift introduced to the wave depending on the gray level programmed on each pixel of the SLM,  $R(\theta)$  is a rotation of  $\theta$  degrees whose value also depends on the gray level value of each pixel of the SLM and  $R_{SLM}$  is the SLM's reflectance. The reflected field is calculated as:

$$E_2(x, y) = T(x, y)E(x, y) = T_1 R_{SLM} \begin{pmatrix} \cos(\theta(x, y))E_H(x, y) \\ \sin(\theta(x, y))E_H(x, y) \end{pmatrix} e^{i\phi(x, y)} \quad (5.4)$$

This field then passes through polarizer P2 in vertical polarisation, with transmittance  $T_2$ , yielding the following transmitted field:

$$E_3(x, y) = T_1 R_{SLM} T_2 \begin{pmatrix} 0 \\ \sin(\theta(x, y))E_H(x, y) \end{pmatrix} e^{i\phi(x, y)} \quad (5.5)$$

It can now be seen that the intensity  $I(x, y) \propto |E_3(x, y)|^2$  that hits on the object has a value that depends solely on  $\theta(x, y)$ , which depends on the programmed value on each pixel of the SLM.



# Bibliography

- [1] H. Freeman. *Discrete-Time Systems: An Introduction to the Theory*. Published by John Wiley & Sons, Inc. (1964).
- [2] E. J. Candès, J. Romberg & T. Tao. *Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information*. IEE Transactions on Information Theory, Vol. **52**, Issue 2, pp. 489-509, February (2006).
- [3] D. L. Donoho. *Compressed Sensing*. IEEE Transactions on Information Theory. Vol. **52**, No. 4, (2006).
- [4] Marim, M.; Angelini, E.; Olivo-Marin, J. C.; Atlan, M. (2011). "Off-axis compressed holographic microscopy in low-light conditions". Optics Letters. **36** (1): 79–81. (2011).
- [5] A. Stevens, L. Kovarik, P. Abellan, X. Yuan, L. Carin, N. D. Browning. "Applying compressive sensing to TEM video: a substantial frame rate increase on any camera". Advanced Structural and Chemical Imaging. **1** (1), (2015).
- [6] A. Stern. *Optical Compressive Imaging*. CRC Press. Taylor & Farncis Group. (2017).
- [7] J. Zhang, S. Liu & D. Zhao. *Improved Total Variation based Image Compressive Sensing Recovery by Nonlocal Regularization* 2013 IEEE International Symposium on Circuits and Systems (ISCAS).
- [8] J. Zhang, R. Xiong, S. Ma & D. Zhao. *High-quality image restoration from partial random samples in spatial domain* 2011 IEEE Visual Communications and Image Procesing (VCIP).
- [9] A. V. Oppenheim & J. S. Lim. *The Importance of Phase in Signals*. Proceedings of the IEEE, vol. **69**, Issue 5, May (1981).
- [10] M. H. Hayes, J. S. Lim & A. V. Oppenheim. *Signal Reconstruction from Phase or Magnitude*. IEEE Transactions on Acoustics, Speech and Signal Processing. Vol. ASSP-**28**, No. 6, December (1980).
- [11] Y. Shih. *Quantum Imaging*. IEEE Journal Of Selected Topics In Quantum Electronics, Vol. **13**, No. 4, July/August (2007).
- [12] G. Scarcelli, V. Berardi & Y. Shih. *Can Two-Photon Correlation of Chaotic Light Be Considered as Correlation of Intensity Fluctuations?*. Phys. Rev. Lett. Vol. **96**, 063602,(2006).
- [13] B. I. Erkmen & J. H. Shapiro. *Ghost Imaging: from quantum to classical to computational*. Advances in Optics and Photonics, Vol. **2**, (2010).

- [14] J. H. Shapiro. *Computational Ghost Imaging*. Phys. Rev. A. Vol. **78**, 061802, (2008).
  - [15] H. Xie, Z. Cao, J. Zhou & L. Xu. *Ghost Imaging of Binary-valued Objects using a CCD and an equivalent Photodiode*. IEEE International Conference on Imaging Systems and Techniques, Pages 1-5, (2015).
  - [16] Y. Bromberg, O. Katz & Y. Silberberg. *Ghost Imaging with a single detector*. Phys. Rev. A. **79**, 053840, (2009).
  - [17] Y. Bromberg, O. Katz & Y. Silberberg. *Compressive Ghost Imaging*. App. Phys. Lett. Vol, **95**, 131110, (2009).
  - [18] Gong, W. et al. *Three-dimensional ghost imaging lidar via sparsity constraint*. Sci. Rep. **6**, 26133; doi: 10.1038/srep26133 (2016).
  - [19] C. Deng, W. Gong & S. Han. *Pulse-compression ghost imaging lidar via coherent detection*. Optics Express, Vol. **24**, Issue 23. (2016).
  - [20] Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao & M. Segev. *Phase retrieval with Application to Optical Imaging*. IEEE Signal Processing Magazine. Vol. **87**, (2015).
  - [21] J. Wang, R. Yu, Y. Xin, Y. Shao, Y. Chen & Q. Zhao. *Ghost Imaging with Different Sizes of Thermal Light*. Journal of the Optical Society of Korea, Vol. **20**, No. 1, (2016).
  - [22] R. G. Baraniuk. *Compressive Sensing*. IEEE Signal Processing Magazine, Vol. **118**, (2007).
  - [23] A. Torralba & A. Oliva. *Statistics of Natural Image Categories*. Inst. Phys. Publ. Network: Comput. Neural Syst. 14 (2003).
  - [24] A. P. Sánchez. *Colección Nueva York*. (2016).
  - [25] T. Tao. *Ostrowski lecture: The uniform uncertainty principle and compressed sensing*. What's New Research Blog. (2007).
  - [26] Z. Wang, E. P. Simoncelli & A. C. Bovik. *Multi-scale structural similarity for image quality assessment* Proc. IEEE Asilomar Conference on Signals, Systems, and Computers, 1398-1402(2003).
  - [27] D. M. Rouse & S. S. Hemami. *Understanding and symplifying the structural similarity metric*. 15th IEEE International Conference on Image Processing, (2008).
  - [28] C. Li, W. Yin & Y. Zhang. *TV Minimization by Augmented Lagrangian and Alternating Direction Algorithm*. Total Variation Algorithm available online. Version 1 (2010).
  - [29] M. Friedlander & M. Saunders. *Active-Set Procedures for Basis Pursuit and Related Problems*. Basis Pursuit Algorithm implementation in MATLAB available online.
-