

# Phd Thesis

M.C. Orlando Miguel Medina Cázares

November 10, 2014



# Contents

<b>List of Figures</b>	<b>4</b>
<b>List of Figures</b>	<b>5</b>
<b>1 Fringe pattern demodulation</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Fringe pattern temporal carrier . . . . .	2
1.3 Phase-shifting algorithms (PSAs) . . . . .	3
<b>2 Regularized iterative least-squares algorithm for phase-shifting interferometry</b>	<b>5</b>
2.1 Introduction . . . . .	5
2.2 Classical Least-Squares . . . . .	6
2.3 Full-field 2D least-squares method . . . . .	7
2.4 Numerical Experiments . . . . .	9
2.5 Experimental Results . . . . .	10
2.6 Comments and conclusions . . . . .	10
<b>3 Regularized self-tuning phase demodulation for phase-shifting interferometry with arbitrary phase shifts</b>	<b>13</b>
3.1 Motivation . . . . .	13
3.2 Introduction . . . . .	13
3.3 Method . . . . .	15
3.4 Numerical experiments and results . . . . .	17
3.5 Conclusions . . . . .	19
<b>4 Robust adaptive phase-shifting demodulation for testing moving wavefronts</b>	<b>21</b>
4.1 Introduction . . . . .	21
4.2 Method . . . . .	22

---

4.3	Tests and results . . . . .	26
4.4	Discussion and commentaries . . . . .	28
<b>5</b>	<b>Removing detuning distortions of wrapped phase by using robust quadrature filters</b>	<b>31</b>
5.1	Introduction . . . . .	31
5.2	Phase detuning correction method (PDCM) . . . . .	32
5.3	A priori local frequency calculation . . . . .	35
5.4	Tests and experimental examples . . . . .	36
5.5	Discussion and commentaries . . . . .	38
5.6	Appendix. Local frequency calculation in presence of noise . .	39
	<b>Bibliography</b>	<b>43</b>

# List of Figures

2.1 Numerical examples. a) One of four simulated interferogram sequences. b) Recovered phase map using classic least-squares. c) Recovered phase map using our proposed <i>Full-field 2D least-squares</i> . . . . .	7
2.2 Experimental results. a) One of four experimental interferogram sequences. b) Recovered phase map using classic least-squares. c) Recovered phase map using our proposed <i>Full-field 2D least-squares</i> . . . . .	9
3.1 Interferogram sequence and the recovered phase. (A) shows the recovered phase and error using the regularized self-tuning method proposed here. (B) shows the recovered phase and error using the AIA method[15]. The error shown (in radians) is the standard deviation respecting the true phase map. The interferogram frames has a size of $512 \times 512$ . . . . .	18
4.1 Numerical examples. From (a) to (b) we show the wrapped wavefronts generated with Eq. (11). From (e) to (h) we show the estimated wavefronts using the RAPS algorithm presented here. From (i) to (l) we show the estimated wavefronts using the AIA. Above each estimated wavefront we show the estimation error. . . . .	25
4.2 Experimental setup. Electronic Speckle Pattern Interferometry (ESPI) setup. The object under test is a metal plate perturbed with a horn. The reference mirror has attached a piezoelectric transducer (PZT) as phase shifter. . . . .	27

4.3	Experimental tests. From (a) to (d) we show the interferogram sequence. From (e) to (h) we show the estimated wavefront using the AIA, and from (i) to (l) we show the estimated wavefront using the RAPS. . . . .	28
4.4	Phase difference with respect to x (partial approximated derivative). The dynamic range of the difference is quantized to the gray levels 1, 102, 153, 203 and 255, for a better appreciation of the detuning error. (a) shows the phase difference corresponding to the AIA and (b) shows the difference corresponding to RAPS method. . . . .	29
5.1	Simulated wrapped phase comparison. (a) Synthetic input wrapped phase with detuning error. (b) Recovered wrapped phase map using the proposed PDCM. . . . .	33
5.2	Comparison between an unwrapped row of the distorted input and the enhanced phase; the dashed line is the unwrapped phase with detuning error, and the solid line is the unwrapped phase processed with the PDCM. Note: The noise had been removed from the original distorted wrapped phase in Fig. 5.1(a) for clarity and comparison purposes. . . . .	34
5.3	Unwrapped simulated phase comparison of Fig. 5.1. (a) Shows the unwrapped phase with detuning error of Fig. 5.1(a). (b) Shows the recovered unwrapped phase of Fig. 5.1(b) after using the proposed PDCM. Note: The noise had been removed from the original distorted wrapped phase in Fig. 5.1(a) for clarity and comparison purposes. . . . .	35
5.4	Experimental wrapped phase comparison. (a) Shows the experimental input wrapped phase with detuning error. (b) Shows the recovered wrapped phase map using the proposed PDCM. . . . .	37
5.5	Comparison between an unwrapped row of the distorted input and the enhanced phase; the dashed line is the phase with detuning error, and the solid line is the phase processed by the PDCM. . . . .	37
5.6	Recovered wrapped phase using the proposed PDCM with miscalculated frequencies. . . . .	39
5.7	Wrapped phase comparison after applying a low-pass filter to the noisy wrapped phase in Fig. 5.1(a). (a) Shows the result of applying a gaussian filter directly to our wrapped phase, while in (b) we see the result of applying Eq. 5.10. . . . .	40

- 5.8 Example of the frequency change along the refinement process of the distorted wrapped phase. For this sequence, we show the  $u(x, y)$  frequency calculation using Eq. 5.6. . . . . 41



# Chapter 1

## Fringe pattern demodulation

### 1.1 Introduction

A fringe pattern is defined as a sinusoidal signal where a continuous map, analogous of the physical quantity being measured, is phase-modulated by an interferometer, Moire' system, etc. An ideal stationary fringe pattern is usually modeled by

$$I(x, y) = a(x, y) + b(x, y)\cos[\phi(x, y)], \quad (1.1)$$

where  $a(x, y)$  and  $b(x, y)$  are the background and local contrast functions, respectively; and  $\phi(x, y)$  is the searched phase function.

Analyzing Eq. (1.1) one can see that the phase function  $\phi(x, y)$  cannot be directly estimated since it is screened by two other functions,  $a(x, y)$  and  $b(x, y)$ . Additionally,  $\phi(x, y)$  can only be determined modulo  $2\pi$  because the sinusoidal fringe pattern  $I(x, y)$  depends periodically on the phase ( $2\pi$  phase ambiguity); and its sign cannot be extracted from a single measurement without a priori knowledge (sign ambiguity) due to even character of the cosine function [ $\cos(\phi) = \cos(-\phi)$ ]. Finally, in all practical cases some noise  $\eta(x, y)$  is introduced in an additive and/or multiplicative way, and the fringe pattern may suffer from a number of distortions, degrading its quality and further screening the phase information [142,18].

It must be noted that even if careful experimental setups could prevent the screening of  $\phi(x, y)$  due to the unknown signals  $a(x, y)$ ,  $b(x, y)$  and  $\eta(x, y)$ , one would still have to deal with the sign ambiguity and the  $2\pi$  phase ambiguity. Because of these ambiguities, the solution for this inverse problem is not unique; this is because several phases can produce exactly the same sinusoidal signal.

If we rewrite Eq. (1.1) by means of the complex representation of the cosine function

$$I(x, y) = a(x, y) + \frac{1}{2}b(x, y)[e^{i\phi(x, y)} + e^{-i\phi(x, y)}], \quad (1.2)$$

and one is able to isolate one of the analytic signals in Eq. (1.2), lets say  $\frac{1}{2}b(x, y)e^{i\phi(x, y)}$ , we have

$$\tan \hat{\phi}(x, y) = \frac{\text{Im}\{(1/2)b(x, y)e^{i\phi(x, y)}\}}{\text{Re}\{(1/2)b(x, y)e^{i\phi(x, y)}\}}. \quad (1.3)$$

Computing the arc-tangent of the above formula one obtains a wrapped estimation of the phase under study:  $\phi(x, y) \bmod 2\pi$ . Thus, the final step of this fringe pattern demodulation process usually involves an additional phase unwrapping process. Nevertheless, when working with good quality data this last step is straightforward. Next we will illustrate the easiest way to obtain these analytic signals.

## 1.2 Fringe pattern temporal carrier

A fringe pattern obtained as the output of a measuring system may be modified by the optoelectronic-mechanical hardware (sensors and actuators) and software (virtual sensors and actuators) of the system [83]. With these modifications one is able to introduce known changes in the argument of the sinusoidal signal:

$$I(x, y, t) = a(x, y) + b(x, y)\cos[\phi(x, y) + c(x, y, t)], \quad (1.4)$$

where  $c(x, y, t)$  is a known function (typically a reference plane) and it is called the spatio-temporal carrier of the interferogram. The spatial and/or temporal carriers are of extreme importance in modern interferometry: first of all, its presence allows to solve the sign ambiguity since in general  $\cos(\phi + c) \neq \cos(-\phi + c)$ . They also allow to isolate the analytic signal  $\frac{1}{2}b(x, y)e^{i\phi(x, y)}$ . In phase-shifting interferometry (PSI) the linear temporal carrier can be written as:

$$c(x, y, t) = \alpha k, \quad (1.5)$$

where  $\alpha$  is the temporal lineal carrier, typically  $\alpha = \pi/2$ , and  $k$  is the discrete temporal variable. The independent temporal variable  $k$  represents the  $k$ -frame of the PSI sequence. In this context, knowing the temporal carrier

$\alpha$  (phase step as known in PSI), the objective of the PSI demodulation methods is to estimate the complex field

$$f(x, y) = \frac{1}{2}b(x, y)e^{i\phi(x, y)}, \quad (1.6)$$

of the interferometric signal at the  $(x, y)$  pixel. Then, the phase at  $(x, y)$  is obtained as:

$$\hat{\phi}(x, y) = \arg[f(x, y)] = \arctan\left(\frac{\sin[\phi(x, y)]}{\cos[\phi(x, y)]}\right), \quad (1.7)$$

which is an alternative form of Eq. (1.3).

In the following sections we will analyze several methods to estimate the analytic signal  $f(x, y)$ .

### 1.3 Phase-shifting algorithms (PSAs)

In general, a phase-shifting algorithm (PSA) can be described as quadrature linear filter which is completely characterized by its impulse response function,  $h(k)$ , or by its frequency transfer function (FTF) in the Fourier domain,  $H(\omega)$ :

$$h(t) = \sum_{n=0}^{N-1} c_n \delta(k - n), \quad (1.8)$$

$$H(t) = \sum_{n=0}^{N-1} c_n e^{-i\omega n}, \quad (1.9)$$

where  $c_n \in \mathbb{C}$ . In order to be a valid PSA, this FTF must fulfill the so-called quadrature conditions:

$$H(0) = H(-\alpha) = 0, \quad H(\alpha) \neq 0. \quad (1.10)$$

Thus, the application of this quadrature filter produces the following analytic signal for  $k = N - 1$  (where all the available data is involved):

$$A_0(x, y)e^{i\phi} = \sum_{n=0}^{N-1} c_n I(x, y, n). \quad (1.11)$$

From the above analytic signal, we obtain the searched phase  $\phi(x, y)$ , modulo  $2\pi$ , by computing its angle as Eq. (1.7). For completeness, solving for  $\phi(x, y)$  is found the so-called arc-tangent formulation of a PSA:

$$\hat{\phi}(x, y) \bmod 2\pi = \arctan\left(\frac{\operatorname{Im}\{c_0 I_0 + c_1 I_1 + \dots + c_{N-1} I_{N-1}\}}{\operatorname{Re}\{c_0 I_0 + c_1 I_1 + \dots + c_{N-1} I_{N-1}\}}\right) \quad (1.12)$$

The above results are obtained under the assumption that the quadrature filter and the temporal samples of the interferogram are perfectly tuned at the same frequency  $\alpha$  (radians per sample). When the actual sampling frequency is  $\alpha + \Delta$ , that is when some detuning error occurs, the resulting phase estimation is given by

$$\hat{\phi}(x, y) = \phi(x, y) - D(\Delta) \sin[2\phi(x, y)], \quad (1.13)$$

where the amplitude of the detuning-error is given by the ratio

$$D(\Delta) = \frac{|H(-\alpha - \Delta)|}{|H(+\alpha + \Delta)|}, \quad (1.14)$$

for  $|\Delta/\alpha| \ll 1$ . This equation shows that detuning-error causes the estimated phase to be distorted by a component with twice the original fringes frequency. Moreover, actual signals are always distorted by some amount of random noise so a more cautious approach it is to consider additive white-noise distortion (with random components in every spectral frequency). Under this assumption, when filtering with quadrature linear filters, the signal-to-noise power ratio gain of the PSA is given by:

$$G_{S/N}(\alpha) = \frac{|H(\alpha)|^2}{(1/2\pi) \int_{-\pi}^{\pi} |H(\alpha)|^2 d\alpha}. \quad (1.15)$$

## Chapter 2

# Regularized iterative least-squares algorithm for phase-shifting interferometry

### 2.1 Introduction

Phase Shifting Interferometry (PSI) demodulation methods are useful 1D temporal linear systems that allow us to recover the modulating phase of the PSI sequence. When the number of samples is small, typically between 3 and 15, we speak of PSI methods [4, 38], but they require a constant phase and do not tolerate missing data. On the other hand, when the number of samples is large, between  $10^2$  and  $10^3$ , we speak of temporal analysis methods [25, 34, 11] which have many problems with the interferogram borders and missing data. Another possibility for analyzing the temporal signal is the use of a running PSI method tuned at the carrier frequency [25, 34, 11, 33, 32, 48]. For example if we use a tree step PSI method we could demodulate the phase locally for each of three consecutive samples. Although this method will deal well with borders, missing data cannot be handled and can even impede the use of this strategy. In experimental methods missing data appear in the case of a saturated signal and also in heterodyne temporal speckle-pattern interferometry when temporal decorrelation appears. Also missing data and discontinuities due to occlusions or shadows are very common in projected fringe profilometry. Besides these problems, noise is another important issue to solve; for example, in speckle techniques [4, 11] noisy interferograms are obtained, in consequence, recovered phase have to be treated to obtain a clean phase easy to unwrap.

Hence, in this paper we are going to present a full-field 2D linear demodulation method that uses in conjunction the temporal and spatial information in order to recover a clean phase, while interpolates empty small sections of missing data from the image space all with low computational time and in the same process.

## 2.2 Classical Least-Squares

Each  $(x, y)$  pixel of the PSI sequence is a 1D temporal discrete interferometric signal modeled in the following way,

$$\begin{aligned} I_{x,y}(k) &= a_{x,y} + b_{x,y} \cos(\phi_{x,y} + k\alpha) \\ &= a_{x,y} + c_{x,y} \sin(k\alpha) - s_{x,y} \cos(k\alpha), \end{aligned} \quad (2.1)$$

where  $c_{x,y} = b_{x,y} \cos(\phi_{x,y})$  and  $s_{x,y} = b_{x,y} \sin(\phi_{x,y})$  are the quadrature components of the 1D temporal interferometric signal,  $k$  is the discrete temporal variable,  $a_{x,y}$  the background illumination,  $b_{x,y}$  the modulation term,  $\alpha$  the phase step or temporal carrier, and  $\phi_{x,y}$  the modulating phase sought at the  $(x, y)$  pixel; note that all these variables are scalars. The independent temporal variable  $k$  represents the  $k$ -frame of the PSI sequence. In this context, knowing the temporal carrier  $\alpha$  (phase step as known in PSI), the objective of the PSI demodulation methods is to estimate the quadrature components  $c_{x,y}$  and  $s_{x,y}$  of the interferometric signal at the  $(x, y)$  pixel. Then, the phase at  $(x, y)$  is obtained as:

$$\phi_{x,y} = \arctan \left( \frac{s_{x,y}}{c_{x,y}} \right). \quad (2.2)$$

Scanning all pixels in this way, we obtain the wrapped phase image of the PSI sequence. One of the first approaches to demodulate a PSI sequence was the least-squares model for PSI [28, 9, 15, 17]. The least-squares model (cost function) for PSI is the following,

$$U(a_{x,y}, c_{x,y}, s_{x,y}) = \sum_{k=0}^{N-1} [a_{x,y} + c_{x,y} \sin(k\alpha) - s_{x,y} \cos(k\alpha) - I_{x,y}(k)]^2, \quad (2.3)$$

where  $I_{x,y}(k)$  is the observed value of the  $k$ -frame at the  $(x, y)$  pixel modeled as in Eq. (2.1). To have a well-posed mathematical model for Eq. (2.3), it is necessary to have at least three interferograms in the PSI sequence; that

is,  $N \geq 3$ . The parameters  $c_{x,y}$  and  $s_{x,y}$  that minimize Eq. (2.3) are the quadrature components used in Eq. (2.2) to obtain the phase. To minimize Eq. (2.3), we solve a  $3 \times 3$  linear equation system where  $a_{x,y}$ ,  $c_{x,y}$  and  $s_{x,y}$  are the unknowns. As the temporal interferometric signal of each pixel has the same model [see Eq. (2.1)], the solution of the linear equation system is always the same. This allows us to demodulate the interferogram sequence using very simple closed mathematical forms [4]. However, as shown in Eq. (2.3), this system never uses the spatial information but the temporal one; in fact, not one of the temporal PSI demodulation methods uses the spatial information [38]. Reason why, using temporal PSI demodulation methods we filter only through the interferogram sequence, but not through the interferogram image space. As a consequence, temporal PSI demodulation methods can not remove unwanted spatial harmonics or noise that may be introduced while recording the interferograms.

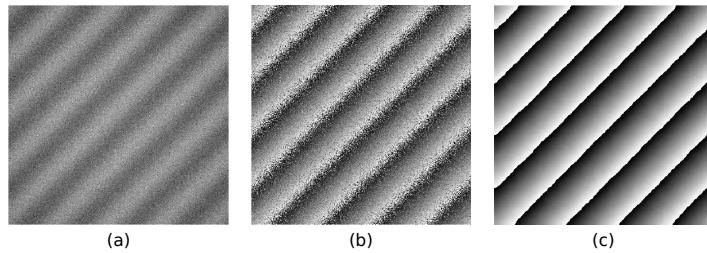


Figure 2.1: Numerical examples. a) One of four simulated interferogram sequences. b) Recovered phase map using classic least-squares. c) Recovered phase map using our proposed *Full-field 2D least-squares*.

### 2.3 Full-field 2D least-squares method

Regularization systems are very useful full-field systems that can use all the information needed to obtain the data sought as expected. In PSI, we can use these techniques to include the temporal and spatial information to recover the modulating phase as a smooth 2D function, removing unwanted harmonics and noise. Actually, regularization techniques have been used before in PSI for these purposes, the first were Marroquin et al. [25, 26, 27] and more recently others [40, 41, 22, 33, 45, 32, 48]. However, the approach used in those works obtained non-linear systems with a considerable computational work load. Besides, these algorithms need a pre-processing method

to remove background illumination in order to demodulate a correct phase. In our case, this preprocess is not needed and, also and more important, we will maintain the linearity of the least-squares cost function (2.3), adding spatial constraints to recover the wrapped modulating phase while removing noise and unwanted harmonics present in the interferograms [25], besides interpolating small sections of missing data. These constraints will penalize the spatial variations of the quadrature components  $c_{x,y}$  and  $s_{x,y}$  by using first order potentials as regularization terms. Proceeding in this way, the *Full-field 2D least-squares* cost function for PSI is the following:

$$\begin{aligned} U(\mathbf{a}, \mathbf{c}, \mathbf{s}) = & \sum_{k=0}^{N-1} \sum_{x,y \in L} [a_{x,y} + c_{x,y} \sin(k\alpha) - s_{x,y} \cos(k\alpha) - I_{x,y}(k)]^2 M_{x,y} \\ & + \lambda \sum_{x,y \in L} [(c_{x,y} - c_{x-1,y})^2 + (s_{x,y} - s_{x,y-1})^2] \\ & + \mu \sum_{x,y \in L} (a_{x,y} - a_{x-1,y})^2, \end{aligned} \quad (2.4)$$

where  $M_{x,y}$  is a binary mask with valid measurement,  $\lambda$  is the regularization parameter that penalizes the spatial variations of quadrature components  $\mathbf{c}$  and  $\mathbf{s}$ , and  $\mu$  penalizes the spatial variations of background illumination  $\mathbf{a}$ . Note that in this case, the parameters  $(\mathbf{a}, \mathbf{c}, \mathbf{s})$  of the cost function in Eq. (2.4) are scalar fields with dimensions  $L_x \times L_y$  and elements  $a_{x,y}$ ,  $c_{x,y}$  and  $s_{x,y}$ , respectively, while the parameters  $(a_{x,y}, c_{x,y}, s_{x,y})$  of the cost function in Eq. (2.3) are just scalars. As with the least-squares cost function of Eq. (2.3), here, at least three interferograms are needed in the sequence in order to have a well-posed mathematical model. To minimize Eq. (2.4), in order to obtain the quadrature components  $\mathbf{c}$  and  $\mathbf{s}$  that will give us the modulating phase, we need to solve a linear equation system of  $3(L_x \times L_y)$  equations and  $3(L_x \times L_y)$  unknowns. Compared with the  $3 \times 3$  linear equation system of Eq. (2.3), the linear equation system of Eq. (2.4) is larger; however, solving this linear equation system is not so complicated when using numerical methods such as *Gauss-Seidel*. One of the advantages of the *Gauss-Seidel* method is that it is numerically stable, and it is not necessary to build the associated matrix of the linear equation system; besides, the *Gauss-Seidel* method can be programmed for today's modern parallel processors, such as the *Graphics Processing Unit* (GPU), speeding up the minimization process. For illustration purposes, in this paper we programmed the algorithm in *C++* language.

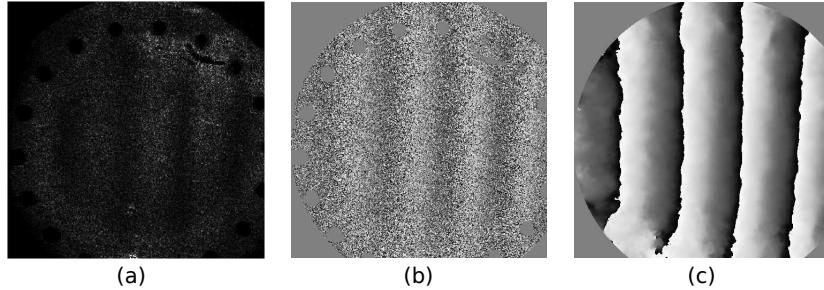


Figure 2.2: Experimental results. a) One of four experimental interferogram sequences. b) Recovered phase map using classic least-squares. c) Recovered phase map using our proposed *Full-field 2D least-squares*.

## 2.4 Numerical Experiments

To show the performance of the *Full-field 2D least-squares* algorithm, we simulated a PSI sequence of four interferograms of  $512 \times 512$  pixels in the following way:  $I_{x,y}(k) = a_{x,y} + b_{x,y}\cos(\phi_{x,y} + k\alpha) + \eta_{x,y}$ , for  $k = 0, 1, 2, 3$  and  $\alpha = \pi/2$ . The modulated phase  $\phi$  was modeled as a plane using the following expression:  $\phi_{x,y} = 0.05x + 0.05y$ . The background illumination term  $a$  was modeled as a parabola centered at pixel (256,256) of the image frames with a dynamic range between 0 and 1. The  $b$  term was set to 1. Last, we added a random field of white noise  $\eta$ , with mean  $\gamma = 0$  and variance  $\sigma^2 = 1$ . In Fig. 2.1(a), we see the first interferogram of the simulated sequence. Figures 2.1(b) and 2.1(c) show the wrapped phase using classic least-squares [9] and the *Full-field 2D least-squares* method, respectively. To estimate the wrapped phase in Fig. 2.1(c), we solve the linear system in Eq. (2.4) using the Gauss-Seidel method and setting  $\lambda$  and  $\mu$  to 50. The number of iterations was 500. For this example, the mask  $M_{x,x}$  in Eq. (2.4) is one over all the image; since all the image it is valid information. Computational time was 4.6934 seconds, on a PC with an Intel Core i7 processor and 8 GB RAM memory . We can see in these figures that our proposed method recovers a phase with much less noise than the classic least-squares method, given the regularization terms in Eq. (2.4).

## 2.5 Experimental Results

Now, we are going to show the performance of our method with experimentally obtained interferograms and compare it qualitatively with the classical least-squares method. The interferogram sequence was generated using an ESPI technique, and the wave-front under test was modified applying pressure. For the phase step, a phase-shift of  $\pi/2$  radians was introduced. The object under test was a circular metal plate with circular perforations all along its edge. In order to increase reflexion, we coated the plate with white powder, except for a small part, as can be seen in Fig. 2.2(a). Fig. 2.2(a) shows the first experimental phase-shifting interferogram of a 4 samples sequence. In Fig. 2.2(b), we show the wavefront estimation of the classic least-squares method, while in Fig. 2.2(c), we see the wave-front estimation of the *Full-field 2D least-squares* method proposed here. Computational time in this case was 7.6483 seconds using the PC described above. As we can see, the proposed method was able to estimate a phase free of noise. Another significant feature of this algorithm is that in sections where there is no information, such as black circles and scratches, the algorithm was able to fill-up the empty spaces satisfactorily; this is because it takes into account the neighboring pixel information and the regularization terms; very useful feature in the aforementioned cases.

## 2.6 Comments and conclusions

The calculation of a free-noise phase in PSI is very useful, since it allows us to use simple algorithms to unwrap the phase. Normally, to get a soft phase, we need to filter the interferogram samples or the output phase to remove the noise. The problem of this process is that we may be removing important information during the filtering. For this reason, the *Full-field 2D least-squares* algorithm represents a significant improvement to the classical least-squares method. Besides, as we have seen before, the presented algorithm is capable of interpolating small empty spaces of missing data, since it takes into account the temporal and spatial information. Therefore, all results presented in this paper can be directly applied to the spatial case where missing data and discontinuities are present. Examples of this are occlusions or shadows in projected fringe profilometry, temporal decorrelation and saturated signal in heterodyne temporal speckle-pattern interferometry.

Previous to this work, all phase-shifting algorithms only use a single pixel signal to estimate the wave-front under test, regardless of adjacent

information. This paper presents the usefulness of taking into account the temporal and spatial information in conjunction to estimate a best phase map. It is important to highlight that the functional of Eq. (2.4) is a linear system; therefore, it is stable and easy to compute. In conclusion, we present a full-field 2D linear demodulation algorithm able to recover a clean phase and also able to interpolate small empty sections of information, all with low computational time and in the same process.



# Chapter 3

## Regularized self-tuning phase demodulation for phase-shifting interferometry with arbitrary phase shifts

### 3.1 Motivation

In this work, we develop a regularization technique to demodulate a phase-shifting interferogram sequence with arbitrary inter-frame phase shifts. With this method, we can recover the modulating phase and inter-frame phase shifts in the same process. As all phase-shifting algorithms, the assumption is that the wavefront under test does not change over time but the phase-shifting introduction can vary in a non constant way. A notable characteristic of this demodulation method is that it not only can recover the modulating phase, but also it is capable of filtering-out large quantities of corrupting noise. We will show numerical experimental results and comparisons with other already published method to see the performance of the herein developed demodulation technique.

### 3.2 Introduction

Nowadays, Phase Shifting Interferometry (PSI) techniques are one of the most used techniques in optical metrology [4]. In PSI, one obtains a small sequence of at least 3 interferograms with a phase-shifting among them [1].

To recover the modulating phase there are standard demodulation PSI methods; the well known 3-, 4-, and 5-steps phase-shifting algorithms. Knowing the inter-frame phase-shifts (or temporal carrier) the standard methods recover the modulus  $2\pi$  phase map with the minimum possible error [4, 8, 39]. If we do not know the phase-shifts exactly, we obtain a phase map with an unavoidable detuning error whose magnitude depends on the number of interferograms employed and how far we are from the actual phase-shifts [39, 36, 1, 20, 19]. This unfortunate case can occur when the optical interferometer setup is uncalibrated or perturbations from the environment affect the interferometer's optical path. For example, for most phase shifters such as a PZT there is a repeatability problem from hysteresis, non linearity, and temperature linear drift [1, 3]. Curiously, first phase-shifting algorithms were self-tuning nonlinear algorithms. Other approaches, propose error compensating algorithms to reduce detuning errors that basically use redundant data such as the Schwider-Hariharan 5-steps algorithm [36, 12, 42], and more recently by constructing a wide-band frequency response of the phase-shifting algorithm as the 7-steps algorithm shown in [6]. Further methods use the Fourier transform in order to estimate the inter-frame phase-shifts [20, 18], and other are based on the least-squares scheme estimating iteratively the inter-frame phase-shifts and phase [17, 46]. In Ref. 16[7], we presented an approach that estimates the local temporal carrier (the phase-shift) as the average of the phase difference between two consecutive phase maps obtained from two realizations of the tunable 3-steps algorithm. What we are going to show in this work, is a regularized self-tuning demodulation technique that obtains the analytical image (complex interferogram) and inter-frame phase-shifts from an interferogram sequence. Thus, we can recover the modulating phase modulus  $2\pi$  and the inter-frame phase shifts in the same process. Here, it is not necessary to know the inter-frame phase-shifts. This inter-frame phase-shifts can vary arbitrary. The main difference between the demodulation method presented here, and the reported in [17, 46, 21], is that the herein demodulation method is based on a regularization technique that is able to remove noise from its input and is robust to non constant modulation variations, which is an issue that introduce errors in methods of works [17, 46]. Besides, we do not require estimate the fringe orientation as the method of work [21].

### 3.3 Method

In general, an interferogram sequence with arbitrary inter-frame phase-shifts can be modeled as

$$I_k(x, y) = a(x, y) + b(x, y)\cos(\phi(x, y) + \alpha_k), \quad k = 0, 1, 2, \dots, N - 1, \quad (3.1)$$

where  $I_k(x, y)$  is the intensity at the site  $(x, y)$  of the  $k$ -interferogram in a sequence of  $N - 1$  interferograms, being  $a(x, y)$  its background illumination,  $b(x, y)$  its contrast or modulation,  $\phi(x, y)$  the modulating phase under test and  $\alpha_k$  the phase-shifting of the  $k$ -interferogram. We can remove the background illumination of each interferogram in the following way:

$$I'_k(x, y) = I_k(x, y) - [I_k * h](x, y), \quad k = 0, 1, 2, \dots, N - 1, \quad (3.2)$$

where  $h(x, y)$  is the impulse response of a low-pass filter such as a Gaussian or mean filter and  $*$  is the convolution operator[14]. Making this, the new interferogram sequence looks like

$$I'_k(x, y) = b'(x, y)\cos(\phi(x, y) + \alpha_k), \quad k = 0, 1, 2, \dots, N - 1. \quad (3.3)$$

The main idea of the regularized self-tuning demodulation method that we show here comes from the article of Marroquin et al. [25]. The demodulation process presented by Marroquin et al., minimizes a nonlinear system that estimates the complex field of the first interferogram and its local spatial frequencies. We, unlike the Marroquin et al. work, estimate the inter-frame phase-shifts from the interferogram sequence. Therefore, our demodulation method minimizes the following quadratic functional

$$\begin{aligned} U(f, \alpha) &= \sum_{(x,y)} (\varphi(x, y) - I'_0(x, y))^2 + \sum_{k=1}^{N-1} \sum_{(x,y)} [\frac{1}{2}[f(x, y)e^{i\alpha_k} + f^*(x, y)e^{-i\alpha_k}] - I'_k(x, y)]^2 \\ &\quad + \lambda \sum_{(x,y)} [|D_x[f(x, y)]|^2 + |D_y[f(x, y)]|^2], \text{ nonumber} \end{aligned} \quad (3.5)$$

where  $f = \{f(x, y) = \varphi(x, y) + i\psi(x, y) : (x, y) \in L\}$  is the complex field,  $i = \sqrt{-1}$ , and  $f^*$  its complex conjugated. The sums with the notation  $(x, y)$  underneath, runs over all valid sites  $(x, y)$  of the interferograms. Operators  $D_x[]$  and  $D_y[]$  takes the first order differences along  $x$  and  $y$  direction, as follows:

$$D_x[f(x, y)] = f(x, y) - f(x - 1, y) + f(x, y) - f(x + 1, y), \quad (3.6)$$

$$D_y[f(x, y)] = f(x, y) - f(x, y - 1) + f(x, y) - f(x, y + 1). \quad (3.7)$$

The regularization parameter  $\lambda$  controls the smoothness of the complex field [25, 27]. The first and second terms are the data terms, and the third term is the regularization term. Our reference is the first interferogram, therefore, we consider that its phase-shifting is  $\alpha_0 = 0$ . This is the reason of the first data term, which results when  $k = 0$ , and therefore, the second data term starts in  $k = 1$ . The minimization process of the functional (4) leads to a robust to noise nonlinear phase-shifting algorithm of N-steps that can recover the modulating phase and inter-frame phase shifts. With the complex field  $\hat{f}$  and phase-shifts  $\alpha$  that minimize (4), the modulating phase is recovered as

$$\phi(x, y) = \arg[\hat{f}(x, y)] = \arctan \left[ \frac{\hat{\psi}(x, y)}{\hat{\varphi}(x, y)} \right]. \quad (3.8)$$

The minimization of functional (4), turns us to a nonlinear system that mathematically is impossible to solve by a direct numerical method. The dimension problem is  $m \times n \times N$ , where  $m \times n$  is the interferogram dimension and  $N$  is the number of interferograms. For nonlinear systems, the iterative steepest-descent algorithm can converge to local minimums if its parameters are set adequately [31], but its converge speed results very slow in this case. Then, we split the problem in two: the linear part and the nonlinear part. The linear part are the equations that result by making zero the following partials:  $\frac{\partial U}{\partial \varphi(x, y)}$  and  $\frac{\partial U}{\partial \psi(x, y)}$ , for all  $(x, y) \in L$ . The nonlinear part are the equations that result by making zero the following partial:  $\frac{\partial U}{\partial \alpha_k}$ , for  $k = 0, 1, 2, \dots, N - 1$ . Thus, to speed up the minimization process, our iterative minimization strategy combines in each iteration the Gauss-Seidel update for the linear part, and the steepest-descent update for the nonlinear part. Then, the iterations of our minimization strategy are given with the following updates:

$$\varphi^{n+1}(x, y) = \text{Solve for } \varphi^n(x, y) \text{ Eq.} \left[ \frac{\partial U(\varphi^n + i\psi^n, \alpha^n)}{\partial \varphi(x, y)} = 0 \right]; \forall (x, y) \in L \quad (3.9)$$

$$\psi^{n+1}(x, y) = \text{Solve for } \psi^n(x, y) \text{ Eq.} \left[ \frac{\partial U(\varphi^n + i\psi^n, \alpha^n)}{\partial \psi(x, y)} = 0 \right]; \forall (x, y) \in L \quad (3.10)$$

$$\alpha_k^{n+1} = \alpha_k^n - \mu \frac{\partial U(\varphi^n + i\psi^n, \alpha^n)}{\partial \alpha_k}, \quad k = 0, 1, 2, \dots, N-1. \quad (3.11)$$

Equations (8) and (9) correspond to the Gauss-Seidel update, and equation (10) is the steepest-descent update. In the appendix at the end of this paper, we show the explicit formulas of these equations. To see this implementation, the reader can download the source code following the web link of Ref. 22. Note: the provided source code of Ref. 22 is for illustration purposes, and it is no optimized as the C-language code used for our numerical experiments.

### 3.4 Numerical experiments and results

To obtain the results presented here, the minimization process described here made 1000 iterations to reach a relative convergence error of  $1.322 \times 10^{-4}$ .

This relative convergence error is calculated as  $\sqrt{\sum(\alpha_k^n - \alpha_k^{n+1})^2}$ , where the sum runs over  $k = 1, 2, 3, \dots, N-1$ ,  $\alpha_k^n$  is the  $k$ -phase-shift estimated in the current iteration and  $\alpha_k^{n+1}$  is the  $k$ -phase-shift of the next iteration. This minimization process, coded and compiled in C-language, took a time of 11.310 seconds for these  $512 \times 512$  interferogram frames in a computer with a 8-cores CPU of 1.73GHz having 8GB of memory RAM. The regularization parameter  $\lambda$  (see Eq. (4)) was set to  $\lambda = 10$ , and the parameter  $\mu$  of the steepest-descent update was  $\mu = \frac{1.2}{512 \times 512}$  (see Eq. (10)). Actually, choosing  $\mu = \frac{1.2}{512 \times 512}$  is a very good parameter for the steepest-descent update of Eq. (10), where  $m \times n$  is the dimension of the interferogram frames. In our numerical experiments, we start always the minimization process with initial values of  $f = 0$ , and  $\alpha_k = k\frac{\pi}{2}$  for  $k = 0, 1, 2, \dots, N-1$ . The interferogram sequence was generated as follows: The  $k$ -frame is given as  $I_k = b(x, y)\cos(\phi(x, y) + \alpha_k) + \eta(x, y)$ , being  $\eta(x, y)$  a random field of white noise with mean  $\gamma = 0$  and variance  $\sigma^2 = 4.84$  radians. The modulation, or contrast term  $b(x, y)$ , was modeled as a parabola centered at pixel (256, 256) of the image frames with a dynamic range between 1 and 3. The inter-frame phase-shifts were generated as  $\alpha_k = \pi + 0.4 * \varepsilon$ , where  $\varepsilon_k$  is a random scalar with a uniform distribution between  $-\pi$  and  $\pi$  radians. Here, we compare our results with the so called Advanced Iterative Algorithm (AIA) presented in Ref. [46] because this method estimates the phase and inter-frame phase-shifts as well, but using other approach. In Table. 1, we show the errors values of the estimated phase-shifts using our regularized method and the estimated using the AIA method. These errors are calculated as  $|\alpha_k - \hat{\alpha}_k|$ ,

Steps	Proposed Method	AIA Method
0	0	0
1	0.0766	0.0904
2	0.0637	0.7146
3	0.0569	0.7083
4	0.0569	0.5520

Table 3.1: This table shows the error obtained between the phase-shift estimation and the actual phase-shift.

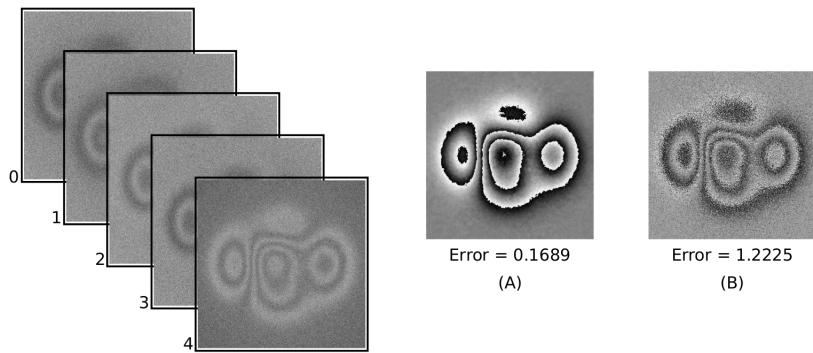


Figure 3.1: Interferogram sequence and the recovered phase. (A) shows the recovered phase and error using the regularized self-tuning method proposed here. (B) shows the recovered phase and error using the AIA method[15]. The error shown (in radians) is the standard deviation respecting the true phase map. The interferogram frames has a size of  $512 \times 512$ .

where  $\hat{\alpha}_k$  and  $\alpha_k$  are the values of the estimated and actual phase-shifts for the  $k$ -frame, respectively. There we can see that our method estimates the inter-frame phase-shifts with less error. On the other hand, in Fig. 1 , we show the interferogram sequence and the recovered phase. Fig. 1.(A) shows the recovered phase using the regularized self-tuning demodulation method presented here, while Fig. 1.(B) shows the recovered phase using the AIA method. We can see in this figure that our proposed regularized self-tuning demodulation method recovers the phase with less noise and error than with the AIA method. The errors shown in Fig. 1.(A) and Fig. 1.(B) are calculated as standard deviation of the difference between the recovered phase map and the true phase map used to generate the interferograms.

### 3.5 Conclusions

We have presented a regularized self-tuning phase-shifting demodulation method for interferogram sequences having arbitrary variations of the inter-frame phase-shifts. This method is robust to non constant spatial modulations. As shown in the results, our demodulation method is able to filter-out noise, and recover the modulating phase and the inter-frame phase-shifts with a minimum error. The demodulation method presented here is a non-linear demodulation method, however, we innovate the minimization strategy by mixing the steepest-descent update with the Gauss-Seidel update. In this way, we were able to speed up the minimization process and obtain the expected results.

## APPENDIX

The iteration updates shown in Eqs. (8), (9) and (10) are given by taking the gradient of (4) and solving in the following way:

$$\varphi^{n+1}(x, y) = \frac{F_r(\varphi^n + i\psi^n, \alpha^n)}{H_r(\alpha^n)} \quad (3.12)$$

$$\psi^{n+1}(x, y) = \frac{F_i(\varphi^n + i\psi^n, \alpha^n)}{H_i(\alpha^n)} \quad (3.13)$$

where

$$\begin{aligned} F_r(\varphi^n + i\psi^n, \alpha^n) &= I_0(x, y) + \sum_{k=1}^{N-1} [I_k \cos(\alpha) - \psi(x, y) \sin(\alpha_k) \cos(\alpha_k)] \\ &\quad + \lambda [\varphi(x-1, y) s(x-1, y) + \varphi(x+1, y) s(x+1, y) \\ &\quad + \varphi(x, y-1) s(x, y-1) + \varphi(x, y+1) s(x, y+1)], \\ F_i(\varphi^n + i\psi^n, \alpha^n) &= \sum_{k=1}^{N-1} [I_k \cos(\alpha) - \varphi(x, y) \sin(\alpha_k) \cos(\alpha_k)] \\ &\quad + \lambda [\psi(x-1, y) s(x-1, y) + \psi(x+1, y) s(x+1, y) \\ &\quad + \psi(x, y-1) s(x, y-1) + \psi(x, y+1) s(x, y+1)], \\ H_r(\alpha) &= \sum_{k=1}^{N-1} \cos^2(\alpha_k) + \lambda [s(x-1, y) + s(x+1, y) + s(x, y-1) + s(x, y+1)] \end{aligned} \quad (3.16)$$

$$H_i(\alpha) = \sum_{k=1}^{N-1} \sin^2(\alpha_k) + \lambda[s(x-1, y) + s(x+1, y) + s(x, y-1) + s(x, y+1)] \quad (3.17)$$

The function  $s(x_0, y_0)$  is an indicator function that is 1 if the point  $(x_0, y_0)$  is into the spatial domain of the interferograms, otherwise it is zero. Now, for the steepest-descent update (the phase-shifts  $\alpha$ ) the iteration update is:

$$\begin{aligned} \alpha_0^{n+1} &= 0 \\ \alpha_k^{n+1} &= \alpha_k^n - \mu \sum_{\forall(x,y) \in L} [\varphi^n(x, y) \cos(\alpha_k^n) + \psi^n(x, y) \sin(\alpha_k^n) - I_k(x, y)] \\ &\quad [\psi^n(x, y) \cos(\alpha_k^n) - \varphi^n(x, y) \sin(\alpha_k^n)]. \end{aligned} \quad (3.18)$$

**Note:** suppose that  $\alpha$  has the inter-frame phase-shifts that minimize (4). Its negative values minimize (4) as well. Then, while minimizing (4) it is possible obtain phase-shift values that looks different to the actual phase-shift values. This is not a problem, since we actually are interested in the modulating phase of the interferograms. However, it is always worth fix the inter-frame phase-shifts obtained in the following way:

$$\hat{\alpha}_k = \begin{cases} \hat{\alpha}_k & \text{if } |\hat{\alpha}_k - \hat{\alpha}_{k-1}| < \pi \\ \hat{\alpha}_k - 2\pi & \text{if } \hat{\alpha}_k - \hat{\alpha}_{k-1} > \pi \\ \hat{\alpha}_k + 2\pi & \text{if } \hat{\alpha}_k - \hat{\alpha}_{k-1} < -\pi \end{cases} \quad (3.20)$$

for  $k = 1, 2, 3 \dots N-1$ , in order to have our inter-frame phase-shifts within the variation range  $(-\pi, \pi)$ .

## Chapter 4

# Robust adaptive phase-shifting demodulation for testing moving wavefronts

### 4.1 Introduction

Phase-Shifting Interferometry (PSI) is a very known technique designed for testing static wavefronts. In PSI, it is generated an interferogram sequence of at least 3 interferograms with a phase shifting between them. To recover the modulating phase the well known phase-shifting demodulation methods are used [2, 28, 36, 3, 12, 6]. When the wavefront under test remains static and the phase shifts are introduced correctly, these phase-shifting methods recover the modulating phase without error. However, in optical interferometer setups the optical path is easily affected by environment perturbations. When environment perturbations exist, the wavefront under test is moving and the phase-shifting algorithms introduce an unavoidable detuning error [39, 38]. Nowadays, we can deal with a moving wavefront by taking all the interferograms in the same instant of time, for example, by using pixelated polarized cameras [37, 16]. But, today this technology is expensive and is patent protected. To deal with the miscalibrations of the optical set up, previous works propose self-tuning phase-shifting demodulation methods [47, 46, 7]. However, all the previous published works for PSI assume that the wavefront under test remains static and there is a global phase-shifting for all pixels of the interferograms. This is not true when environment perturbations affects the wavefront under test. Suppose that you have a PSI optical setup, but, the wavefront under test is perturbed by the

environment in such a way that it is moving. The method presented here is a robust adaptive phase-shifting demodulation method that let us demodulate the interferogram sequence tolerating the movements of the wavefront under test and miscalibrations from the PSI setup. This method allows each pixel have its own phase-shifting, reducing considerably detuning errors and improving the estimation of the modulating phase. To show the performance of the Robust Adaptive Phase-Shifting (RAPS) method presented here, we will present tests and results from simulated and experimentally obtained data.

## 4.2 Method

The interferometric phase-shifting signal for a single pixel has the classic model found in all papers about phase-shifting. In that case, it is assumed that all pixels has the same phaseshifting, therefore the interferometric signal for any pixel is the following:

$$I_k(x, y) = a(x, y) + b(x, y)\cos[\theta_0(x, y) + \omega_0 k], \quad (4.1)$$

where  $I_k(x, y)$  is the  $k$ -interferogram of  $M \times N$  pixels,  $a(x, y)$  is its background illumination,  $b(x, y)$  its contrast or modulation term,  $\theta_0(x, y)$  is the wavefront under test and  $\omega_0$  is the phaseshifting introduced by the PSI system for all pixels. When the wavefront under test is perturbed by environment, it is moving and the movements affects the phase shift on each pixel of the interferograms in the following way:

$$I_k(x, y) = a(x, y) + b(x, y)\cos[\theta_0(x, y) + \eta_k(x, y) + \omega_0 k], \quad (4.2)$$

where  $\eta_k(x, y)$  is the environment perturbation. To process this information, we are going to take  $\eta_k(x, y)$  and  $\omega_0$  as  $\beta_k(x, y) = \eta_k(x, y) + \omega_0 k$  in such a way that the interferogram sequence can be rewritten as

$$I_k(x, y) = a(x, y) + b(x, y)\cos[\theta_0(x, y) + \beta_k(x, y)], \quad (4.3)$$

where  $\beta_k(x, y)$  represents the induced non-static phase-shifting variation of the  $k$ -interferogram. Now, we are going to estimate the wavefront  $\omega_0(x, y)$  and the spatial  $\beta_k(x, y)$  variations of each interferogram of the sequence.

The Eq. (3.4) shows the least-squares cost function to recover the wavefront under test  $\omega_0(x, y)$ ; assuming that we know its spatial variations  $\beta_k(x, y)$  for each interferogram.

$$E[a(x, y), f(x, y)] = \sum_{k=0}^{k-1} [a(x, y) + Re\{f(x, y)e^{i\beta_k(x, y)}\} - I_k(x, y)]^2. \quad (4.4)$$

In this equation,  $i = \sqrt{-1}$  and  $f(x, y)$  is a complex value for the  $(x, y)$  site. The operator  $\text{Re}\cdot$  takes the real part of its argument, that is,  $\text{Re}z = \frac{1}{2}(z + z^*)$ ; being  $z$  a complex value and  $z^*$  its complex conjugate.  $K$  is the number of interferograms. By minimizing (3.4) with respect to  $a(x, y)$  and  $f(x, y)$ , the wavefront under test (the modulating phase) is recovered as

$$\hat{\theta}(x, y) = \text{angle}[\hat{f}(x, y)], \quad (4.5)$$

being  $\hat{f}(x, y)$  the complex value that minimizes Eq. (3.4) at  $(x, y)$  site. To minimize Eq. (3.4) we solve the following linear system for each  $(x, y)$  pixel:

$$\begin{pmatrix} K & \sum c_k(x, y) & \sum s_k(x, y) \\ \sum c_k(x, y) & \sum c_k(x, y)^2 & \sum c_k(x, y)s_k(x, y) \\ \sum s_k(x, y) & \sum c_k(x, y)s_k(x, y) & \sum s_k(x, y)^2 \end{pmatrix} \begin{pmatrix} \hat{a}(x, y) \\ \hat{\phi}(x, y) \\ \hat{\psi}(x, y) \end{pmatrix} = \begin{pmatrix} \sum I_k(x, y) \\ \sum I_k(x, y)C_k(x, y) \\ \sum I_k(x, y)S_k(x, y) \end{pmatrix}. \quad (4.6)$$

The sums run from  $k = 0$  to  $K = -1$ , being  $K$  the number of interferograms.  $\hat{\phi}(x, y)$  and  $\hat{\psi}(x, y)$  are the real and imaginary parts of  $\hat{f}(x, y)$ , respectively.  $c_k(x, y)$  and  $s_k(x, y)$  are the real and imaginary parts of  $e^{i\beta_k(x, y)}$ , respectively.

Now, in the Eq. (3.7) we propose the local weighted least-squares cost function for estimating the wavefront variations  $\beta_k(x; y)$  for the  $(x, y)$  site at the  $k$ -interferogram; assuming that the wavefront under test  $\theta_0(x, y)$  is known.

$$E[a(x, y), g_k(x, y)] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \left[ \{a(m, n) + \text{Re}\{g_k(m, n)e^{i\theta_0(x, y)}\} - I_k(m, n)\} h(x-m, y-n) \right]^2 \quad (4.7)$$

In this equation,  $h(x, y)$  is an scalar field that weights the least-squares error of Eq. (3.7). By minimizing Eq. (3.7) with respect to  $a(x, y)$  and  $g_k(x, y)$ , the spatial wavefront variations  $\beta_k(x, y)$  are recovered as

$$\hat{\beta}_k(x, y) = \text{angle}[\hat{g}_k(x, y)], \quad (4.8)$$

being  $\hat{g}_k(x, y)$  the complex value that minimizes Eq. (3.7). The Eq. (3.7) can be seeing like the convolution of the least-squares error with the kernel  $h(x, y)$ . The kernel  $h(x, y)$  weights the local neighborhood around the site  $(x, y)$  being processed. In this way, we have a weighted least-squares cost function for each pixel  $(x, y)$  and we can estimate the phase-shifting variation  $\beta_k(x, y)$  for each pixel  $(x, y)$  of the  $k$ -interferogram. The linear equation

system that minimize Eq. (3.7), can be given in the following way:

$$\begin{pmatrix} [1s * h](x, y) & [\phi * h](x, y) & [\psi * h](x, y) \\ [\phi * h](x, y) & [\phi * h]^2(x, y) & [\phi\psi * h](x, y) \\ [\psi * h](x, y) & [\phi\psi * h](x, y) & [\psi * h]^2(x, y) \end{pmatrix} \begin{pmatrix} \hat{a}(x, y) \\ \hat{c}_k(x, y) \\ \hat{s}_k(x, y) \end{pmatrix} = \begin{pmatrix} [I_k * h](x, y) \\ [I_k \phi * h](x, y) \\ [I_k \psi * h](x, y) \end{pmatrix} \quad (4.9)$$

where  $*$  is the convolution operator and  $[*](x; y)$  means the convolution evaluated at  $(x, y)$ . The term  $1s$  is an scalar field of ones with dimension  $M \times N$ . In this case,  $\hat{\phi}(x, y)$  and  $\hat{\psi}(x, y)$  are the real and imaginary parts of  $e^{i\theta_0(x, y)}$ , respectively, and  $\hat{c}_k(x, y)$  and  $\hat{s}_k(x, y)$  are the real and imaginary parts of  $\hat{g}(x, y)$ , respectively.

Having the linear systems that estimate the wavefront under test  $\theta_0(x, y)$  and its variations  $\beta_k(x, y)$  for all pixels of all interferograms (Eqs. (3.6) and (3.9) respectively), we use them iteratively in the following way:

- 1 Set your convolution kernel  $h(x, y)$  that will weight the local neighborhood around the  $(x, y)$  sites. For example,  $h(x, y)$  can be a mean or Gaussian window.
- 2 Start the process by setting initial values for  $\beta_k(x, y), \forall(x, y), k = 0, 1, 2, \dots, K - 1$  being  $K$  the number of interferograms.
- 3 Solve the linear system of Eq. (3.6)  $\forall(x, y)$  and obtain  $\hat{\theta}_0(x, y)$  using Eq. (3.5).
- 4 For  $k = 0, 1, 2, \dots, K - 1$ , use the previous wavefront estimation  $\hat{\theta}_0(x, y)$  to solve the linear equation system of Eq. (3.9)  $\forall(x, y)$  and obtain  $\hat{\beta}_k(x, y)$  using Eq. (3.8).
- 5 With the previously estimated  $\hat{\beta}_k(x, y)$  of each interferogram, repeat subsequently the steps 3 and 4 until reach a convergence error.

As said in the step 5, this iterative process do its job until a convergence error is reached. This convergen error can be given aas the relative error between the values in a previous iteration and the values in the current intereration in the following way:

$$\varepsilon = \sum_{x,y} | \hat{\theta}_0^+(x, y) - \hat{\theta}_0^-(x, y) |, \quad (4.10)$$

where  $\hat{\theta}_0^+(x, y)$  is the phase obtained in the current iteration and  $\hat{\theta}_0^-(x, y)$  is the phase obtained in the previous iteration. At the end of this process,

$\hat{\theta}_0(x, y)$  and  $\hat{\beta}_k(x, y)$  will have the best estimations of the wavefront under test and its non-constant variations, respectively. The initial values in step 2 are important. By experimenting numerically with simulations, we found the following initial values as good starting points:  $\hat{\beta}_k(x, y) = \frac{\pi}{2}k, \forall(x, y)$ . We could use other value different to zero as starting point of the iterative process, however, starting at  $\pi/2$  is a good starting point since in practice phase-shifting interferometers are calibrated to introduce a phase shift of  $\pi/2$  radians. Certainly, starting at  $\pi/2$  is not always the best if we have a priori information about the phase shifts introduced. For example, if we know that the phase-shifting interferometer introduce  $\pi/4$  phase shift radians, then the best starting point is  $\pi/4$  radians.

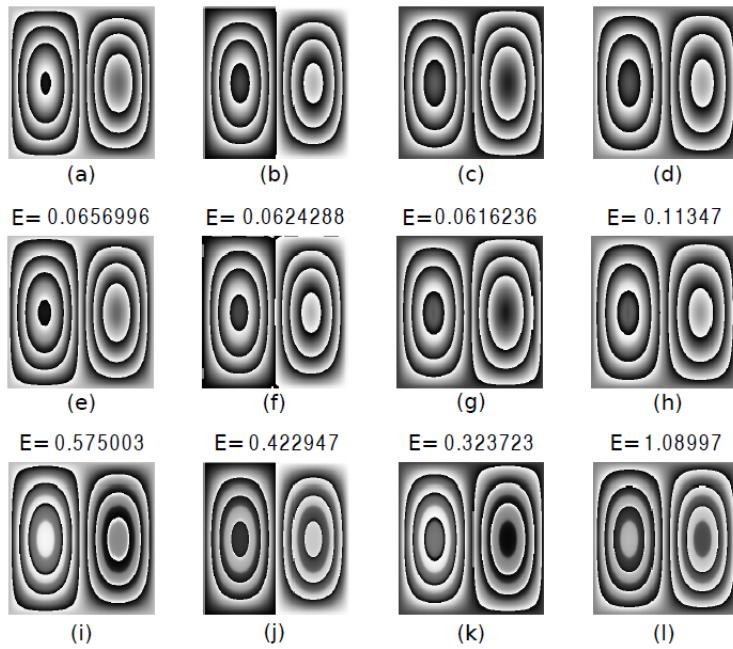


Figure 4.1: Numerical examples. From (a) to (b) we show the wrapped wavefronts generated with Eq. (11). From (e) to (h) we show the estimated wavefronts using the RAPS algorithm presented here. From (i) to (l) we show the estimated wavefronts using the AIA. Above each estimated wavefront we show the estimation error.

### 4.3 Tests and results

In the following test, we are going to quantify the wavefront estimation error of the RAPS demodulation method presented here. We are going to compare our results with the estimation error that is obtained with the Advanced Iterative Algorithm (AIA) presented in [46]. For this test, we simulated a moving wavefront and generated 4 interferograms with a phase-shifting of  $\pi/2$ . The moving wavefront was modeled using a two-modes vibrating plate in the following way:

$$\theta_k(x, y) = A \cos\left(\frac{2\pi}{256}x\right) \sin\left(\frac{2\pi}{256}y\right) \cos\left(\frac{\pi}{17}k\right) \quad (4.11)$$

The Figs. 3.1(a-d) shows the wrapped wavefronts of this simulated moving wavefront. Using the RAPS and the AIA method, we estimate the wavefront for each interferogram and the results are shown at Figs. 3.1(e-h) and 3.1(i-l) respectively. The global error of the phase estimation is shown above its wrapped phase image and it was calculated in the following way

$$\varepsilon = \sqrt{\frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} |\theta_k(x, y) - \hat{\theta}_k(x, y)|^2}, \quad (4.12)$$

where  $\theta_k(x, y)$  and  $\hat{\theta}_k(x, y)$  are the simulated and estimated wavefronts for the  $k$ -interferogram, respectively, and the dimension of interferograms is  $M \times N = 256 \times 256$ . For the RAPS method, the kernel  $h$  was a mean window of  $32 \times 32$ . By the estimated errors obtained in each phase estimation (shown above each image of the estimated phase maps of Fig. 3.1), we can see that the RAPS method reduces one order of magnitude the estimation error, compared with an standard iterative phase-shifting algorithm like the AIA. The computational time was 0.374 seconds for the AIA and 4.99 seconds for the RAPS method.

Now, we are going to test our method with interferograms experimentally obtained and compare qualitatively the results with the AIA method. For this test, the interferograms has a dimension of  $480 \times 640$  and the kernel that we used for the RAPS method was a mean window of  $132 \times 132$ . The interferogram sequence was generated by using the ESPI array like the shown in Fig. 3.2. The purpose here is to take four experimental phase-shifting interferograms of a moving wavefront. Here the mechanical properties of the metal plate are not analyzed. We introduce phase-shifts of  $\pi/2$  radians with the PZT and perturb the wavefront from the metal plate with a horn in such a way that it is moving while recording the phase-shifting interferogram

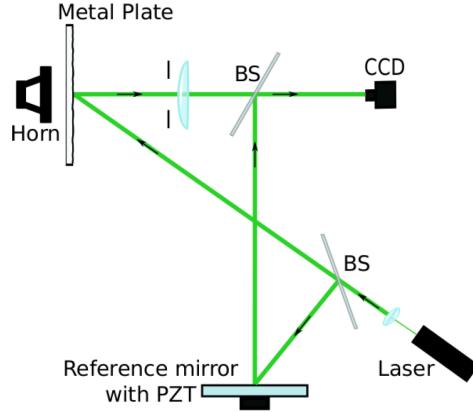


Figure 4.2: Experimental setup. Electronic Speckle Pattern Interferometry (ESPI) setup. The object under test is a metal plate perturbed with a horn. The reference mirror has attached a piezoelectric transducer (PZT) as phase shifter.

sequence. The estimation results are shown in Fig. 3.3. In Fig. 3.3(a-d) we show the phase-shifting interferogram sequence. In Fig. 3.3(e-h) we show the estimated wavefronts with the AIA method for each interferogram. In Fig. 3.3(i-l) we show the estimated wavefronts with the RAPS method proposed here. From the estimations shown in Fig. 3.3, it is hard to see any improvement of the demodulation method proposed here compared with the AIA. However, there is a hidden detuning error that introduce the AIA method since the wavefront under test was moving. This detuning error is augmented by differentiating (taking its partial derivatives) the wrapped phase as shown in Fig. 3.4. For a better appreciation of the detuning error, we quantized the dynamic range of the phase difference to the gray levels 1, 102, 153, 203 and 255. In Fig. 3.4(a) we show the difference of the wrapped phase obtained with the AIA for the first interferogram and the Fig. 3.4(b) shows the difference of the wrapped phase obtained with the RAPS for the first interferogram as well. In these results, we can see that the RAPS method presented here does not have a detuning error like the introduced with the AIA. The detuning error introduced with the AIA looks like a fringe pattern with twice the frequency than the original interferogram, this is known and demonstrated theoretically on previous published works [36, 39]. The computational time for these phase estimations was of 5.3 seconds for the AIA and 69.23 seconds for the RAPS.

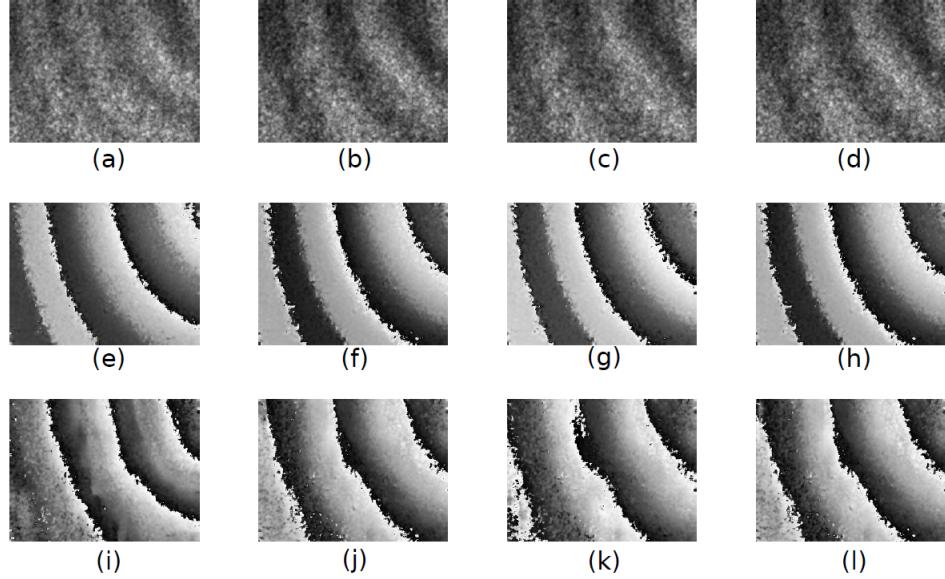


Figure 4.3: Experimental tests. From (a) to (d) we show the interferogram sequence. From (e) to (h) we show the estimated wavefront using the AIA, and from (i) to (l) we show the estimated wavefront using the RAPS.

#### 4.4 Discussion and commentaries

Accuracy of the wavefront estimation is the most important issue in optical tests. When using Phase Shifting Interferometry (PSI) techniques, the estimation accuracy depends on the stability of the object under test and the right calibration of the PSI setup. When the wavefront under test is moving, being by the environment perturbations or by its own nature, phase-shifting algorithms introduce an unavoidable detuning error. Previous to this work, all phase-shifting algorithms make the phase estimation by assuming an static wavefront and a global phase shifting. For example, the following very known four-steps phase-shifting algorithm:

$$\theta = \arctan \left( \frac{I_0 - I_2}{I_1 - I_3} \right). \quad (4.13)$$

This formula is the same for all pixels  $(x, y)$  because it is assumed that all pixels has the same phase shift of  $\pi/2$  radians, and the modulating phase is spatially static. However, this is not true when the wavefront under

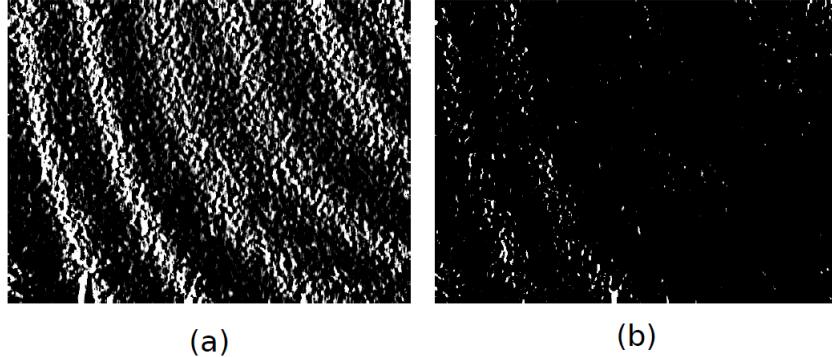


Figure 4.4: Phase difference with respect to  $x$  (partial approximated derivative). The dynamic range of the difference is quantized to the gray levels 1, 102, 153, 203 and 255, for a better appreciation of the detuning error. (a) shows the phase difference corresponding to the AIA and (b) shows the difference corresponding to RAPS method.

test is moving, in this case, each pixel should have its own phaseshifting formula/algorithm. When the wavefront under test is moving, even a self-tuning or self-calibrating method like the one presented in Refs.[47, 46, 7] introduce detuning errors. As shown in the results of this paper, the demodulation method presented here reduces considerably detuning errors by estimating the modulating phase and its non constant variations for each interferogram. These estimations are made by using iteratively the linear systems of Eqs. (3.6) and (3.9). The linear system of Eq. (3.6) looks very similar to the one used in Ref. [46] firstly proposed in Ref. [28], but, there is a big difference in the propose of this paper. The main difference is that the phase-shifting of Eq. (3.6) depends on the pixel  $(x, y)$  being processed and is not constant as supposed in [46]. Nevertheless, the most important contribution of the Robust Adaptive Phase-Shifting (RAPS) method presented here, is with the Eqs. (3.7) and (3.9). The Eq. (3.7) is a weighted least-squares cost function expressed as a quadratic convolution with a kernel that weights the local neighborhood that will be used for the phase-shifting estimation at the pixel  $(x, y)$  of the  $k$ -interferogram. In this way, we estimate the phase-shifting at each pixel of the  $k$ -interferogram and not a global phase-shifting as in Refs. [47, 46].

The kernel  $h$  in Eq. (3.7) defines the local weighted neighborhood that is taken into account by the least-squares function for each pixel  $(x, y)$ . For

example, suppose that you use a kernel with ones with the same dimension than the interferograms. If we take the convolution of the least-squares error with this kernel and evaluate it at the central pixel, we will have the classic global least-squares error used in phase-shifting. Then, the size of the kernel  $h$  affects directly the results of the demodulation method presented here. As greater it is, the results will be more comparable with the AIA method for example and it will be less tolerant with the movements of the wavefront under test. The convergence error of the RAPS does not depends on the  $h$  kernel size, but the results of the demodulated phase does. If the kernel size is too small, we do not obtain the demodulated phase as expected.

On the other hand, when the wavefront is actually not temporal static but dynamic, its model in a local neighborhood around a time  $t_0$  is the following:

$$\theta(x, y, t) = \theta(x, y, t_0) + \frac{\partial\theta(x, y, t_0)}{\partial t}(t - t_0). \quad (4.14)$$

In this case, the perturbation  $\eta_k(x, y)$  shown in Eq. (3.2) corresponds to the local temporal variation of the wavefront in that interferogram, that is,  $\eta_k(x, y) = \frac{\partial\theta(x, y, k\Delta t)}{\partial t}k\Delta t$ , being  $\Delta t$  the discretization of the temporal  $t$  variable. As a consequence, this method can be used for testing dynamic events, regarding that the phase shifting introduced is greater than the maximum  $\eta_k(x, y), \forall k$ . Actually, the simulated wavefront given in Eq. (3.11) is a dynamic two-modes vibrating plate. We generated the interferograms with that wavefront model and introduced a phase-shifting of  $\pi/2$  radians. As the reader can see in the first test, we have reduced considerably the estimations errors when the wavefront under test is moving.

Summing up, here we have presented a Robust Adaptive Phase-Shifting (RAPS) algorithm that estimates the wavefront under test regardless its movements by environment perturbations. However, this method can be used for testing dynamic events using PSI techniques and with an appropriate phase-shifting. This method estimates the wavefront under test and its non-constant variations given in each interferogram. Compared with a traditional self-tuning phase-shifting algorithm, this method makes better estimations by reducing considerably the detuning errors when the wavefront under test is moving.

# **Chapter 5**

## **Removing detuning distortions of wrapped phase by using robust quadrature filters**

### **5.1 Introduction**

In optical interferometry tests, the modulating phase has the information of interest. Therefore, estimating the phase with the least possible error is the most important task in interferometry. However, this is not always possible, and one of the most common errors of phase interferometry demodulation algorithms (PIDA) is the so-called detuning error [4, 38, 30, 23]. In literature, we can find several works that study and measure the effects of the detuning error and the factors that cause it [13, 43, 4, 29, 24, 44, 5, 35, 10, 37]. For example, in phase shifting interferometry, if the object under test is not static or the piezoelectric transducer used to introduce the phase shifts is incorrectly calibrated, the PIDA produces a detuning error. Another example where this detuning error commonly occurs is in dynamic interferometry. In dynamic interferometry, detuning error occurs when we do not have fast enough cameras or high repetition lasers with long coherence lengths, or a combination of both. In the Fourier method, the detuning error is present when the filtering process can not be performed properly, since the carrier is not enough to separate the complex signal in the Fourier domain. Therefore, in general, when the PIDA does not receive the interferogram or interferograms as expected, it recovers a modulating phase with detuning

error.

This detuning error is present in the demodulated phase as a low powered signal having twice the frequency than the interferograms under analysis. Mosiño et al. [30] demonstrate that this detuning error can be described by the following expression:

$$\Delta\phi \approx -\frac{\varepsilon}{c} \sin(2\phi), \quad (5.1)$$

where  $\Delta\phi$  is the difference between the desired phase  $\phi$  and the erroneous spurious phase  $\phi^\varepsilon$ ,  $\varepsilon$  is the erroneous spurious signal that our PIDA does not remove properly, and  $c$  is the complex desired signal we want to recover. This detuning effect is graphically shown in the unwrapped phase of Fig. 5.3(a). As we see, the distorted phase has a double frequency component mounted, while the desired unwrapped phase has to look like the one in Fig. 5.3(b). Therefore, removing the detuning error from the wrapped phase without affecting the information is not a trivial problem, and, as far as we know, there is no published work on processing the distorted wrapped phase in order to reduce this detuning error.

Hence, in this paper, we are going to show how we can reduce this detuning error from the demodulated wrapped phase by using a variant of the Robust Quadrature Filters (RQF) [25]. The variant that we implement here is such that the RQF use the gradient of the distorted demodulated wrapped phase as *a priori* local frequency information. As we work with a wrapped phase, the data term of the RQF compares the complex signal of this wrapped phase with the signal that we are expecting to obtain; that is, the one without distortions. To show the performance of the **phase detuning correction method (PDCM)** presented here, we will present test and results from simulated and experimentally obtained data.

## 5.2 Phase detuning correction method (PDCM)

As we said above, our PDCM uses a specially tuned RQF system to process a complex signal generated by the given demodulated wrapped phase. The energy function of this specially tuned RQF is the following:

$$\begin{aligned} U(f) = & \sum_{x,y \in L} |f_{x,y} - 2g_{x,y}|^2 + \lambda \sum_{x,y \in L} |f_{x,y} - f_{x-1,y} \exp(u_{x,y})|^2 \\ & + \lambda \sum_{x,y \in L} |f_{x,y} - f_{x,y-1} \exp(v_{x,y})|^2, \end{aligned} \quad (5.2)$$

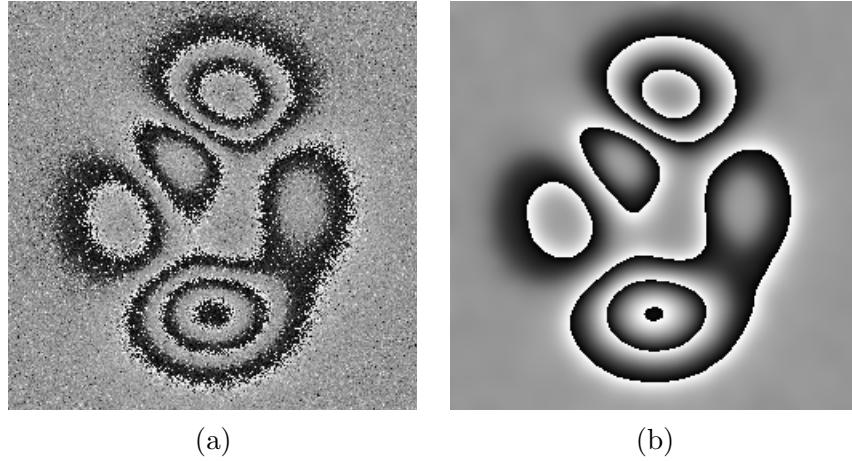


Figure 5.1: Simulated wrapped phase comparison. (a) Synthetic input wrapped phase with detuning error. (b) Recovered wrapped phase map using the proposed PDCM.

where  $g_{x,y} = \exp(i\phi_{x,y}^\varepsilon)$  is the complex signal generated by the demodulated wrapped phase  $\phi_{x,y}^\varepsilon$  that we are going to process,  $f_{x,y}$  is the independent complex value that minimizes Eq. (5.2) (and hence, the output of our PDCM),  $\lambda$  is the regularization parameter that controls the bandwidth of the quadrature filter, and  $u_{x,y}$  and  $v_{x,y}$  are the a priori local frequency information obtained from the wrapped phase; that is,  $u_{x,y}$  and  $v_{x,y}$  are the  $x$  and  $y$  components of the gradient of  $\phi_{x,y}^\varepsilon$ . Below, we will show how this prior information is obtained from the wrapped phase  $\phi_{x,y}^\varepsilon$ . The sums run over all valid sites  $x, y$  in  $L$ , being  $L$  the lattice domain of the demodulated wrapped phase. To minimize Eq. (5.2), one could use any method for solving systems of linear equations; in particular, we use the *Gauss-Seidel* method. Having the complex field  $\hat{f}$  that minimizes Eq. (5.2), the refined demodulated wrapped phase is obtained as

$$\hat{\phi}_{x,y} = \angle[\hat{f}_{x,y}]; \quad (5.3)$$

that is, as the angle of the complex signal  $\hat{f}$  obtained by the minimization process.

The new wrapped phase  $\hat{\phi}$ , estimated with Eq. (5.3), is an improved version of our input wrapped phase  $\phi^\varepsilon$ ; i.e., the detuning error has been lessened. If we use this process recursively, we can refine the wrapped phase until we obtain an almost error free enhanced wrapped phase. Using this

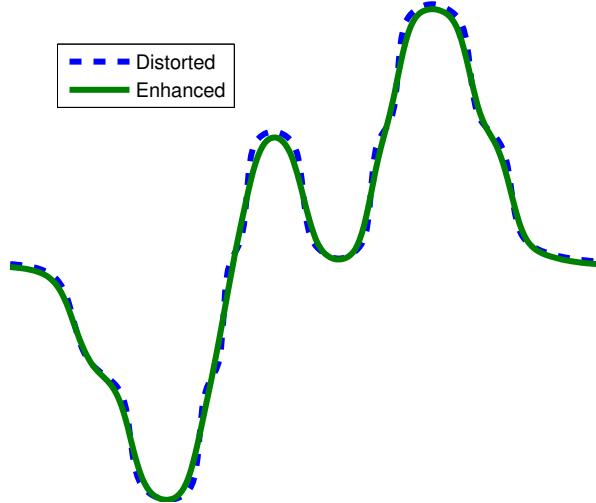


Figure 5.2: Comparison between an unwrapped row of the distorted input and the enhanced phase; the dashed line is the unwrapped phase with detuning error, and the solid line is the unwrapped phase processed with the PDCM. Note: The noise had been removed from the original distorted wrapped phase in Fig. 5.1(a) for clarity and comparison purposes.

approach, we ensure that the new calculated phase is almost free of detuning errors. Then, this recursive process can be given as:

- 1.- Estimate local frequencies  $u$  and  $v$  taking the wrapped gradient of the input wrapped phase  $\phi^\varepsilon$  [Eqs. (5.6) and (5.7)].
- 2.- Take  $\hat{f}$  that minimizes Eq. (5.2) using  $u$ ,  $v$  and  $g = \exp(i\phi^\varepsilon)$  as data input.
- 3.- Calculate an enhanced  $\phi$  using  $\hat{f}$  with Eq. (5.3).
- 4.- Recalculate local frequencies  $u$  and  $v$  with the new wrapped phase  $\phi$ .
- 5.- Repeat steps 2 to 4 until a desired convergence point.

This convergence point can be given as the relative error between the former and the new wrapped phases in the following way:

$$\varepsilon = \sum_{x,y} |\phi_{x,y}^+ - \phi_{x,y}^-|, \quad (5.4)$$

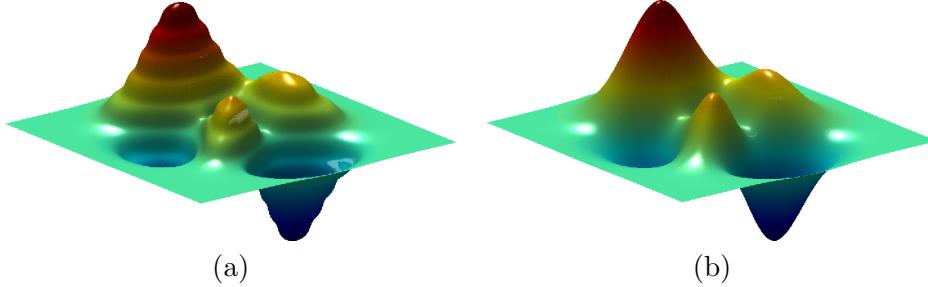


Figure 5.3: Unwrapped simulated phase comparison of Fig. 5.1. (a) Shows the unwrapped phase with detuning error of Fig. 5.1(a). (b) Shows the recovered unwrapped phase of Fig. 5.1(b) after using the proposed PDCM. Note: The noise had been removed from the original distorted wrapped phase in Fig. 5.1(a) for clarity and comparison purposes.

where  $\phi_{x,y}^+$  and  $\phi_{x,y}^-$  are the wrapped phases obtained in the current and previous iterations, respectively. At the end of this iterative process, the new wrapped phase  $\phi$  will be a wrapped phase without detuning distortions and without noise.

### 5.3 A priori local frequency calculation

Since in order to calculate the local frequencies we have a wrapped phase, we can not use finite differences directly due to the  $2\pi$  phase jumps of the wrapped phase. To estimate the local frequencies correctly, we propose taking the gradient in an explicit way from the complex field as follows:

$$\nabla \phi_{x,y} = \nabla \left[ \arctan \left( \frac{\sin \phi_{x,y}}{\cos \phi_{x,y}} \right) \right]. \quad (5.5)$$

Therefore, differentiating Eq. (5.5) and simplifying, we obtain the following mathematical expression for local frequencies  $u_{x,y}$  as

$$u_{x,y} = \frac{\sin \phi \frac{\partial}{\partial x} \cos \phi - \cos \phi \frac{\partial}{\partial x} \sin \phi}{\cos^2 \phi + \sin^2 \phi}, \quad (5.6)$$

and in a similar way, the local frequencies  $v_{x,y}$  are estimated as follows

$$v_{x,y} = \frac{\sin \phi \frac{\partial}{\partial y} \cos \phi - \cos \phi \frac{\partial}{\partial y} \sin \phi}{\cos^2 \phi + \sin^2 \phi}. \quad (5.7)$$

The spatial dependence of the input wrapped phase map was omitted for clarity in the notation. The partial differentials  $\frac{\partial}{\partial x}\phi_{x,y}$  and  $\frac{\partial}{\partial y}\phi_{x,y}$  of Eqs. (5.6) and (5.7) are calculated as finite differences as follows

$$\frac{\partial}{\partial x} \cos \phi_{x,y} = \cos \phi_{x,y} - \cos \phi_{x+1,y}, \quad (5.8)$$

$$\frac{\partial}{\partial y} \sin \phi_{x,y} = \sin \phi_{x,y} - \sin \phi_{x,y+1}. \quad (5.9)$$

It is important to note that on the borders, it is not possible to calculate the frequencies as Eqs. (5.8) and (5.9) propose; for this reason, we propose using the value of a correctly calculated adjacent local frequency.

## 5.4 Tests and experimental examples

To verify the efficiency of the PDCM, we simulated four interferograms of  $256 \times 256$  pixels with phase shifts of  $\pi/5$ , and demodulated them with a 4-step algorithm tuned at  $\pi/2$ . In Fig. 5.1(a), we can see the distorted wrapped phase map obtained by the PIDA. The phase was simulated using the peaks function of Matlab. Using this distorted wrapped phase map and the proposed PDCM, we obtain the enhanced wrapped phase map displayed in Fig. 5.1(b). As it can be seen, the detuning distortions have been notably reduced as well as noise. To obtain the result in Fig. 5.1(b), the regularized parameter  $\lambda$  of the energy function of Eq. (5.2) was  $\lambda = 5$ , and the number of iterations of the algorithm described above was 10. The computational time for this phase estimation was 0.3932 seconds, in a PC with an Intel Core i7 processor and 8 GB RAM memory. To minimize Eq. (5.2), we used the Gauss-Seidel method. For clarity and comparison purposes, in Fig. 5.2 we plot the central row of both unwrapped phases without noise; the dashed curve is the erroneous phase, while in the solid curve, we have the corrected phase map. This graph lets us see that the detuning error has been fairly lessened. For best viewing, Fig. 5.3 shows both unwrapped phases. Fig. 5.3(a) is the unwrapped phase with the detuning error shown in Fig. 5.1(a), while Fig. 5.3(b) shows the unwrapped phase obtained using the proposed PDCM. For clarity and comparison purposes of the detuning distortion, the noise was removed from the unwrapped phase shown in Fig. 5.3(a).

Finally, we show the performance in an experimental wrapped phase map. This experimental wrapped phase map corresponds to a vibration mode of a latex membrane excited with a horn at 1.6 kHz and with an amplitude of 1.8 V [Fig. 5.4(a)]. Given that the object was vibrating during

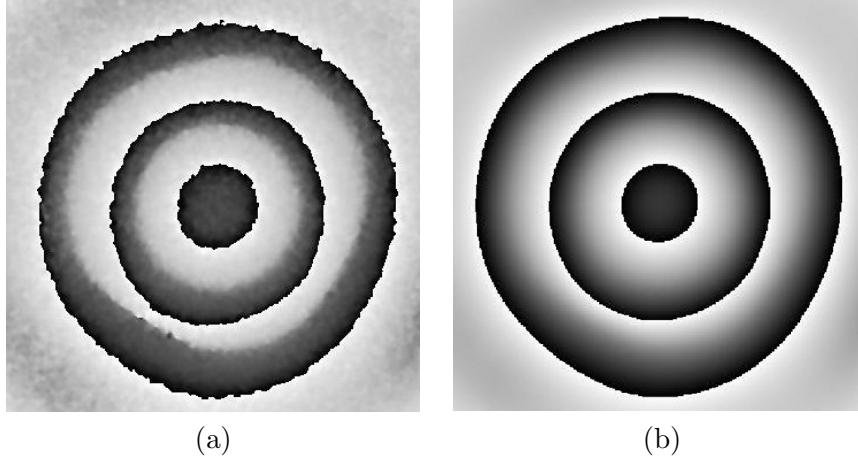


Figure 5.4: Experimental wrapped phase comparison. (a) Shows the experimental input wrapped phase with detuning error. (b) Shows the recovered wrapped phase map using the proposed PDCM.

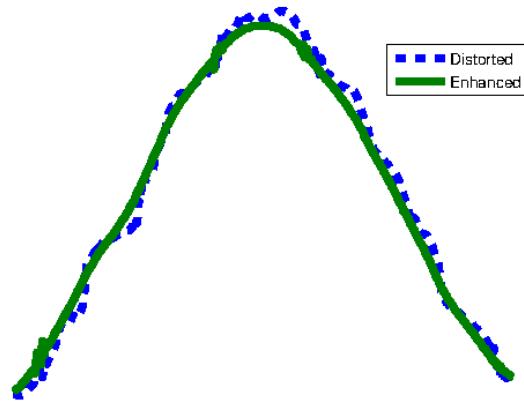


Figure 5.5: Comparison between an unwrapped row of the distorted input and the enhanced phase; the dashed line is the phase with detuning error, and the solid line is the phase processed by the PDCM.

the capture, it was not possible to correctly tune up the phase shift steps to  $\pi/2$ ; as a consequence, the four step algorithm obtains a distorted wrapped phase map caused by the detuning. This distorted wrapped phase map

is shown in Fig. 5.4(a). In Fig. 5.4(b), we show the performance of the proposed PDCM. The processing time for this phase estimation was 1.3447 seconds using the same PC described above. As well as shown in the simulations, in Fig. 5.5 we can see the central row of both unwrapped phases. As it is evident, the noise was removed and detuning distortions have almost disappeared.

## 5.5 Discussion and commentaries

Accuracy of the wave-front estimation is the most important issue in optical tests. When using phase interferometry techniques, estimation accuracy depends on the stability of the object under test and the right calibration of the phase interferometry set-up. When the wave-front under test is moving, be it because of environmental perturbations or its own nature, and / or the phase interferometry set-up is not properly calibrated, PIDA introduce an unavoidable detuning error. Previous to this work, as far as we know, all PIDA have done their best to avoid this detuning error by means of applying different strategies; instead of that, here we present a method to process the distorted wrapped phase in order to reduce the detuning distortions introduced by the PIDA.

As shown in the results of this paper, the presented PDCM considerably removes detuning errors using the gradient of the distorted wrapped phase as a priori information about the local frequencies, generating a complex signal with the demodulated wrapped phase, and processing it with a variant of the RQF to obtain a more refined demodulated wrapped phase. This process of refinement is carried out using the linear system of Eq. (5.2) iteratively. The proposed system looks very similar to the one used in [25], but there are differences in the proposal of this paper. Instead of using a global carrier for the entire image as in [25], here, each pixel has its own frequency, in a way that the filter is locally tuned. The functional in [25] is non-linear, while the proposed functional [Eq. (5.2)] is a linear system. Another difference is that the proposed functional is constructed using the demodulated wrapped phase and its local frequencies as a priori information. These two differences result in a stable method that converges in a few iterations.

Note that using Eqs. (5.6) and (5.7), it is not necessary to unwrap the phase map to calculate the local frequencies  $u$  and  $v$ . This way of calculating the local frequencies on each iteration speeds up the global process.

Summing up, we have presented a method capable of reducing the detuning distortions of the demodulated wrapped phase from the PIDA. This

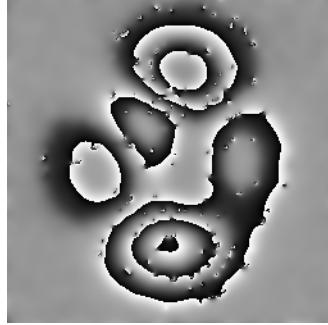


Figure 5.6: Recovered wrapped phase using the proposed PDCM with miscalculated frequencies.

method is an iterative process that uses the distorted wrapped phase and its local frequencies as a priori information. This method is stable and converges after a few iterations. The main contribution of the presented method is that it is capable of removing detuning distortions without eliminating wrapped phase information, which is very hard to do with simple low-pass filters. As it is evident in the results, the proposed method obtains an smooth wrapped phase map with very attenuated detuning distortions. The practical use of this method is that it allows one to avoid complex PIDA in order to obtain a wrapped phase fairly free of detuning distortions, or even avoid the need to recalibrate and repeat the optical set-up.

## 5.6 Appendix. Local frequency calculation in presence of noise

In presence of noise in our input wrapped phase, the local frequencies could be erroneously calculated given that Eqs. 5.6 and 5.7 are essentially high pass filters. For example, Fig. 5.6 is the output of our proposed PDCM when the local frequencies are miscalculated. As we can notice, the detuning distortion and the noise were removed, but the recovered wrapped phase has certain spurious errors product of the miscalculated local frequencies. To avoid this problem, we propose applying a low-pass filter to our noisy input wrapped phase. It is important to say that if we apply the low-pass filter directly to our wrapped phase, we will lose fringe phase information. For

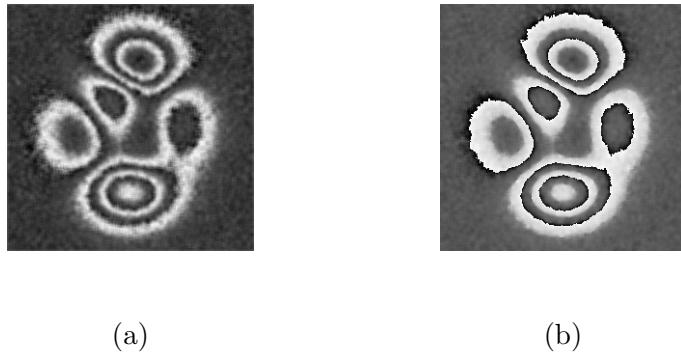


Figure 5.7: Wrapped phase comparison after applying a low-pass filter to the noisy wrapped phase in Fig. 5.1(a). (a) Shows the result of applying a gaussian filter directly to our wrapped phase, while in (b) we see the result of applying Eq. 5.10.

this reason, we propose filtering the wrapped phase as follows:

$$\hat{\phi} = \arctan \left[ \frac{\sin(\phi^\eta) * h}{\cos(\phi^\eta) * h} \right], \quad (5.10)$$

where  $*$  is the convolution operator,  $h$  is a low-pass convolution kernel,  $\phi^\eta$  is the noisy wrapped phase, and  $\hat{\phi}$  is the filtered wrapped phase, both with detuning distortions; this way of doing the filtering allows us to keep the  $2\pi$  jumps in our wrapped phase map. In Fig. 5.7 we show the difference between these two approaches. Fig. 5.7(a) is the result of applying the low-pass filter directly to our noisy wrapped phase in Fig. 5.1(a), while Fig. 5.7(b) is the outcome of filtering using Eq. 5.10. As it can be seen, using the proposed approach we are able to maintain the  $2\pi$  jumps, and therefore, the fringe phase information.

It is important to note that this process does not affect the phase information, given that the enhanced wrapped phase is calculated using the original noisy wrapped phase map. Also and more important, the local frequencies are a signal with very small variations, whereas the noise signal is the opposite.

After having removed the noise from the noisy wrapped phase, the local frequencies can be correctly calculated. To illustrate, Fig. 5.6 shows a sequence of the calculation of three different local frequencies during the wrapped phase refinement process.

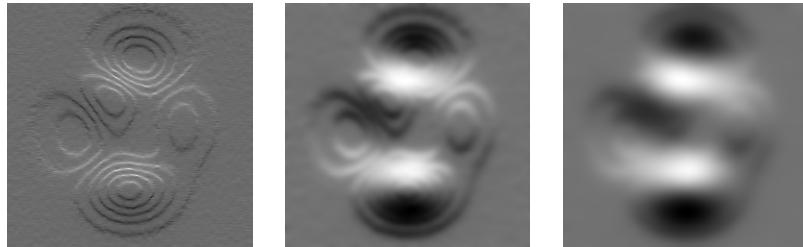


Figure 5.8: Example of the frequency change along the refinement process of the distorted wrapped phase. For this sequence, we show the  $u(x, y)$  frequency calculation using Eq. 5.6.

## Acknowledgements

The authors would like to acknowledge the financial support of the Mexican Science Council CONACYT.



# Bibliography

- [1] Chiayu Ai and James C. Wyant. Effect of piezoelectric transducer nonlinearity on phase shift interferometry. *Appl. Opt.*, 26(6):1112–1116, Mar 1987.
- [2] J. H. Bruning, D. R. Herriott, J. E. Gallagher, D. P. Rosenfeld, A. D. White, and D. J. Brangaccio. Digital wavefront measuring interferometer for testing optical surfaces and lenses. *Appl. Opt.*, 13(11):2693–2703, Nov 1974.
- [3] Yeou-Yen Cheng and James C. Wyant. Phase shifter calibration in phase-shifting interferometry. *Appl. Opt.*, 24(18):3049–3052, Sep 1985.
- [4] M. Servin D. Malacara and Z. Malacara. *Interferogram Analysis for Optical Testing*. Taylor & Francis, 2005.
- [5] Peter de Groot. Phase-shift calibration errors in interferometers with spherical fizeau cavities. *Appl. Opt.*, 34(16):2856–2863, Jun 1995.
- [6] J. C. Estrada, M. Servin, and J. A. Quiroga. Easy and straightforward construction of wideband phase-shifting algorithms for interferometry. *Opt. Lett.*, 34(4):413–415, Feb 2009.
- [7] Julio C. Estrada, Manuel Servin, and Juan A. Quiroga. A self-tuning phase-shifting algorithm for interferometry. *Opt. Express*, 18(3):2632–2638, Feb 2010.
- [8] Klaus Freischlad and Chris L. Koliopoulos. Fourier description of digital phase-measuring interferometry. *J. Opt. Soc. Am. A*, 7(4):542–551, Apr 1990.
- [9] J. E. Greivenkamp. Generalized data reduction for heterodyne interferometry. *Optical Engineering*, 23(4):234350–234350–, 1984.

- [10] Bernd Gutmann and Herbert Weber. Phase-shifter calibration and error detection in phase-shifting applications: a new method. *Appl. Opt.*, 37(32):7624–7631, Nov 1998.
- [11] Pascal Haible, Mahenda P. Kothiyal, and Hans J. Tiziani. Heterodyne temporal speckle-pattern interferometry. *Appl. Opt.*, 39(1):114–117, Jan 2000.
- [12] P. Hariharan, B. F. Oreb, and T. Eiju. Digital phase-shifting interferometry: a simple error-compensating phase calculation algorithm. *Appl. Opt.*, 26(13):2504–2506, Jul 1987.
- [13] J. E. Hernández and D. Malacara. Exact linear detuning error in phase shifting algorithms. *Opt. comm.*, 180:9–14, 2000.
- [14] B. Jahne. *Digital Image Processing*. Springer, 2005.
- [15] A. Sato K. Okada and J. Tsujiuchi. Simultaneous calculation of phase distribution and scanning phase shift in phase shifting interferometry. *Opt. Commun.*, 84:118–124, 1991.
- [16] Bradley T. Kimbrough. Pixelated mask spatial carrier phase shifting interferometry algorithms and associated errors. *Appl. Opt.*, 45(19):4554–4562, Jul 2006.
- [17] In-Bok Kong and Seung-Woo Kim. General algorithm of phase-shifting interferometry by iterative least-squares fitting. *Optical Engineering*, 34(1):183–188, 1995.
- [18] Guanming Lai and Toyohiko Yatagai. Generalized phase-shifting interferometry. *J. Opt. Soc. Am. A*, 8(5):822–827, May 1991.
- [19] Rajesh Langoju, Abhijit Patil, and Pramod Rastogi. Phase-shifting interferometry in the presence of nonlinear phase steps, harmonics, and noise. *Opt. Lett.*, 31(8):1058–1060, Apr 2006.
- [20] Kieran Larkin. A self-calibrating phase-shifting algorithm based on the natural demodulation of two-dimensional fringe patterns. *Opt. Express*, 9(5):236–253, Aug 2001.
- [21] Kieran G. Larkin, Donald J. Bone, and Michael A. Oldfield. Natural demodulation of two-dimensional fringe patterns. i. general background of the spiral phase quadrature transform. *J. Opt. Soc. Am. A*, 18(8):1862–1870, Aug 2001.

- [22] Ricardo Legarda-Saenz and Mariano Rivera. Fast half-quadratic regularized phase tracking for nonnormalized fringe patterns. *J. Opt. Soc. Am. A*, 23(11):2724–2731, Nov 2006.
- [23] J. Antonio Quiroga M. Servin and M. Padilla. *Fringe Pattern Analisys for Optical Metrology: Theory, Algorithms and Applications*. Wiley-VCH, 2014.
- [24] D. Malacara. *Optical Shop Testing*. John Wiley & Sons, Inc., Hoboken, 2007.
- [25] J. L. Marroquin, J. E. Figueroa, and M. Servin. Robust quadrature filters. *J. Opt. Soc. Am. A*, 14(4):779–791, Apr 1997.
- [26] J. L. Marroquin, M. Servin, and R. Rodriguez-Vera. Adaptive quadrature filters and the recovery of phase from fringe pattern images. *J. Opt. Soc. Am. A*, 14(8):1742–1753, Aug 1997.
- [27] J. L. Marroquin, M. Servin, and R. Rodriguez Vera. Adaptive quadrature filters for multiple phase-stepping images. *Opt. Lett.*, 23(4):238–240, Feb 1998.
- [28] C. J. Morgan. Least-squares estimation in phase-measurement interferometry. *Opt. Lett.*, 7(8):368–370, Aug 1982.
- [29] J. F. Mosi no, D. Malacara Doblado, and D. Malacara Hernández. Calculus of exact detuning phase shift error in temporal phase shifting algorithms. *Opt. Express*, 17(18):15766–15771, Aug 2009.
- [30] J. F. Mosi no, M. Servin, J. C. Estrada, and J. A. Quiroga. Phasorial analysis of detuning error in temporal phase shifting algorithms. *Opt. Express*, 17(7):5618–5623, Mar 2009.
- [31] J Nocedal and S. JW. *Numerical Optimization, Operations Research*. Springer, 2006.
- [32] Julio C. Estrada Orlando Medina and Manuel Servin. Regularized self-tuning phase demodulation for phase-shifting interferometry with arbitrary phase shifts. *Proc. SPIE 8493, Interferometry XVI*, Sep 2012.
- [33] Mariano Rivera, Rocky Bizuet, Amalia Martinez, and Juan A. Rayas. Half-quadratic cost function for computing arbitrary phase shifts and phase: Adaptive out of step phase shifting. *Opt. Express*, 14(8):3204–3213, Apr 2006.

- [34] Pablo D. Ruiz, Jonathan M. Huntley, and Guillermo H. Kaufmann. Adaptive phase-shifting algorithm for temporal phase evaluation. *J. Opt. Soc. Am. A*, 20(2):325–332, Feb 2003.
- [35] Joanna Schmit and Katherine Creath. Extended averaging technique for derivation of error-compensating algorithms in phase-shifting interferometry. *Appl. Opt.*, 34(19):3610–3619, Jul 1995.
- [36] J. Schwider, R. Burow, K.-E. Elssner, J. Grzanna, R. Spolaczyk, and K. Merkel. Digital wave-front measuring interferometry: some systematic error sources. *Appl. Opt.*, 22(21):3421–3432, Nov 1983.
- [37] M. Servin, J. C. Estrada, and O. Medina. Fourier transform demodulation of pixelated phase-masked interferograms. *Opt. Express*, 18(15):16090–16095, Jul 2010.
- [38] M. Servin, J. C. Estrada, and J. A. Quiroga. The general theory of phase shifting algorithms. *Opt. Express*, 17(24):21867–21881, Nov 2009.
- [39] M. Servin, J. C. Estrada, J. A. Quiroga, J. F. Mosino, and M. Cywiak. Noise in phase shifting interferometry. *Opt. Express*, 17(11):8789–8794, May 2009.
- [40] M. Servin, J. L. Marroquin, and F. J. Cuevas. Fringe-follower regularized phase tracker for demodulation of closed-fringe interferograms. *J. Opt. Soc. Am. A*, 18(3):689–695, Mar 2001.
- [41] Manuel Servin, Jose Luis Marroquin, and Juan Antonio Quiroga. Regularized quadrature and phase tracking from a single closed-fringe interferogram. *J. Opt. Soc. Am. A*, 21(3):411–419, Mar 2004.
- [42] Y. Surrel. Phase stepping: a new self-calibrating algorithm. *Appl. Opt.*, 32(19):3598–3600, Jul 1993.
- [43] Brian J . Thompson and Daniel Malacara. *Handbook of Optical Engineering*. CRC Press, 2001.
- [44] Johannes van Wingerden, Hans J. Frankena, and Cornelis Smorenburg. Linear approximation for measurement errors in phase shifting interferometry. *Appl. Opt.*, 30(19):2718–2729, Jul 1991.
- [45] J. Vargas, J. Antonio Quiroga, C. O. S. Sorzano, J. C. Estrada, and J. M. Carazo. Two-step interferometry by a regularized optical flow algorithm. *Opt. Lett.*, 36(17):3485–3487, Sep 2011.

- 
- [46] Zhaoyang Wang and Bongtae Han. Advanced iterative algorithm for phase extraction of randomly phase-shiftedinterferograms. *Opt. Lett.*, 29(14):1671–1673, Jul 2004.
  - [47] X. F. Xu, L. Z. Cai, X. F. Meng, G. Y. Dong, and X. X. Shen. Fast blind extraction of arbitrary unknown phase shifts by an iterative tangent approach in generalized phase-shifting interferometry. *Opt. Lett.*, 31(13):1966–1968, Jul 2006.
  - [48] Fa Zeng, Qiaofeng Tan, Huarong Gu, and Guofan Jin. Phase extraction from interferograms with unknown tilt phase shifts based on a regularized optical flow method. *Opt. Express*, 21(14):17234–17248, Jul 2013.