# โครงการ

## การเรียนการสอนเพื่อเสริมประสบการณ์

ชื่อโครงการ     การถ่ายภาพมุมกว้างแนวระนาบเชิงแสงอาพันธ์ด้วยวิธีอนุพันธ์กึ่งกลาง
**Full-Field Optical Coherence Tomography with Central Derivative-Based Algorithm**

ชื่อนิสิต     นายณัฐจักษ์    พลเสน     5833416023

ภาควิชา     ฟิสิกส์

ปีการศึกษา     2561

คณะวิทยาศาสตร์    จุฬาลงกรณ์มหาวิทยาลัย

# Full-Field Optical Coherence Tomography with Central Derivative-Based Algorithm.

Natthajuks Pholsen

*A senior project submitted in fulfillment of the requirements*

*for the degree of Bachelor of Science*

*in the*

May 13, 2019.

Project Title : Full-Field Optical Coherence Tomography with Central
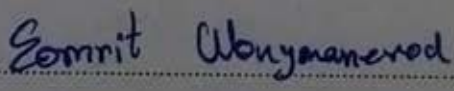Derivative-Based Algorithm.

Author : Natthajuks Pholsen

Supervisor : Asst.Prof.Dr. Montian Tianprateep

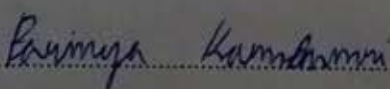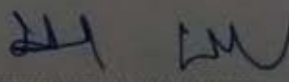Field of Study : Physics

Academic Year : 2018

---

accepted by the Department of Physics, Faculty of Science, Chulalongkorn
University, in fulfillment for the degree of Bachelor of Science.

Project committees

..... Somrit Wongmanerod ..... Chairman

(Dr. Somrit Wongmanerod)

..... Parinya Karndumri ..... Committee

(Assoc.Prof.Dr. Parinya Karndumri)

..... Advisor

(Asst.Prof.Dr. Montian Tianprateep)

จุฬาลงกรณ์มหาวิทยาลัย

# บทคัดย่อ

ภาควิชาฟิสิกส์
คณะวิทยาศาสตร์

วิทยาศาสตร์บัณฑิต

**การถ่ายภาพมุมกว้างแนวระนาบเชิงแสงอาพันธ์ด้วยวิธีอนุพันธ์กึ่งกลาง**

ณัฐจักษ์ พลเสน

การถ่ายภาพมุมกว้างแนวระนาบเชิงแสงอาพันธ์เป็นการถ่ายภาพที่ใช้แสงอาพันธ์ต่ำเพื่อศึกษาข้อมูลบนพื้นผิวซึ่ง
ถูกปิดทับ โดยวิธีดั้งเดิมที่นิยมใช้เพื่อประมวลผลรูปภาพในการถ่ายภาพมุมกว้าง คือ วิธีเลื่อนเฟส ๕ ขั้น และวิธีอนุพันธ์
ถอยหลัง แต่วิธีเหล่านี้ยังต้องได้รับการพัฒนา โครงงานนี้เน้นศึกษาวิธีอนุพันธ์กึ่งกลางสำหรับการถ่ายภาพมุมกว้างแนว
ระนาบเชิงแสงอาพันธ์ วิธีอนุพันธ์กึ่งกลางจะถูกใช้เพื่อประมวลภาพจากการจำลองและการทดลองกับพื้นผิวจริงเพื่อเปรียบ
เทียบกับวิธีเลื่อนเฟส ๕ ขั้น และวิธีอนุพันธ์ถอยหลัง ผลจากการศึกษาพบว่า ถึงแม้ว่าวิธีอนุพันธ์กึ่งกลางจะใช้เวลาในการ
ประมวลผลมากกว่า แต่มีข้อดีในการคงความสว่างของภาพ โดยเฉพาะอย่างยิ่งในกรณีการถ่ายภาพที่ระยะห่างของการ
ถ่ายมีขนาดเล็ก
คำสำคัญ  การถ่ายภาพมุมกว้างแนวระนาบเชิงแสงอาพันธ์, วิธีอนุพันธ์กึ่งกลาง, วิธีอนุพันธ์ถอยหลัง, วิธีเลื่อนเฟส ๕ ขั้น,
           แสงอาพันธ์ต่ำ

CHULALONGKORN UNIVERSITY

# *Abstract*

Faculty of Science

Department of Physics

Bachelor of Science

**Full-Field Optical Coherence Tomography with Central Derivative-Based Algorithm**

Natthajuks Pholsen

Full-field optical coherence tomography (FFOCT) takes the advantage of a low coherence light for revealing an information on a coated surface. The traditional algorithms such as 5-step phase-shift and backward derivative-based algorithms are usually applied for unveiling the images in this method, but improvement is still needed. In this senior project, a central derivative-based (CDB) algorithm for FFOCT is investigated. It is applied in both a simulation and experimental surfaces for comparing the method with the traditional 5-step phase-shift and backward derivative-based algorithms. The result shows that although the CDB-algorithm requires more time to process, it is good at retaining the brightness of images especially at short steps.

Keywords: *full-field optical coherence tomography, central derivative-based algorithm, backward derivative-based algorithm, 5-step phase-shift algorithm, low coherence light*

# *Acknowledgements*

# Contents

# List of Figures

# List of Tables

**Chapter 1**

# Introduction

## 1.1 Background and Motivation

Optical coherence tomography (OCT) was developed for cross-sectional imaging of biological objects in 1991 by Fujimoto et al. [1] This technique normally uses an interferometer with a low coherence light source, such as, white light and superluminescent diode (SLD), to obtain cross-sectional images of a sample at different depths. OCT has been continuously used and developed since then. [2][3][4] The application of OCT to distinguish different ultra thin layers of a non-biologocal sample has also been reported. [5][6] Therefore, OCT enables us to study the cross section of a transparent object with no damage.

However, the OCT system requires a lot of time to obtain a full-field image. That is because a light source in OCT's interferometer has to be focused onto a small area of sample's surface for obtaining the depth information of that area, which is just a small part of overall image area. Although this process gives us accurate results, it consumes a lot of time. To deal with this problem, full-field optical coherence tomography (FFOCT) is proposed and developed. [7] By using the bigger size of light spot in the interferometer and calculating the depth of the surface once at a time, FFOCT can speed the process up. Due to the fact that the light source of FFOCT is not the point source, interference of light from different points of the light source causes unwanted interference fringes on a full-field image. Then, the algorithm for removing unwanted fringes have to be applied in FFOCT. Even the quality of images in FFOCT can be developed in various methods, such as, a 2D smart pixel detector array, a system of multiple sources, and detectors, the removing-fringe algorithms

also have to be developed. [6][8]

Many algorithms have been proposed to clean the fringe of the FFOCT images. Normally, the algorithms, such as, the 4-step and the 5-step phase shift algorithm, were applied in FFOCT. [5] Also, a derivative-based algorithm using backward derivative method was applied for separating the overlapping alphabets coated on different glass chips in both simulation and experiments. [5] The result showed that the derivative-based algorithms spent less time and gave better images than those by the 4-step and 5-step phase shift algorithm. That means the derivative-based algorithm might be the great choice for applying in FFOCT.

Therefore, to improve the images from FFOCT system, a derivative-based algorithm using central derivative method will be investigated and compared to the 5-step phase shift and backward derivative-based (BDB) algorithms in this senior project. It is expected that the central derivative-based (CBD) algorithm takes less time to process and gives us better images than those of the 5-step phase-shift and BDB-algorithm.

For the simplicity of this project, the famous Michelson interferometer (MI) with SLD light source has been employed in FFOCT. That is because SLD is cost-efficient and produces low heat. Moreover, because of the very short coherence length of SLD, the interference pattern of light from different layers does not disrupt one another.

## 1.2  Objective

This project aims to develop the quality of full-field images from FFOCT by using the CDB-algorithm. Thus, the objectives for this project can be concluded as follows:

1. To apply the CDB-algorithm for removing the fringe from the simulation full-field images.

2. To distinguish overlapping alphabets written on the opposite side of the thin transparent layer by FFOCT with CDB-algorithm.

## 1.3 Scope of the Project

For proving the efficiency of the CDB-algorithm, the fringe-removed images applied by this method will be compared with the ones removed by 5-step and BDB-algorithms in both simulation and experimental images. Therefore, in this project, we divide it into 2 parts as follows:

1. **The effect of phase step and fringe contrast to FFOCT with CDB-algorithm**: For removing fringe from the image, the sequence of images taken from MI has to be used. That means phase step and fringe contrast of image's sequence become important. To study the effect of these two factors to FFOCT with CDB-algorithm, sets of the sequence of fringe images will be simulated by varying these two factors as follows:

   (a) For phase step factor, nine sets of the sequence of images will be simulated with the different phase step, which are 10, 20, 30, 40, 50, 60, 70, 80, and 90 degree.

   (b) For fringe contrast factor, five sets of the sequence of images will be simulated with the different values of fringe contrast, which are 0.2, 0.4, 0.6, 0.8, and 1.

   In this part, the efficiency of the CDB-algorithm to remove the fringe in the full-field images will be performed by comparing $\langle \Delta I \rangle$ and $\sigma^2(\Delta I)$ from this algorithm with the ones from 5-step phase shift and BDB-algorithms. $\langle \Delta I \rangle$ and $\sigma^2(\Delta I)$ are the mean and the variance of the intensity difference between the fringe-removed and the original images, respectively.

2. **The efficiency of FFOCT with CDB-algorithm to distinguish the overlapping alphabets**: To prove that FFOCT with CDB-algorithm can be applied to the real experiment, images of alphabets written on different surfaces of a transparent layer, taken by the MI with a SLD light source. These images have unwanted fringes, and the alphabets on different layers cannot be distinguished from each other. For distinguishing the overlapping alphabets on the different surfaces of thin layer, CDB-algorithm will be applied to the taken images. Also, the optical property of alphabet writing materials might

affect the image's quality, thus, in this project, four sets of samples will be prepared which are:

(a) "C" and "U"-alphabets written with a blue marker on the reflected surface under the cover slide and a top of a cover slide, respectively. This sample is called "S1".

(b) "C" and "U"-alphabets written with a silver marker on the reflected surface under the cover slide and a blue marker on a top of a cover slide, respectively. This sample is called "S2".

(c) "C" and "U"-alphabets written with a blue marker on the reflected surface under the cover slide and a silver marker on a top of a cover slide, respectively. This sample is called "S3".

(d) "C" and "U"-alphabets written with a silver marker on the reflected surface under the cover slide and a top of a cover slide, respectively. This sample is called "S4".

"C" and "U"-alphabets in every samples are also on the same position during taking the sequence of images. The efficiency of FFOCT with CDB-algorithm can be defined by comparing the quality of "C" and "U" images applied CDB-algorithm with the ones used 5-step phase shift and BDB-algorithm as well.

## 1.4 Expected Results

We expect that CDB-algorithm should be the method that takes less process time than the 5-step phase shift and BDB-algorithms. It should also produce the better images than those of two algorithms.

# Chapter 2

# Theories

## 2.1 Low Coherence Light

A light source is an important part for determining the characteristic of interference. A practical light source always consists of a range of wavelengths. The range of wavelengths in this manner is called *spectral width*, which is equivalent to the range of frequency or *frequency bandwidth*. The *coherence time* $\tau_c$ is defined as the inversion of frequency bandwidth. It leads to the definition of *coherence length* $L_c$, which is the distance that the phase of wave train remains predictable within coherence time. [9]
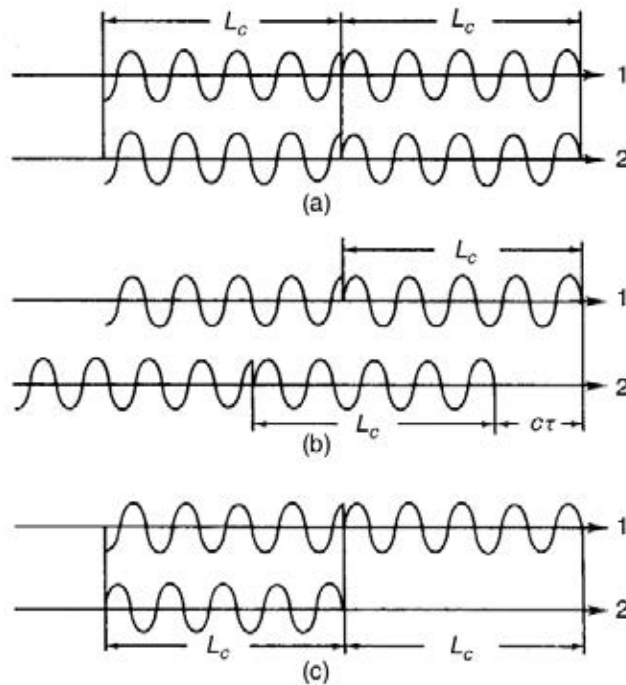


FIGURE 2.1: Wave trains traveling in different paths [10]

When light traveling two different paths interferes, the path difference $c\tau$ of two light beams determines the interference pattern where $c$ is the speed of light and $\tau$ is any time. In the case of Fig. 2.1 (a) and (b), because path difference between two light beam is less than $c\tau_c$, $c\tau < c\tau_c$, a certain pattern of interference can be predicted. However, in Fig. 2.1 (c), completely random interference or no interference fringe occurs, since path difference of two light beams equals or is more than $c\tau_c$.

The difference path $c\tau_c$ is determined as the coherence length $L_c$ or;

$$L_c = c\tau_c, \tag{2.1}$$

when $\tau_c$ is the coherence time. For a light source with a frequency distribution of width $\Delta f$, the coherence time can be approximated as [10]

$$\tau_c = \frac{1}{\Delta f} = \frac{L_c}{c}. \tag{2.2}$$

According to a Gaussian spectral distribution, the accurate coherence length can be calculated from

$$L_c = \sqrt{\frac{2ln2}{\pi}} \frac{\lambda_0^2}{n\Delta\lambda}, \tag{2.3}$$

where $\lambda_0$ the central wavelength of the light source, $\Delta\lambda$ is a full width at half maximum, and $n$ in Eq. 2.3 is the refractive index of the medium. The coherence length of some specific light sources show in Table 2.1.

TABLE 2.1: Coherence length

| Light sources | Central wavelength $\lambda_0$ (nm) | FWHM $\Delta\lambda$ (nm) | Coherence length $L_c$ (nm) |
|---|---|---|---|
| White light source | 1000 | 1700 | 390 |
| Red LED | 750 | 23 | 16246 |
| SLD (SLD-350-HP3-TOW2-PD) | 848.8 | 59.5 | 804 |

In this research, a superluminescent diode (SLD), which is one type of a low coherence light source, is used. Because of its short coherence length, the interference fringe of top and bottom surfaces of ultra thin transparent layer can be separated.

## 2.2 Interference

Since light can be regarded as electromagnetic waves, it exhibits the phenomenon of interference governed by the Superposition Principle. Interference of light occurs when the ripples in the electric and magnetic fields from more than one lightwaves sum up at the same place and time. In our case, the interference of only two lightwaves will be discussed. Rather than the magnetic field, functions for two lightwaves in this report are represented by the electric field as:

$$\vec{E}_1(\vec{r}, t) = \vec{E}_{01} cos(\vec{k}_1 \cdot \vec{r} - \omega t + \phi_1), \tag{2.4}$$

and

$$\vec{E}_2(\vec{r}, t) = \vec{E}_{02} cos(\vec{k}_2 \cdot \vec{r} - \omega t + \phi_2), \tag{2.5}$$

respectively. That is because two field of lightwave is related to the magnetic field by Maxwell's equations. However, only light intensity ($I$) can be real observed, the amplitude terms ($\vec{E}_{01}$, $\vec{E}_{02}$) in Eq. 2.4 and Eq. 2.5 have to be rewritten in terms of $I$. Actually, $I$ is proportional to the time average of the squared amplitude $\langle \vec{E}^2 \rangle_T$, but for easily consideration, the proportionality constant can be neglected without affecting our main purpose. Therefore,

$$I = \langle \vec{E}^2 \rangle_T, \tag{2.6}$$

with

$$\vec{E}^2 = \vec{E}_1^2 + \vec{E}_2^2 + 2\vec{E}_1 \cdot \vec{E}_2. \tag{2.7}$$

Then, the intensity becomes

$$I = \langle \vec{E}_1^2 \rangle_T + \langle \vec{E}_2^2 \rangle_T + 2\langle \vec{E}_1 \cdot \vec{E}_2 \rangle_T. \tag{2.8}$$

Let us consider the first term on the right-hand side of Eq. 2.8. Since the square of the electric field is repeated with the period T,

$$\langle \vec{E}_1^2 \rangle_T = \frac{1}{T} \int_t^{t+T} E_{01}^2 cos^2(\vec{k}_1 \cdot \vec{r} - \omega t + \phi_1)dt = \frac{E_{01}^2}{2}, \qquad (2.9)$$

which is also applies to $\langle \vec{E}_2^2 \rangle_T$. Therefore, the last term on the right-hand side of Eq. 2.8 becomes

$$\langle \vec{E}_1 \cdot \vec{E}_2 \rangle_T = \frac{1}{2}\vec{E}_{01} \cdot \vec{E}_{02} cos(\vec{k}_1 \cdot \vec{r} + \phi_1 - \vec{k}_2 \cdot \vec{r} - \phi_2). \qquad (2.10)$$

From Eq.2.4 and Eq.2.5, $\vec{k}_1 \cdot \vec{r} + \phi_1 - \vec{k}_2 \cdot \vec{r} - \phi_2$ term represents as the *phase difference* between the two waves, newly defined by $\delta$. In other words, the intensity of the interference fringe depends only on the phase difference $\delta$, [9]

$$I = \frac{E_{01}^2}{2} + \frac{E_{02}^2}{2} + \vec{E}_{01} \cdot \vec{E}_{02} cos(\delta). \qquad (2.11)$$

## 2.3 Michelson Interferometer
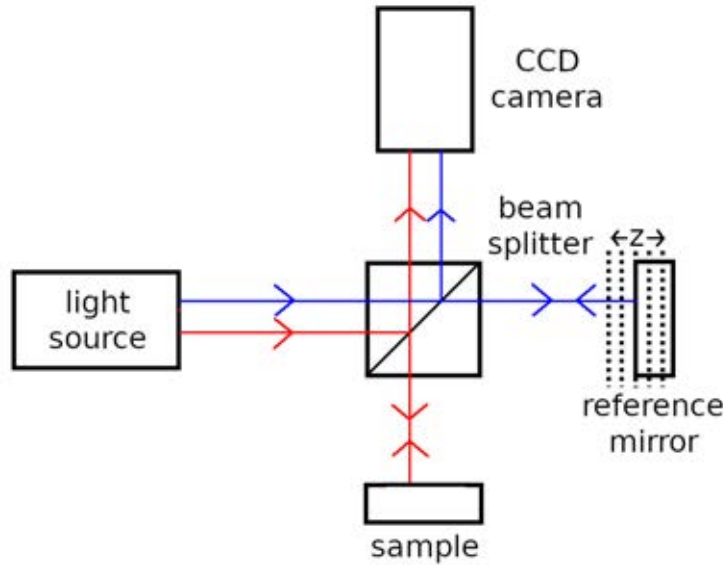


FIGURE 2.2: The Michelson Interferometer

In this project, the Michelson interferometer (MI) depicted in Fig.2.2 is used for generating the interference fringe. It is composed of a light source, a beam splitter, a reference mirror, a test object, and a CCD camera. First, light travels from the light

source to the beam splitter, splitting light into two paths with equal intensity. The first path, colored in blue, goes and reflects from the surface of the reference mirror before going to the camera. The second one, colored in red, goes and reflects from the surface of the object before going to the same camera. From Eq.2.11 by assuming the beginning phase difference $\phi$ between these two beams is zero, the intensity $I$ of interference fringe, taken by the camera, depends on the difference in distance of the two paths as:

$$I = I_0\{1 + \gamma cos[k(r_{mir} - r_{obj})]\} \tag{2.12}$$

where $I_0$ is the background intensity, $\gamma$ is fringe contrast, $r_{mir}$ is the distance travelled by light reflecting at the mirror, and $r_{obj}$ is the distance travelled by light reflecting at the object. The fringe contrast is high when the object reflects a lot of light. Generally, the reference mirror is mounted on a movable base. As the base moves, the intensity of the fringe as a cosine function of $r_{mir}$ can be recorded. However, that is as long as the light source gives ideally monochromatic light. In case coherence light source composing of a range of wavelength, the intensity of the interference fringe becomes

$$I = \int_0^\infty I(f)(1 + \gamma cos(\frac{2\pi}{\lambda} \times 2(z - z_0)))df, \tag{2.13}$$

where $I(f)$ is the distribution of intensity of light as a function of frequency $f$. $z$ is the scanning position of the mirror as in Fig.2.2, and $z_0$ is the scanning position that $r_{mir}$ equals $r_{obj}$. For a source with a Gaussian spectral distribution function, [10]

$$I(f) = I_0\frac{2\sqrt{ln2}}{\sqrt{\pi}\Delta f}exp\{-(2\sqrt{ln2}\frac{f - f_0}{\Delta f})^2\}, \tag{2.14}$$

where $\Delta f$ is the full width at half maximum (FWHM) of the spectral distribution and $f_0$ is the central frequency. Together with Eq. 2.13, we get

$$I = I_0[1 + \gamma exp\{-(\sqrt{\frac{\pi}{2ln2}}\frac{\Delta f}{c}(z - z_0))^2\}cos[\frac{4\pi}{\lambda_0}(z - z_0)]]. \tag{2.15}$$

According to Eq. 2.2, we can write the fringe intensity as a function of $z$ as: [11]

$$I(z) = I_0 + \gamma I_0 exp[-(\frac{z - z_0}{L_c})^2]cos[\frac{4\pi}{\lambda_0}(z - z_0) + \phi_0], \tag{2.16}$$

where $\phi_0$ is a phase offset. The intensity as a function of $z$ is plotted in Fig.2.3 where the dashed line indicates an envelope from the exponential term. The $z_0$ position, referred as zero-order fringe, is the maximum position of the envelope.
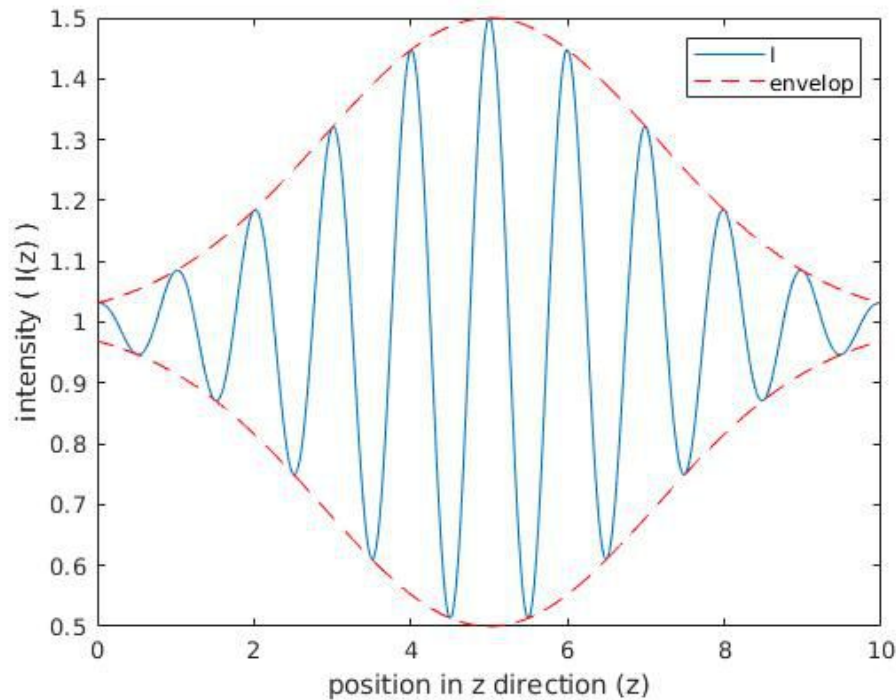


FIGURE 2.3: Interference intensity ($I$) as a function of the scanning position $z$

Due to the fact that our light source has a finite size and the flatness of the object's and the mirror's surfaces, interference of light from different points of the light occurs. Therefore, the phase offset $\phi_0$ in Eq. 2.16 at each point of surface's plane varies and can be written as $\phi_0(x, y)$, which causes interference fringes as in Fig.2.4.
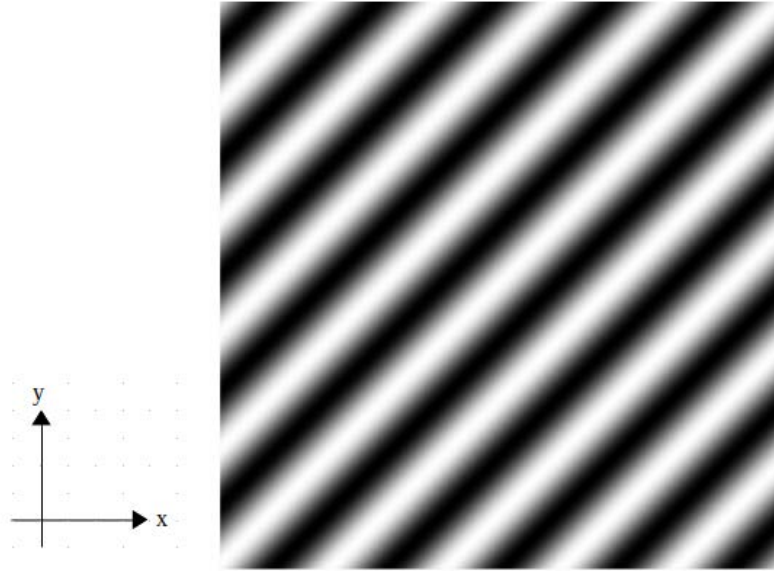
FIGURE 2.4: Simulation of an interference fringe

## 2.4 Full-Field Optical Coherence Tomography

In regular optical coherence tomography (OCT), a low coherence light in MI is focused onto a small area of a object surface and its intensity data are sequences collected along in z-direction by light sensor. The position of object surface in cross-section (z-direction) can be defined from the maximum envelope position ($z_0$) of the intensity data. z-position of the object in XY-plane can be defined by moving the point of light to the next position in XY-plane and calculating the $z_0$ of that point in the same way. Finally, the 3-D surface mapping is built from the $z_0$-data of every xy-position. Despite the accurate result, this process consumes much time. On the other hand, by expanding the size of light beam incidented on a bigger size of the object surface and taking only one sequence of full-field interference fringe image by CCD-camera, it speed the process up. This is referred as *full-field optical coherence tomography (FFOCT)* [5]. But this produces the fringe from the phase offset $\phi_0(x, y)$, because of the interference of light from different points of the light source. For a smooth surface, the interference fringe is similar to the simulated fringe as in Fig.2.4. According to Eq. 2.16, the intensity function can be written in a more general

form as: [5]

$$I(x,y,z) = I_0(x,y) + C(x,y,z)sin[\omega z + \phi(x,y)], \tag{2.17}$$

$$C(x,y,z) = \gamma I_0 exp[-(\frac{z-z_0}{L_c})^2], \tag{2.18}$$

where $z_0$, $\phi$, and $I_0$ are considered as functions of point $(x,y)$ on the surface, $\omega$ is a constant, and $C(x,y,z)$ represents the envelope of intensity. Since our goal is to solve for $C(x,y,z)$, $\phi(x,y)$ has to be removed. 5-step phase shift algorithm and derivative-based algorithm were applied. The calculation methods of these two algorithms are described next.

### 2.4.1 5-step phase shift algorithm

The 5-step phase shift algorithms use 5 respective images taken by the CCD camera to obtain one full-field image with unwanted fringes removed. The algorithms are based on trigonometric properties of the interference pattern function described in Eq. 2.17. Because the scanning range in z-direction is very much smaller than the coherence length $L_c$, $C(x,y,z)$ can be assumed as a constant term. Therefore, the intensity function $I$ can be written as [5]

$$I_1(x,y) = I_0(x,y) + C(x,y)cos[\phi(x,y) - 2\psi] \tag{2.19}$$

$$I_2(x,y) = I_0(x,y) + C(x,y)cos[\phi(x,y) - \psi] \tag{2.20}$$

$$I_3(x,y) = I_0(x,y) + C(x,y)cos[\phi(x,y)] \tag{2.21}$$

$$I_4(x,y) = I_0(x,y) + C(x,y)cos[\phi(x,y) + \psi] \tag{2.22}$$

$$I_5(x,y) = I_0(x,y) + C(x,y)cos[\phi(x,y) + 2\psi] \tag{2.23}$$

where $\psi$ represents phase shift for each scanning step, equivalent to the term $\omega z$ in Eq. 2.17. Using trigonometric identities, we deduce

$$I_2(x,y) - I_4(x,y) = 2C(x,y)sin(\phi)sin(\psi), \tag{2.24}$$

and

$$2I_3(x,y) - I_5(x,y) - I_1(x,y) = 4C(x,y)cos(\phi)sin^2(\psi). \tag{2.25}$$

With the Pythagorean identity, we can write $C(x,y)$ as

$$C(x,y) = \left[ \left( \frac{I_2(x,y) - I_4(x,y)}{2sin(\psi)} \right)^2 + \left( \frac{2I_3(x,y) - I_5(x,y) - I_1(x,y)}{4sin^2(\psi)} \right)^2 \right]^{\frac{1}{2}}. \quad (2.26)$$

The phase step $\psi$ can be recovered from interference data by observing that

$$I_1(x,y) - I_5(x,y) = 4C_i(x,y)sin(\phi)sin(\psi)cos(\psi) \quad (2.27)$$

together with Equation 2.24 can be combined so that

$$cos(\psi) = \frac{1}{2} \frac{I_1(x,y) - I_5(x,y)}{I_2(x,y) - I_4(x,y)}. \quad (2.28)$$

Using Equation 2.26 and 2.28, the envelope of the interference can be extracted. Later, we would refer to this method as "5-step algorithm." It is noted that after $C(x,y)$ from each step of the calculation is arranged together, the complete envelope $C(x,y,z)$ is recovered.

### 2.4.2 Derivative-based algorithms

The derivative-based algorithms make use of derivative properties of the interference pattern to remove unwanted fringes. The number of respective images to be used depends on the type of derivative we choose. For the derivative-based algorithms, let's consider the first-, second-, and third-order derivatives of $I$ in Eq. 2.17 with respect to $z$ as follows

$$I'(x,y,z) = \omega C(x,y,z)cos(\omega z + \phi(x,y)) \quad (2.29)$$

$$I''(x,y,z) = -\omega^2 C(x,y,z)sin(\omega z + \phi(x,y)) \quad (2.30)$$

$$I'''(x,y,z) = -\omega^3 C(x,y,z)cos(\omega z + \phi(x,y)). \quad (2.31)$$

where we neglect the terms $\frac{dC}{dz}$ assuming that the scanning range in z-direction is much smaller than the coherence length $L_c$. From these equations, we deduce

$$I''^2 - I'''I' = \omega^4 C^2(x,y,z) \quad (2.32)$$

Practically, derivative with respect to phase shift $d(\omega z)$ is taken. It helps to get rid of the constant $\omega^4$. To avoid the presence of complex value, we let

$$C(x,y,z) = |I''^2 - I'''I'|^{\frac{1}{2}} \tag{2.33}$$

In this project, two types of derivative-based algorithm are used.

1. **Backward derivative-based (BDB) algorithm**:

   In BDB-algorithm, the backward-difference of $1^{st}$ order is used. It is defined by the set of equations as follows: [5]

   $$I' = \frac{I_2 - I_1}{dz}, \tag{2.34}$$

   $$I'' = \frac{I_3 - 2I_2 + I_1}{dz}, \tag{2.35}$$

   $$I''' = \frac{I_4 - 3I_3 + 3I_2 - I_1}{dz}, \tag{2.36}$$

   where $I_n$ denotes the intensity at $n^{th}$ image, and $n = 1,2,3,4$. $I'$, $I''$, and $I'''$ belong to the first image of the sequence.

2. **Central derivative-based (CDB) algorithm**:

   In CDB-algorithm, the central-difference of $4^{th}$ order is used. It is defined by the set of equations as follows: [12]

   $$I' = \frac{-I_6 + 8I_5 - 8I_3 + I_2}{12dz}, \tag{2.37}$$

   $$I'' = \frac{-I_6 + 16I_5 - 30I_4 + 16I_3 - I_2}{12dz^2}, \tag{2.38}$$

   $$I''' = \frac{-I_7 + 8I_6 - 13I_5 + 13I_3 - 8I_2 + I_1}{8dz^3}, \tag{2.39}$$

where $I_n$ denotes the intensity at $n^{th}$ image, and $n = 1,2,3,4,5,6,7$. $I'$, $I''$, and $I'''$ belong to the fourth image of the sequence. As in 5-step algorithm, the derivative-based algorithm can remove the phase offset and retrieve the envelope scanned along $z$ at each point $(x,y)$. It is noted that not only the phase offset but all phase information is deleted.

## 2.5   Quality of Retrieved Images

After the removal of interference fringes, the envelope of the interference pattern is retrieved. The retrieved envelope is more resemble to the original envelope, the better the retrieved images become. Therefore, to define the quality of images retrieved from each algorithm in section 2.4, we subtract the intensity of retrieved envelope from the original one. For each particular area, $\Delta I(z_i)$ is shown as in Fig2.5, where $z_i$ denotes a particular scanning position. Eventually, $\Delta I(x_i, y_i, z_i)$ is defined for every $(x_i, y_i)$ area. Then, the mean of difference in intensity is reported according to

$$\langle \Delta I \rangle = \frac{\sum\limits_{x_i, y_i, z_i} \Delta I(x_i, y_i, z_i)}{N}, \tag{2.40}$$

where $N$ is the total number of pixels in a complete sequence of retrieved images. Also, the variance of difference in intensity is described as

$$\sigma^2(\Delta I) = \frac{\sum\limits_{x_i, y_i, z_i} (\Delta I(x_i, y_i, z_i) - \langle \Delta I \rangle)^2}{N - 1}, \tag{2.41}$$

with the same $N$.
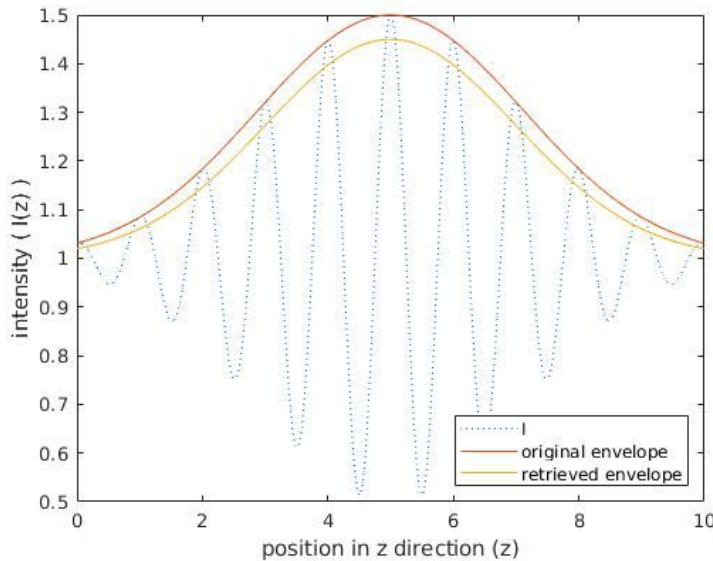


FIGURE 2.5: Original envelope and retrieved envelope as a function of the scanning position $z$

# Chapter 3

# Methodology

## 3.1 The effect of phase step and fringe contrast to CDB-algorithm

Since the sequence of images is important for CDB-algorithm, the phase step and fringe contrast are the main factors that have to be considered. In both cases, the sequence of interference fringes images on a plain white background, $I_0 = 1$, with the size of $300 \times 300$ pixels is simulated by using Eq. 2.16, which is

$$I(z) = I_0 + \gamma I_0 exp[-(\frac{z - z_0}{L_c})^2]cos[\frac{4\pi}{\lambda_0}(z - z_0) + \phi_0].$$ (3.1)

Also, the phase step defined by:

$$\psi = \frac{4\pi}{\lambda_0}dz \cdot \frac{180}{\pi}.$$ (3.2)

is much smaller than the coherence length of light source. Thus, the exponential factor in Eq. 3.1 is assumed to be 1. The characteristics of a light source, used in our simulation, is the same as the SLD in the experiment which its central wavelength $\lambda_0$ is 848.8 nm, a full width at half maximum of spectrum $\Delta\lambda$ is 59.5 nm, and a coherence length $L_c$ is 8.04 $\mu m$. Images with interference fringes are simulated from phase 0 to 720 degree. The phase offset $\phi_0$ determines the characteristics of the fringe through

$$\phi_0(x, y) = 2\pi(\frac{x}{L_x} + \frac{y}{L_y}),$$ (3.3)

where $L_x = 100$ and $L_y = 99$. They determine the width of the fringe in the x and y direction respectively. The effect of phase step and fringe contrast are studied as

follows:

1. The effect of phase step to the CDB-algorithm is studied by varied phase step as 10, 20, 30, 40, 50, 60, 70, 80, and 90 degree. Fringe contrast ($\gamma$) in Eq. 3.1 is given as 1.

2. The effect of fringe contrast to the CDB-algorithm is studied by varied fringe contrast ($\gamma$) as 0.2, 0.4, 0.6, 0.8, and 1. The phase step is fixed to be 84.83 degree which is equivalent to the step size of 100nm, the same value as in the experiment.

In addition, the process time for CDB-, BDB-, and 5-step algorithms with 84.83-degree phase step and 1 for the fringe contrast will be compared. For one particular set of parameters, the *deletion results*, relating with the qualities of fringe-removed images, will be reported as two variables. One is the mean $\langle \Delta I \rangle$, and the other is the variance $\sigma^2(\Delta I)$, according to Eq. 2.40 and 2.41, respectively. Full codes for simulation are given in appendices.

## 3.2 The efficiency of the FFOCT with CDB-algorithm
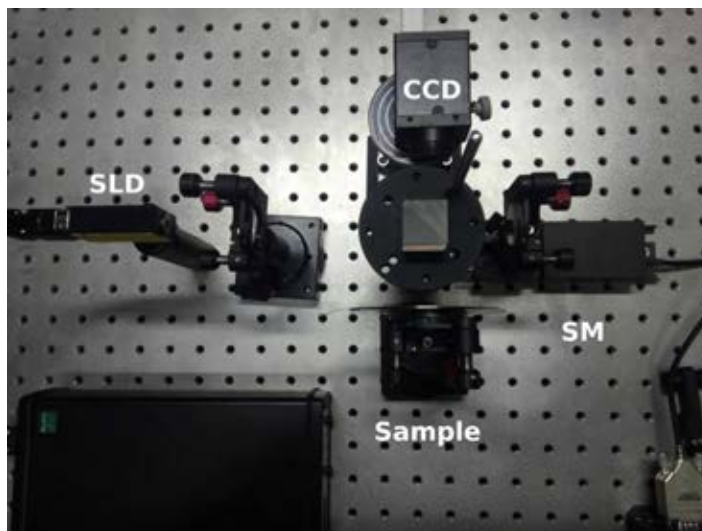


FIGURE 3.1: The configuration of the Michelson interferometer

To prove that FFOCT with CDB-algorithm can be separated the two overlapping alphabets on the different surface of transparent layer, it is applied to the sequence

of images taken by the Michelson interferometer (MI) as shown in Fig. 3.1. The specifications of parts used in our MI are

1. SLD : SLD-350-HP3-TOW2-PD (Superlum Diode Ltd.) with $\lambda_0$ 848.8 nm and $\Delta\lambda$ 59.5 nm

2. CCD : SONY XCD-V50CR

3. Stepping motor (SM) : KCube DC Motor Controller KDC101 with a stage MTS25/M-Z8 (Thorlabs, Inc.)

For the effects of writing materials and surface to the quality of images, four types of samples are prepared in our experiment. They are:

1. "C" and "U"-alphabets are written with a blue marker on the reflected surface under the cover slide and a top of a cover slide, as in Fig. 3.2a and Fig. 3.2b respectively. This sample is called "S1".

2. "C" and "U"-alphabets are written with a silver marker on the reflected surface under the cover slide and a blue marker on a top of a cover slide, as in Fig. 3.3a and Fig. 3.3b respectively. This sample is called "S2".

3. "C" and "U"-alphabets are written with a blue marker on the reflected surface under the cover slide and a silver marker on a top of a cover slide, as in Fig. 3.4a and Fig. 3.4b respectively. This sample is called "S3".

4. "C" and "U"-alphabets are written with a silver marker on the reflected surface under the cover slide and a top of a cover slide, as in Fig. 3.5a and Fig. 3.5b respectively. This sample is called "S4".

The cover slides thickness of all samples are 0.13-0.16 mm. Then, the cover slide is attached onto the reflected surface, such that the alphabets overlap as in Fig. 3.6. Finally, the sample is attached to the magnetic stand. Sequential images of the alphabet at each layer are taken with step size $dz = 100nm$, equivalent to 84.83-degree phase step, and are later processed by CDB-, BDB-, and 5-step algorithms. The quality of fringe-removed images in both "C" and "U" of all samples from FFOCT with CDB-algorithm will be compared with the ones from FFOCT with BDB- and 5-step algorithms.

(a)           (b)
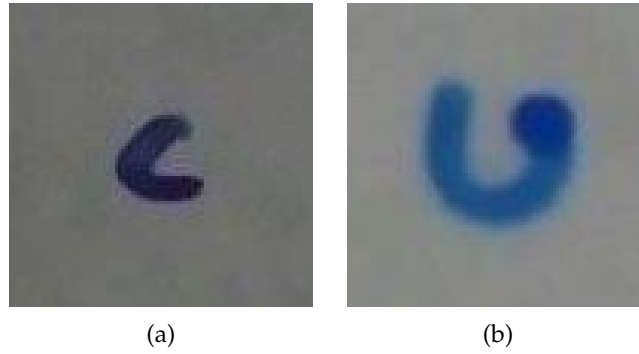
FIGURE 3.2: "C" and "U"-alphabets on each surface of sample S1:
(a) "C" written with the blue marker on a cover slide, and
(b) "U" written with the blue marker on a reflected surface



(a)           (b)
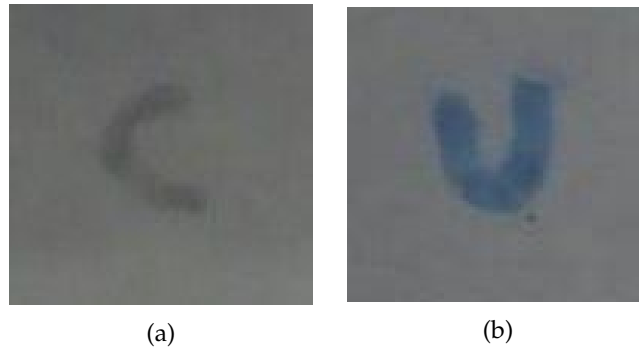
FIGURE 3.3: "C" and "U"-alphabets on each surface of sample S2:
(a) "C" written with the silver marker on a cover slide, and
(b) "U" written with the blue marker on a reflected surface



(a)           (b)
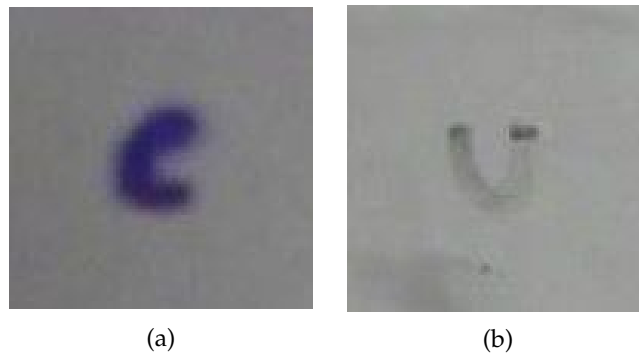
FIGURE 3.4: "C" and "U"-alphabets on each surface of sample S3:
(a) "C" written with the blue marker on a cover slide, and
(b) "U" written with the silver marker on a reflected surface

(a)             (b)
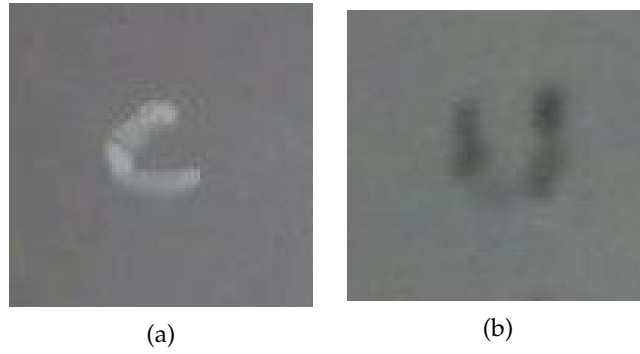
FIGURE 3.5: "C" and "U"-alphabets on each surface of sample S4:
(a) "C" written with the silver marker on a cover slide, and
(b) "U" written with the silver marker on a reflected surface



(a) S1           (b) S2

(c) S3           (d) S4
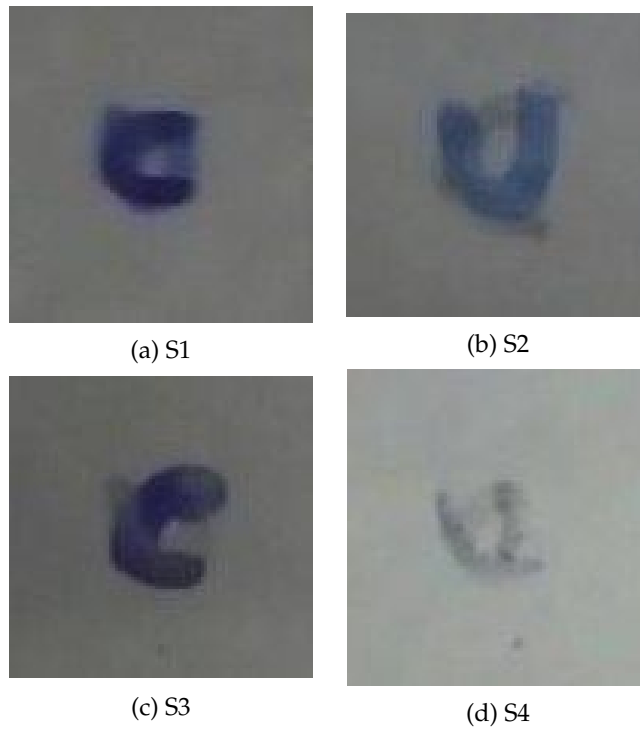
FIGURE 3.6: Overlapping alphabets of samples

# Chapter 4

# Results

## 4.1 The effect of phase step and fringe contrast to CDB-algorithm

In this part, the sequence of interference fringes images on a plain white background, $I_0 = 1$, with the XY-plane size of $300 \times 300$ pixels is simulated by using Eq. 3.1. The central wavelength $\lambda_0$ and the full width at half maximum of spectrum $\Delta\lambda$ for this simulation are 848.8 nm and 59.5 nm, respectively. Calculation time, the effect of phase step, and the effect of fringe contrast will be observed.

### 4.1.1 Calculation time

For calculation time comparison, the phase step of a sequence of images is given as 84.83 degree, which is the same as in our experiment. The fringe contrast $\gamma$ is 1. The simulated sequence of images is deleted interference fringe by applying central derivative-based (CDB), backward derivative-based (BDB), and 5-step algorithms, respectively. Finally, the calculation time for these three fringe-removing methods is compared. The calculation time for these three algorithms are shown in Table 4.1.

TABLE 4.1: Calculation time

| Algorithm | Calculation time (second) |
|:---------:|:-------------------------:|
| CDB       | 0.6550                    |
| BDB       | 0.3826                    |
| 5-step    | 0.5375                    |

According to our calculation, the calculation time for BDB-algorithm is fastest. 5-step and CDB-algorithms require 0.1549 s and 0.2724 s more processing time than the BDB-algorithm, respectively. The difference in processing time for all algorithms is less than a half second, which is insignificant.

### 4.1.2 The effect of phase step to CDB-algorithm

In this study, the sequences of images with fringe contrast of 1 at phase step 10, 20, 30, 40, 50, 60, 70, 80, and 90 degree are simulated. The simulated image's sequence is deleted interference fringe by applying CDB-, BDB-, and 5-step algorithms, respectively. Two variables, the mean $\langle \Delta I \rangle$ defined in Eq. 2.40, and the variance $\sigma^2(\Delta I)$ defined in Eq. 2.41, are plotted as a function of phase step. Since $\langle \Delta I \rangle$ is calculated by finding the mean of the different intensity between the fringe-removed images and the original ones, $\langle \Delta I \rangle$ relates to the average of intensity drop in the fringe-removed images obtained by each algorithm. That means more minus $\langle \Delta I \rangle$ shows more decreasing intensity of fringe-removed images. Also, $\sigma^2(\Delta I)$ represents the parasitic fringe of the fringe-removed images from each algorithm, as well. The less $\sigma^2(\Delta I)$, the cleaner fringe removing. Data of $\langle \Delta I \rangle$ and $\sigma^2(\Delta I)$, which are the function of phase step, for CDB-, BDB-, and 5-step algorithms are presented at Appendix A.1 in Table A.1, A.2, and A.3, respectively. A curve of phase step and $\langle \Delta I \rangle$ of three algorithms are shown in Fig. 4.1a. The relation between phase step and $\sigma^2(\Delta I)$ of three algorithms are also plotted in Fig. 4.1b.

According to our results:

1. For CDB-algorithm, $\langle \Delta I \rangle$ and $\sigma^2(\Delta I)$ vary with phase step. The reduction of $\langle \Delta I \rangle$ increases with the phase step. The range of $\langle \Delta I \rangle$ is from 0 to -0.1206. $\sigma^2(\Delta I)$ is also proportional to the phase step. However, in the range 10-40 degree of phase step, $\sigma^2(\Delta I)$ is no significant difference compared to other algorithms.

2. For BDB-algorithm, the reduction of $\langle \Delta I \rangle$ also increases with the phase step. The range of $\langle \Delta I \rangle$ is from $-0.0063$ to $-0.4268$. Comparing with CDB-algorithm, $\langle \Delta I \rangle$ of BDB-algorithm is more reducing when the phase step increases. On

the other hand, $\sigma^2(\Delta I)$ does not vary with phase step. Because $\sigma^2(\Delta I)$ is in the order of $10^{-25}$, its variation is insignificantly low.

3. For 5-step algorithm, $\langle \Delta I \rangle$ and $\sigma^2(\Delta I)$ are almost constant with the change of phase step. Also, $\langle \Delta I \rangle$ and $\sigma^2(\Delta I)$ are just only in the order of $10^{-5}$ and $10^{-8}$, respectively.
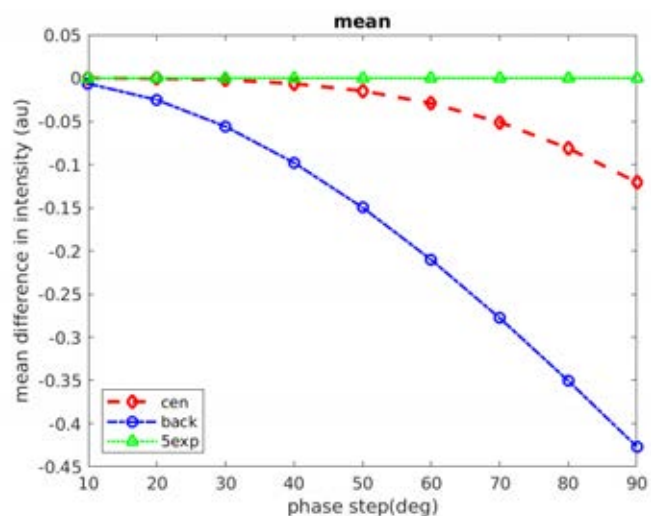


(a)
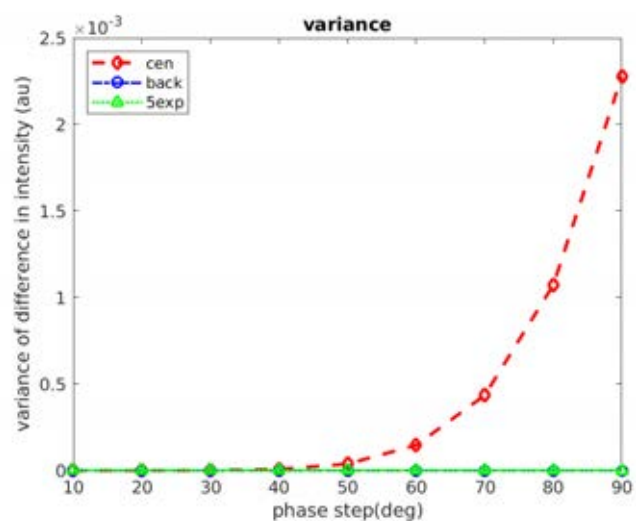
(b)

FIGURE 4.1: The relation curve between phase step and
(a) the mean $\langle \Delta I \rangle$, (b) the variance $\sigma^2(\Delta I)$

.

In summary, the reduction of $\langle\Delta I\rangle$ of CDB-algorithm is around one-tenth of the background intensity, while the one of BDB-algorithm is almost half of the background intensity. That means the fringe-removed image from CDB-algorithm is brighter than the one from BDB-algorithm. However, the intensity of the image from both algorithms will be dropped while the phase step increases. In contrast, $\langle\Delta I\rangle$ of 5-step algorithm is nearly zero. It can be concluded that the intensity from 5-step algorithm is no change. Also, $\sigma^2(\Delta I)$ from BDB- and 5-step algorithm is nearly zero, these two algorithms shall get rid of almost interference fringe from the image in any value of phase step. Moreover, since $\sigma^2(\Delta I)$ of CDB-algorithm is insignificantly low especially at small phase step, CDB-algorithm should be suited for removing fringe from the images in case phase step is not more than 40 degree.

### 4.1.3   The effect of fringe contrast to CDB-algorithm

In this section, the sequences of images at phase step 84.83 degree, which is the same as in our experiment, with fringe contrast of 0.2, 0.4, 0.6, 0.8, and 1 are simulated. The simulated image's sequence is deleted interference fringe by applying CDB-, BDB-, and 5-step algorithms, respectively. Two variables, the mean $\langle\Delta I\rangle$ defined in Eq. 2.40, and the variance $\sigma^2(\Delta I)$ defined in Eq. 2.41, are also plotted as a function of fringe contrast. Data of $\langle\Delta I\rangle$ and $\sigma^2(\Delta I)$ ,which are the function of fringe contrast, for CDB-, BDB-, and 5-step algorithms are presented at Appendix A.1 in Table A.4, A.5, and A.6, respectively. A curve of fringe contrast and $\langle\Delta I\rangle$ of all three algorithms are shown in Fig. 4.2a. The relation between fringe contrast and $\sigma^2(\Delta I)$ of three algorithms are plotted in Fig. 4.2b.

In accordance with our results,

1. For CDB-algorithm, $\langle\Delta I\rangle$ and $\sigma^2(\Delta I)$ vary with fringe contrast. The reduction of $\langle\Delta I\rangle$ linearly increases with the increasing of fringe contrast. The range of $\langle\Delta I\rangle$ is from -0.0198 to -0.0991. But $\sigma^2(\Delta I)$ is increasingly proportional to the fringe contrast. The more fringe contrast, the more $\sigma^2(\Delta I)$. However, $\sigma^2(\Delta I)$ is in the order of $10^{-3}$, it can be considered that its variation is low.

2. For BDB-algorithm, $\langle\Delta I\rangle$ varies with fringe contrast as in CDB-algorithm. The reduction of $\langle\Delta I\rangle$ also linearly increases with the increasing of fringe contrast.

The range of $\langle \Delta I \rangle$ is from $-0.0774$ to $-0.3871$, which is lower than the one of CDB-algorithm. That means the intensity of fringe-removed images from BDB-algorithm is lower than the one from CDB-algorithm. On the other hand, $\sigma^2(\Delta I)$ does not vary with fringe contrast and is only in the order of $10^{-25}$. In other words, the fringe-removed image from this method is almost clear from interference fringe.

3. For 5-step algorithm, $\langle \Delta I \rangle$ and $\sigma^2(\Delta I)$ are rarely to vary with fringe contrast. They are only in the order of $10^{-13}$ and $10^{-27}$, respectively, which are insignificantly low. Also, $\sigma^2(\Delta I)$ from 5-step algorithm is close to the one of BDB-algorithm. It can be concluded that the intensity of fringe-removed image, applied with this algorithm, is almost the same as the original one. The fringe-removed image from this method is also clear from interference fringe.

In summary, the reduction of $\langle \Delta I \rangle$ from CDB- and BDB-algorithms increases linearly with fringe contrast comparing with the one from 5-step algorithm. The more fringe contrast, the more $\langle \Delta I \rangle$ from these two algorithms. The reduction of $\langle \Delta I \rangle$ for CDB-algorithm is less than one-tenth of the background intensity while that for BDB-algorithm is around 0.3. Even the brightness of fringe-removed images from the two methods decreases with fringe contrast, fringe-removed images from CDB-algorithm are still brighter than the ones from BDB-algorithm. In contrast, the fringe-removed images from 5-step algorithm shows almost no drop in intensity. Also, $\sigma^2(\Delta I)$ from BDB- and 5-step algorithms is very low, compared to the one from CDB-algorithm, which increases with fringe contrast. Thus, fringe-removed images from CDB-algorithm also have more remaining interference fringe compared to the ones from BDB- and 5-step algorithms.
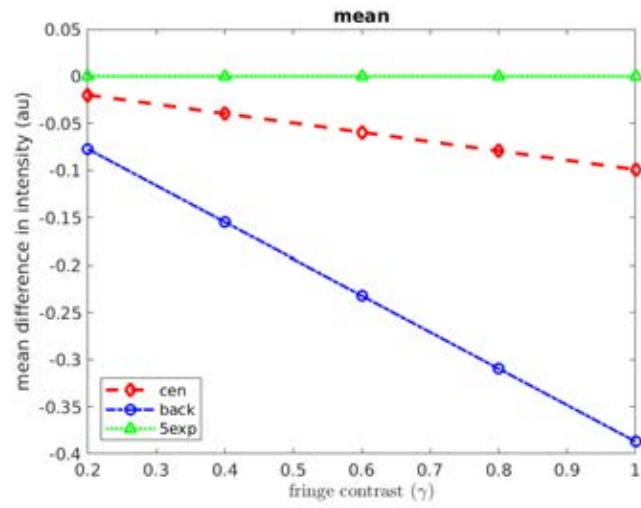
(a)
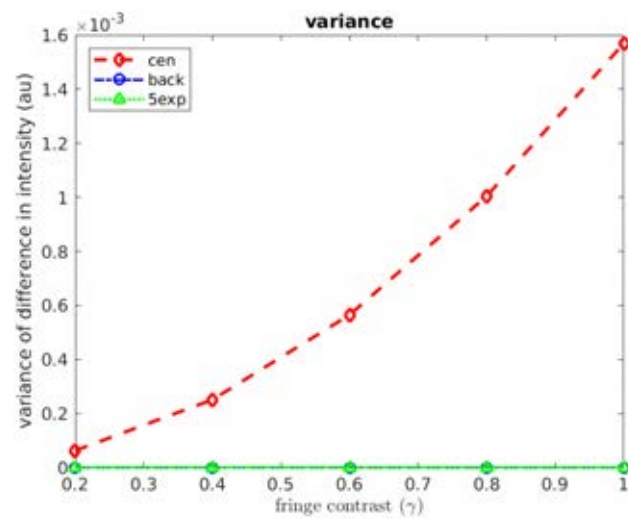


(b)

FIGURE 4.2: The relation curve between fringe contrast and
(a) the mean $\langle \Delta I \rangle$, (b) the variance $\sigma^2(\Delta I)$

.

## 4.2 The effect of optical properties of material to FFOCT with CDB-algorithm

To study the effects of the optical properties of writing materials and surface to the quality of images, sample S1, S2, S3, and S4 are prepared in this experiment as follows:

1. **S1**: "C"- and "U"-alphabets are written with a blue marker on the reflected surface under the cover slide and a top of a cover slide, as in Fig. 3.2a and Fig. 3.2b respectively.

2. **S2**: "C"- and "U"-alphabets are written with a silver marker on the reflected surface under the cover slide and a blue marker on a top of a cover slide, as in Fig. 3.3a and Fig. 3.3b respectively.

3. **S3**: "C"- and "U"-alphabets are written with a blue marker on the reflected surface under the cover slide and a silver marker on a top of a cover slide, as in Fig. 3.4a and Fig. 3.4b respectively.

4. **S4**: "C"- and "U"-alphabets are written with a silver marker on the reflected surface under the cover slide and a top of a cover slide, as in Fig. 3.5a and Fig. 3.5b respectively.

The size of alphabets is around 2 mm $\times$ 2 mm, and the cover slides thickness of all samples are 0.13-0.16 mm. Also, the "C"- and "U"- alphabets are set to be overlap as in Fig. 3.6. Sequential images of the overlapping alphabets at each layer are taken with step size, $dz = 100nm$, which is equivalent to 84.83-degree of phase step. Finally, they are processed by CDB-, BDB-, and 5-step algorithms, for producing the fringe-removed images in both "C" and "U" of all samples.

### 4.2.1 The effect of fringe-removing algorithms

In this section, fringe-removed images from CDB-, BDB-, and 5-step algorithms in the same sample will be shown. The quality of fringe-removed images from each algorithm will be compared.

**Sample S1**

In case of sample S1, Fig. 4.3a and 4.3b show images with interference fringe taken by the CCD camera of the reflected surface (bottom layer) and the top of a cover slide (top layer), respectively. Fig. 4.3c and 4.3d show fringe-removed images of the bottom and top layers of sample from CDB-algorithm, respectively. Fig. 4.3e and 4.3f show fringe-removed images of the bottom and top layers of sample from BDB-algorithm, respectively. Fig. 4.3g and 4.3h show fringe-removed images of the bottom and top layers of sample from 5-step algorithm, respectively.

In Fig. 4.3a and 4.3b, the alphabets are not clearly seen as "C" and "U". After applied each three fringe-removing algorithm, "C" on the bottom and "U" on the top layer are separately seen. However, since "U"-alphabet is written on the top surface of the cover slide, which the intensity of reflected light is lower than the one of "C"-alphabet, the image of "U"-alphabet is darker than the one of "C"-alphabet in every method. The fringe-removed images from CDB-algorithm also have higher intensity than the ones from BDB-algorithm. But its intensity is almost the same as the ones from 5-step algorithm. This agrees with the simulation result in Section 4.1.2. However, the fringe-removed images from CDB-algorithm have most parasitic fringe. Moreover, "C"-alphabet is seen more obviously than "U"-alphabet in every fringe-removing method. This is because both "C"- and "U"-alphabets are written with the blue marker, which produces low fringe contrast. But "C"-alphabet is written on the reflected surface, producing high fringe contrast background. Thus, the contradiction between the fringe contrast of "C"-alphabet and the one of background is high. On the other hands, "U"-alphabet is written on the surface of the cover slide, producing low fringe contrast. Therefore, the contradiction between the fringe contrast of "U"-alphabet and the one of background is low.

FIGURE 4.3: FFOCT of sample S1 for:
(a) bottom and (b) top layer fringe,
(c) bottom and (d) top layer applied by CDB-algorithm,
(e) bottom and (f) top layer applied by BDB-algorithm,
(g) bottom and (h) top layer applied by 5-step algorithm.

**Sample S2**

In case of sample S2, Fig. 4.4a and 4.4b show images with interference fringe taken by the CCD camera of the reflected surface (bottom layer) and the top of a cover slide (top layer), respectively. Fig. 4.4c and 4.4d show fringe-removed images of the bottom and top layers of sample from CDB-algorithm, respectively. Fig. 4.4e and 4.4f show fringe-removed images of the bottom and top layers of sample from BDB-algorithm, respectively. Fig. 4.4g and 4.4h show fringe-removed images of the bottom and top layers of sample from 5-step algorithm, respectively.

In Fig. 4.4a and 4.4b, the alphabets are not also clearly seen as "C" and "U" as in sample S1. After applied each three fringe-removing algorithm, "C" on the bottom and "U" on the top layer are separately seen. However, since "U"-alphabet is written on the top surface of the cover slide, which the intensity of reflected light is lower than the one of "C"-alphabet, the image of "U"-alphabet is darker than the one of "C"-alphabet in every method. The fringe-removed images from CDB-algorithm have also higher intensity than the ones from BDB-algorithm. But its intensity is almost the same as the ones from 5-step algorithm as in sample S1. This agrees with the simulation result in Section 4.1.2. However, the fringe-removed images from CDB-algorithm have most parasitic fringe. Moreover, both "C"- and "U"-alphabets are not obviously seen in every fringe-removing method. This is because "C"-alphabet is written with the silver marker, producing high fringe contrast, on the reflected surface, also producing high fringe contrast background. Also, "U"-alphabet is written with the blue marker, producing low fringe contrast, on the surface of the cover slide, also producing low fringe contrast. Thus, the contradiction between the fringe contrast of both alphabets and their background is low, making them seen vaguely.
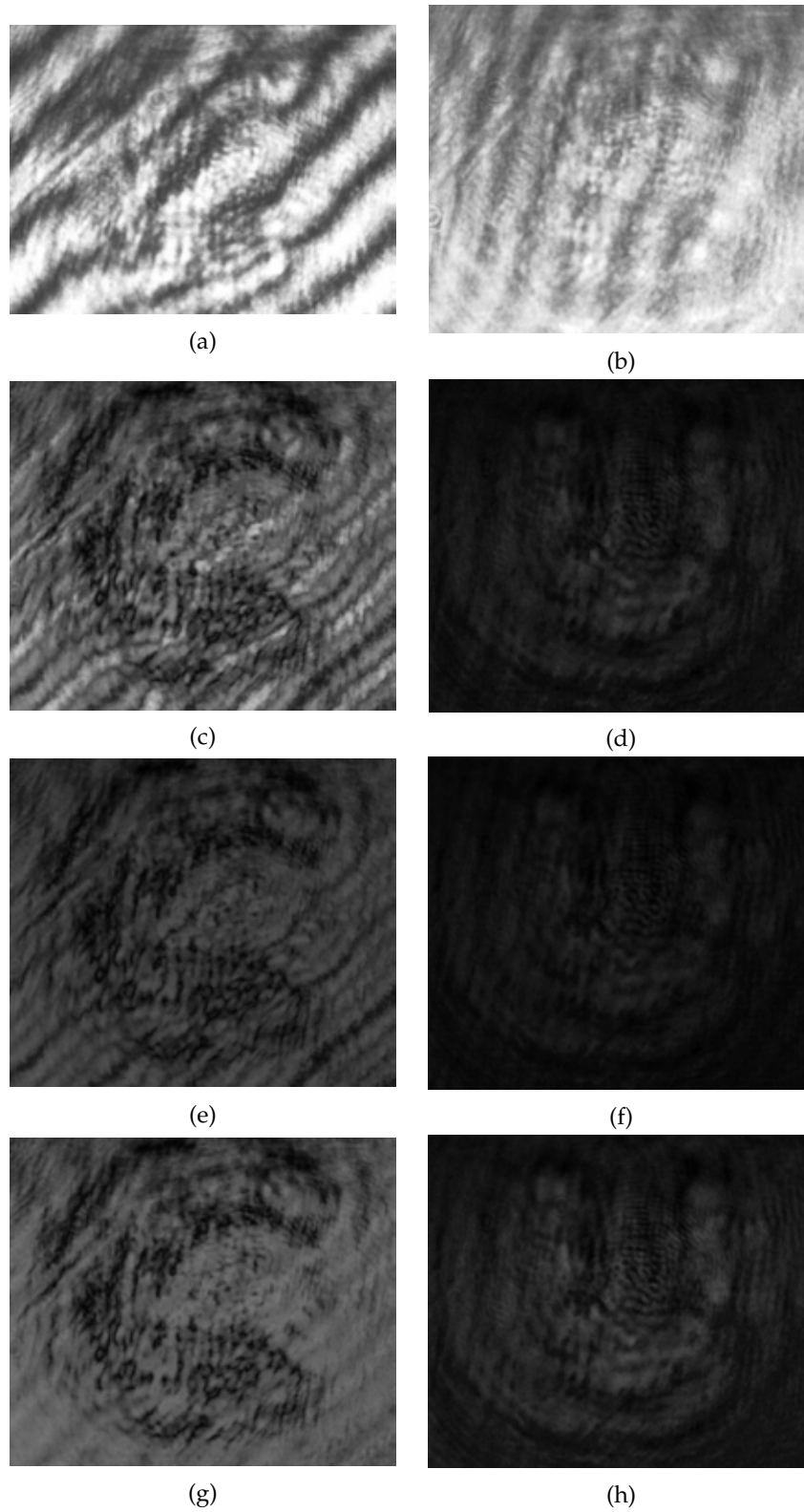
FIGURE 4.4: FFOCT of sample S2 for:
(a) bottom and (b) top layer fringe,
(c) bottom and (d) top layer applied by CDB-algorithm,
(e) bottom and (f) top layer applied by BDB-algorithm,
(g) bottom and (h) top layer applied by 5-step algorithm.

**Sample S3**

In case of sample S3, Fig. 4.5a and 4.5b show images with interference fringe taken by the CCD camera of the reflected surface (bottom layer) and the top of a cover slide (top layer), respectively. Fig. 4.5c and 4.5d show fringe-removed images of the bottom and top layers of sample from CDB-algorithm, respectively. Fig. 4.5e and 4.5f show fringe-removed images of the bottom and top layers of sample from BDB-algorithm, respectively. Fig. 4.5g and 4.5h show fringe-removed images of the bottom and top layers of sample from 5-step algorithm, respectively.

In Fig. 4.5a and 4.5b, the alphabets are not clearly seen as "C" and "U" as in sample S1 and S2. After applied each three fringe-removing algorithm, "C" on the bottom and "U" on the top layer are also separately seen as in sample S1 and S2. In the same way as in previous samples, the image of "U"-alphabet is darker than the one of "C"-alphabet in every method because "U"-alphabet is on the top surface of the cover slide, but "C"-alphabet is on the reflected surface. Also, the images from CDB-algorithm have higher intensity than the ones from BDB- and 5-step algorithms. This disagrees with the simulation result in Section 4.1.2, that is because of the experimental factors, such as, flatness of sample's surface, etc. However, the fringe-removed images from CDB-algorithm have most parasitic fringe. Moreover, both "C"- and "U"-alphabets are obviously seen in every fringe-removing method. This is because "C"-alphabet is written with the blue marker, producing low fringe contrast, on the reflected surface, producing high fringe contrast background. Also, "U"-alphabet is written with the silver marker, producing high fringe contrast, on the surface of the cover slide, producing low fringe contrast. Thus, the contradiction between the fringe contrast of both alphabets and their background is high, making them seen obviously.
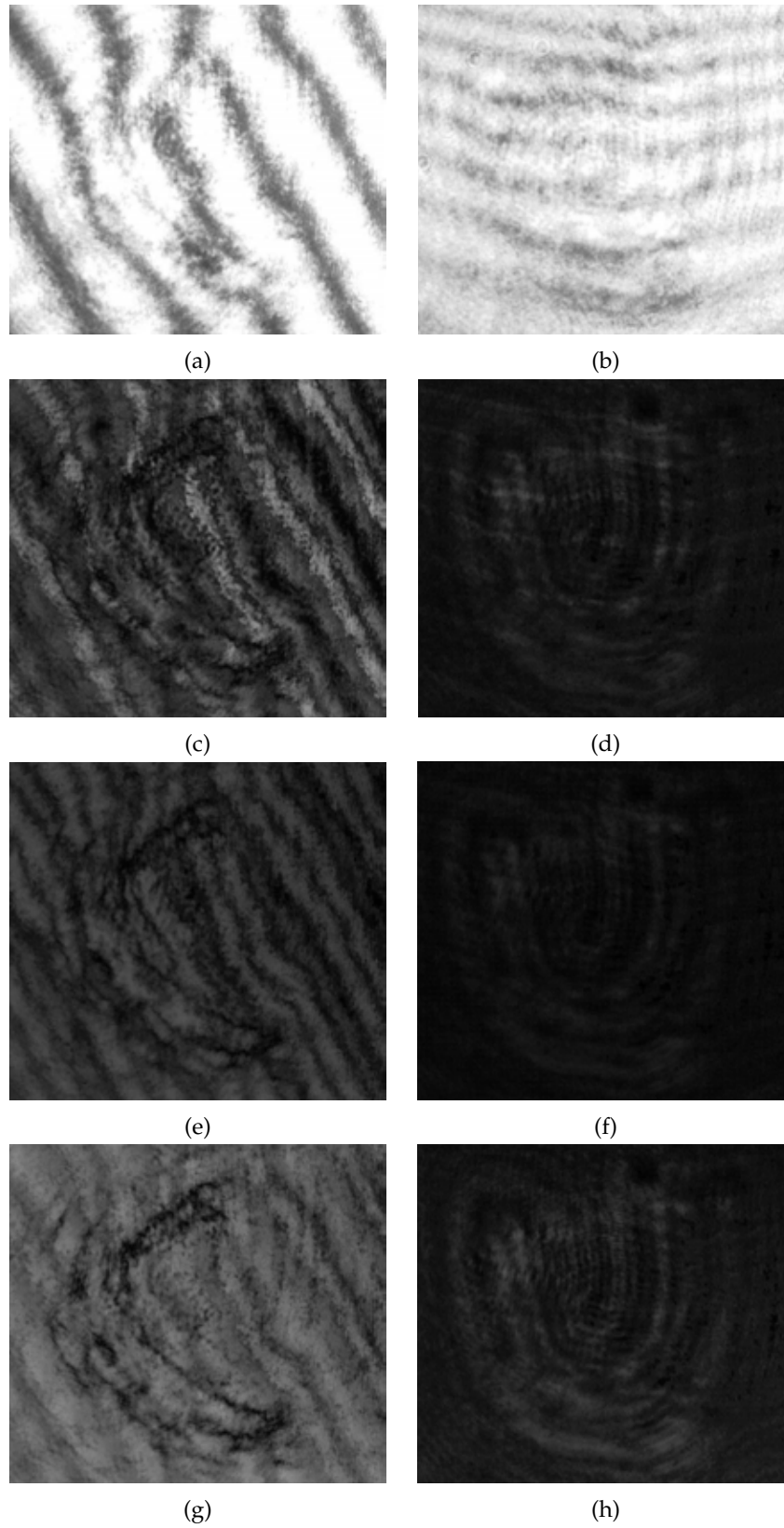
FIGURE 4.5: FFOCT of sample S3 for:
(a) bottom and (b) top layer fringe,
(c) bottom and (d) top layer applied by CDB-algorithm,
(e) bottom and (f) top layer applied by BDB-algorithm,
(g) bottom and (h) top layer applied by 5-step algorithm.

**Sample S4**

In case of sample S4, Fig. 4.6a and 4.6b show images with interference fringe taken by the CCD camera of the reflected surface (bottom layer) and the top of a cover slide (top layer), respectively. Fig. 4.6c and 4.6d show fringe-removed images of the bottom and top layers of sample from CDB-algorithm, respectively. Fig. 4.6e and 4.6f show fringe-removed images of the bottom and top layers of sample from BDB-algorithm, respectively. Fig. 4.6g and 4.6h show fringe-removed images of the bottom and top layers of sample from 5-step algorithm, respectively.

In Fig. 4.6a and 4.6b, the alphabets are not clearly seen as "C" and "U" as in sample S1, S2, and S3. After applied each three fringe-removing algorithm, "C" on the bottom and "U" on the top layer are also separately seen as in sample S1, S2, and S3. However, since "U"-alphabet is on the top surface of the cover slide and "C"-alphabet is on the reflected surface, the image of "U"-alphabet is darker than the one of "C"-alphabet in every method, the same as in the previous samples. The images from CDB-algorithm have also higher intensity than the ones from BDB- and 5-step algorithms. This disagrees with the simulation result in Section 4.1.2, that is because of the experimental factors, such as, flatness of sample's surface, etc. However, the fringe-removed images from CDB-algorithm have most parasitic fringe. Moreover, "U"-alphabet is obviously seen more than "C"-alphabet in every fringe-removing method; however, it is noted that although the edge of "C"-alphabet is clearly seen, the inside is not. This is because both "C"- and "U"-alphabets are written with the silver marker, producing high fringe contrast. "C"-alphabet is also written on the reflected surface; thus, the contradiction between the fringe contrast of "C"-alphabet and the one of background is low. On the other hand, "U"-alphabet is written on the surface of the cover slide; therefore, the contradiction between the fringe contrast of "U"-alphabet and the one of background is high.
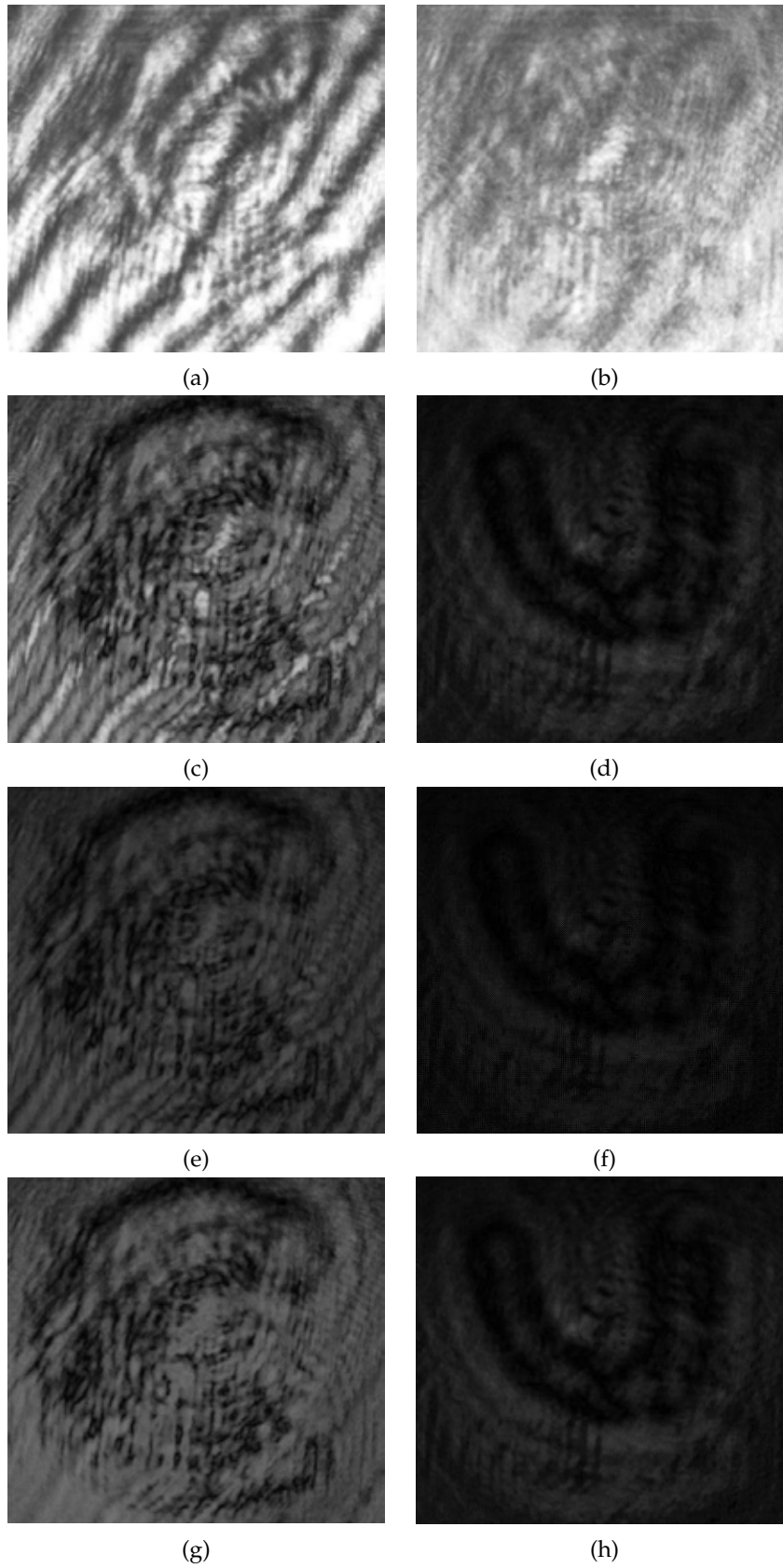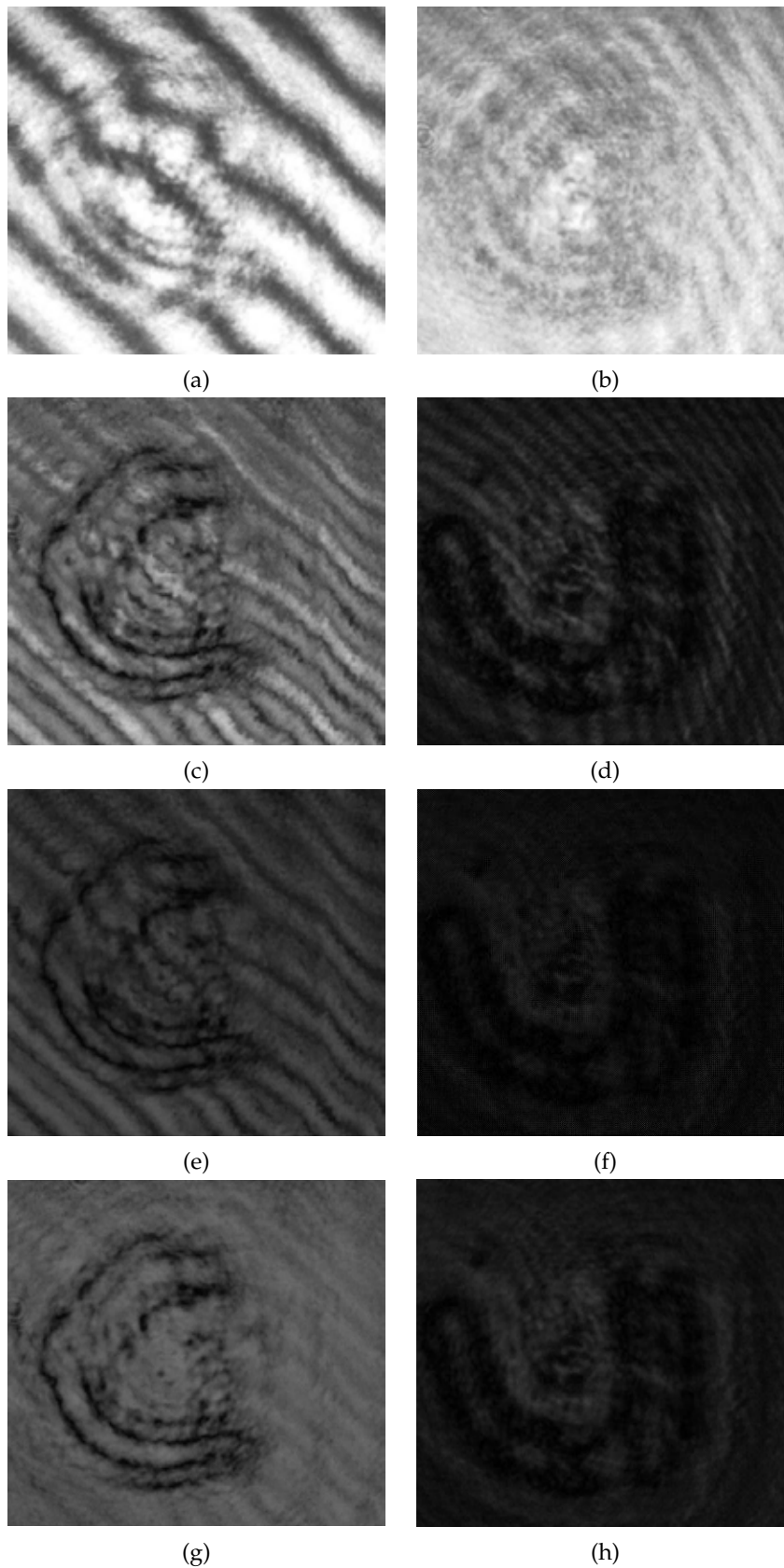
FIGURE 4.6: FFOCT of sample S4 for:
(a) bottom and (b) top layer fringe,
(c) bottom and (d) top layer applied by CDB-algorithm,
(e) bottom and (f) top layer applied by BDB-algorithm,
(g) bottom and (h) top layer applied by 5-step algorithm.

In summary, every algorithms can be used to separate the overlapping "C" and "U"-alphabets. The intensity of images from CDB-algorithm is higher than that of images from BDB-algorithm but almost the same as that of images from 5-step algorithm. Images from BDB-algorithm have lower intensity making the alphabets hard to be seen. This agrees with the simulation result in Section 4.1.2. However, parasitic fringe remains most in the images from CDB-algorithm for all samples. This agrees with the simulation result of $\sigma^2(\Delta I)$ in Section 4.1.2. There is almost no parasitic fringe for 5-step algorithm, but some parasitic fringe remains in images from BDB-algorithm, which contradicts the simulation result for all samples. The remaining parasitic fringe would be induced by experimental error.

### 4.2.2   The effect of properties of markers and reflecting surfaces

In this section, fringe-removed images of the sample S1, S2, S3, and S4 for each algorithm, namely CDB-, BDB-, and 5-step algorithms, will be shown. The quality of fringe-removed images of each sample will be compared.

**Images from CDB-algorithm**

In case of CDB-algorithm, Fig. 4.7a and 4.7b show fringe-removed images of the reflected surface (bottom layer) and the top of a cover slide (top layer) of sample S1, respectively. Fig. 4.7c and 4.7d show fringe-removed images of the bottom and top layers of sample S2, respectively. Fig. 4.7e and 4.7f show fringe-removed images of the bottom and top layers of sample S3, respectively. Fig. 4.7g and 4.7h show fringe-removed images of the bottom and top layers of sample S4, respectively.

According to our results, CDB-algorithm can separate the overlapping alphabets in every sample. All images have high intensity; also, some parasitic fringe can be observed. Sample S3 with "C" written by a blue marker and "U" written by a silver marker can be seen most obviously. Moreover, parasitic fringe on the top of a cover slide is less than the one on the reflected surface since the surface of a cover slide produces lower fringe contrast than the reflected surface. This agrees with the simulation result as in Section 4.1.3 that the variance $\sigma^2(\Delta I)$ of CDB-algorithm increases with fringe contrast.

(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

FIGURE 4.7: Fringe-removed images from FFOCT with
CDB-algorithm of : (a) bottom and (b) top layers of S1,
(c) bottom and (d) top layers of S2,
(e) bottom and (f) top layers of S3,
(g) bottom and (h) top layers of S4.

**Images from BDB-algorithm**

In case of BDB-algorithm, Fig. 4.8a and 4.8b show fringe-removed images of the reflected surface (bottom layer) and the top of a cover slide (top layer) of sample S1, respectively. Fig. 4.8c and 4.8d show fringe-removed images of the bottom and top layers of sample S2, respectively. Fig. 4.8e and 4.8f show fringe-removed images of the bottom and top layers of sample S3, respectively. Fig. 4.8g and 4.8h show fringe-removed images of the bottom and top layers of sample S4, respectively.

According to our results, BDB-algorithm can separate the overlapping alphabets in every sample. All images have low intensity; also, some parasitic fringe can be observed. A large drop in intensity together with the fact, that the top of a cover slide less reflect light, makes fringe-removed images for "U"-alphabets very dark and hard to see. However, sample S3 with "C" written by a blue marker and "U" written by a silver marker can be seen most obviously. Moreover, parasitic fringe on the top of a cover slide is less than the one on the reflected surface since the surface of a cover slide produces lower fringe contrast than the reflected surface. This does not agree with the simulation result as in Section 4.1.3 that the variance $\sigma^2(\Delta I)$ of BDB-algorithm does not vary with fringe contrast.
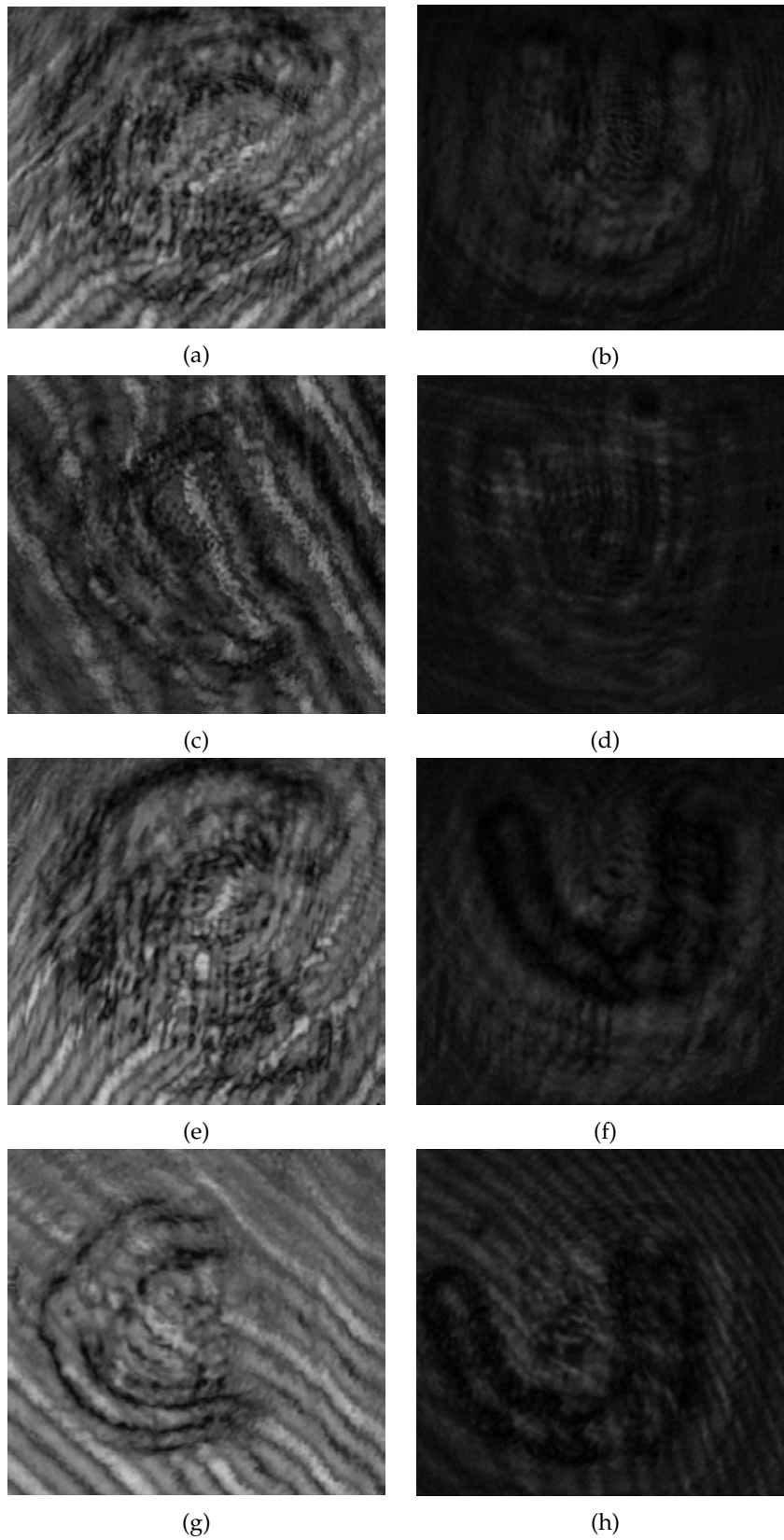
FIGURE 4.8: Fringe-removed images from FFOCT with
BDB-algorithm of : (a) bottom and (b) top layers of S1,
(c) bottom and (d) top layers of S2,
(e) bottom and (f) top layers of S3,
(g) bottom and (h) top layers of S4.

**Images from 5-step algorithm**

In case of 5-step algorithm, Fig. 4.9a and 4.9b show fringe-removed images of the reflected surface (bottom layer) and the top of a cover slide (top layer) of sample S1, respectively. Fig. 4.9c and 4.9d show fringe-removed images of the bottom and top layers of sample S2, respectively. Fig. 4.9e and 4.9f show fringe-removed images of the bottom and top layers of sample S3, respectively. Fig. 4.9g and 4.9h show fringe-removed images of the bottom and top layers of sample S4, respectively.

According to our results, 5-step algorithm can separate the overlapping alphabets in every sample. All images have high intensity; also, almost no parasitic fringe can be observed. Sample S3 with "C" written by a blue marker and "U" written by a silver marker can be seen most obviously. Moreover, parasitic fringe on the top of a cover slide is less than the one on the reflected surface since the surface of a cover slide produces lower fringe contrast than the reflected surface. This does not agree with the simulation result as in Section 4.1.3 that the variance $\sigma^2(\Delta I)$ of 5-step algorithm does not vary with fringe contrast.
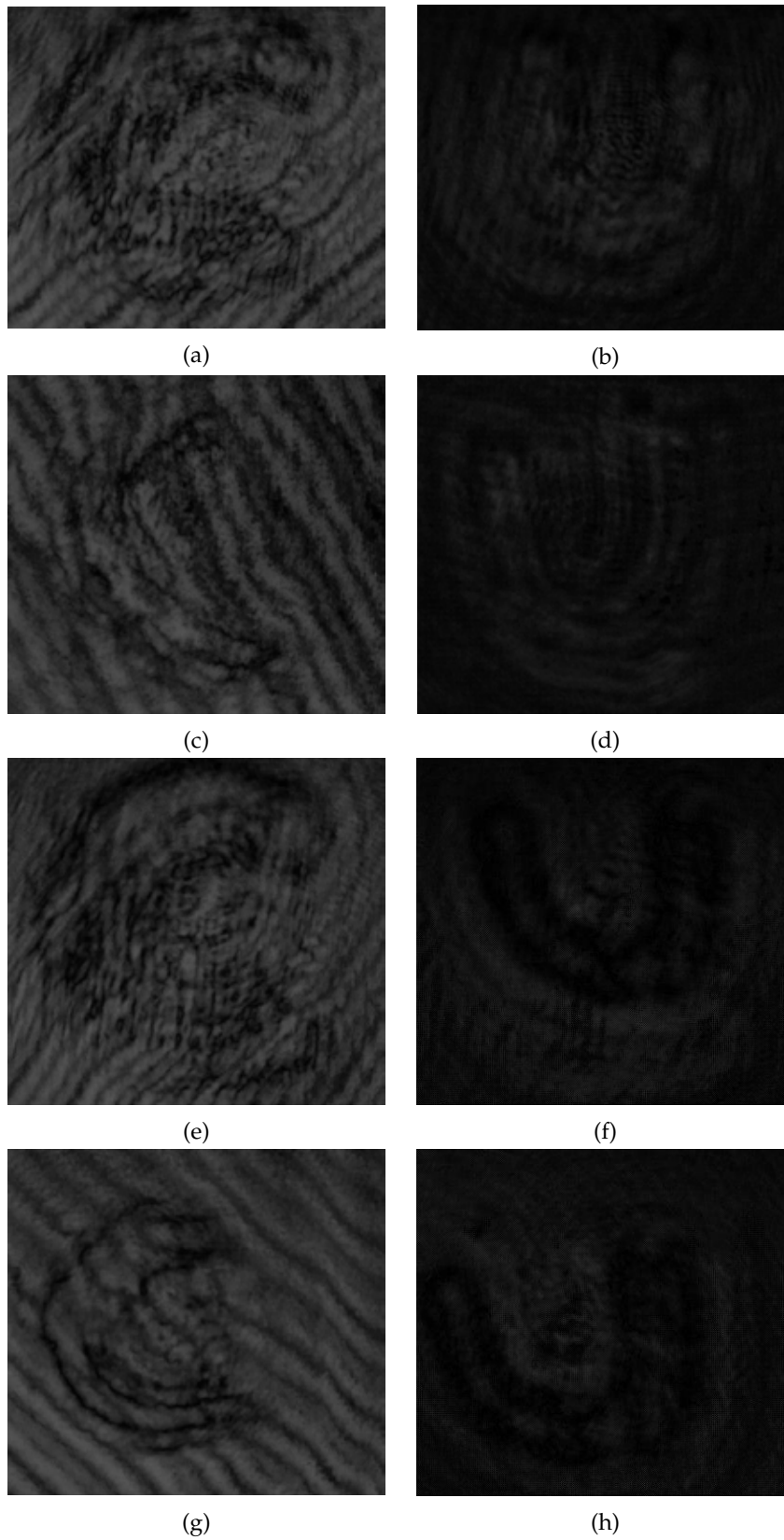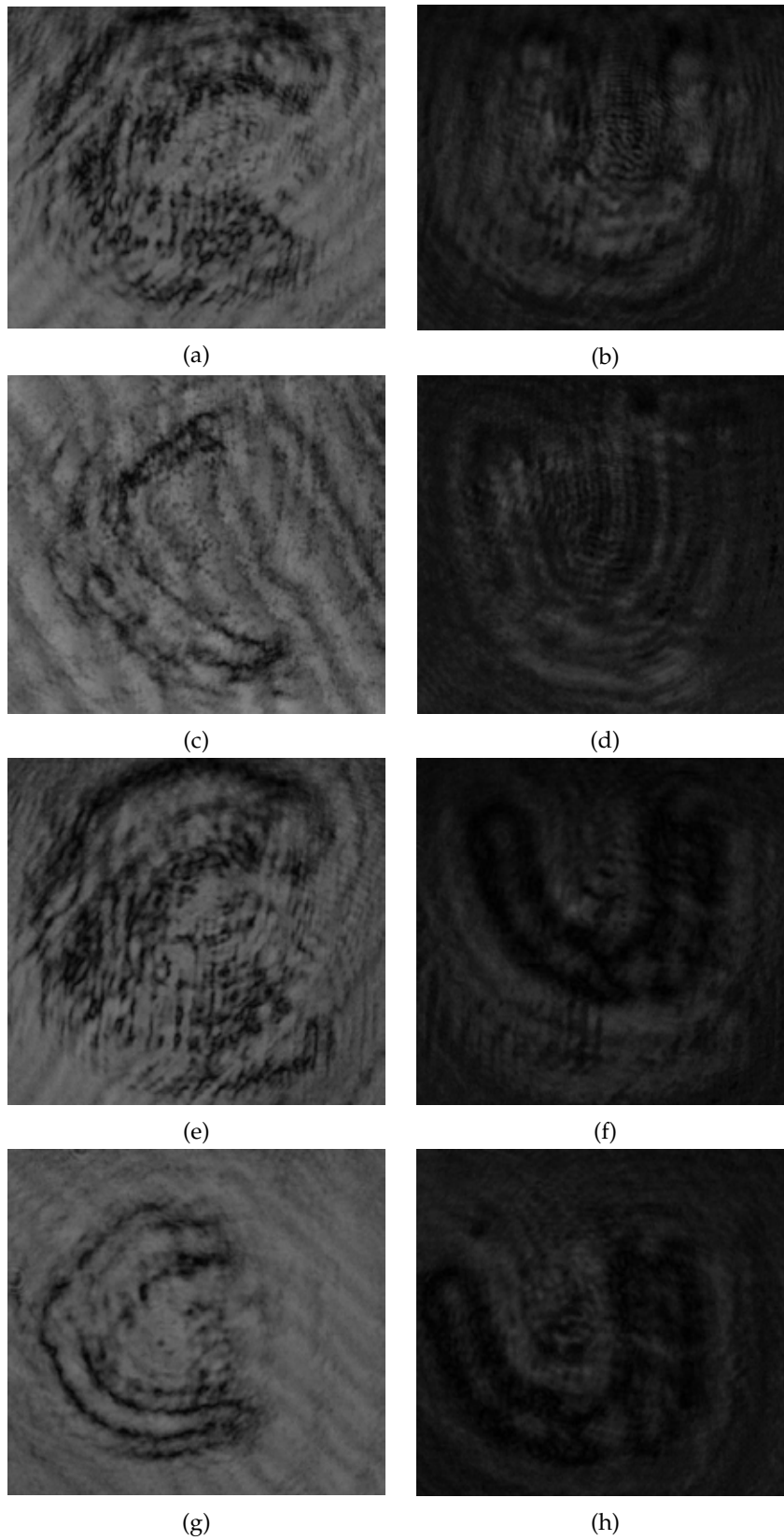
FIGURE 4.9: Fringe-removed images from FFOCT with
5-step algorithm of : (a) bottom and (b) top layers of S1,
(c) bottom and (d) top layers of S2,
(e) bottom and (f) top layers of S3,
(g) bottom and (h) top layers of S4.

It should be noted that a surface or a color with high reflective property gives us the images with high fringe contrast $\gamma$. It can be seen that the bottom layer gives us far higher fringe contrast images than those from the top layer as in Fig. 4.3a, 4.3b, 4.4a, 4.4b, 4.5a, 4.5b, 4.6a, and 4.6b. Additionally, blue alphabets have lower reflective property, producing lower fringe contrast, than silver alphabets.

In case that the contradiction of fringe contrast between the surface and the background is high, the alphabet image can be seen obviously, as in Fig. 4.7e, 4.7f, 4.8e, 4.8f, 4.9e, and 4.9f. When the fringe contrast of an alphabet and a surface is almost similar, the alphabet is hard to be seen, as in Fig. 4.7c, 4.7d, 4.8c, 4.8d, 4.9c, and 4.9d. Therefore, the most easily seen images are those written by a blue marker on a reflected surface as in sample S1 and S3 and those written by a silver marker on a cover slide as in sample S3 and S4.

# Chapter 5

# Discussion and Conclusion

Full-field optical coherence tomography (FFOCT) is a technique, using the Michelson interferometer (MI) with a low coherence light source to retrieve full-field images of a sample at different layer. It is developed to increase the speed of optical coherence tomography (OCT). However, in FFOCT there is unavoidable interference fringe in the full-field images. To remove such fringe, fringe-removing algorithm like backward derivative-based (BDB) and 5-step algorithms are applied to the images. This project aims to introduce a central derivative-based (CDB) algorithm for developing the quality of full-field images taken from FFOCT.

To define the efficiency of CDB-algorithm, a simulation and an experiment are conducted. In an simulation, the calculation time, the effect of phase step, and the effect of fringe contrast in each algorithm are reported. In our experiment, CDB-algorithm is applied to distinguish overlapping alphabets on different layers for studying the effect of fringe-removing algorithms and the optical properties of alphabets and surfaces. Also, the results from the simulation are compared to those from the experiment. Moreover, the similarity and difference in results of CDB-algorithm and other two algorithms are discussed.

For calculation time, it is shown that CDB- and 5-step algorithms require 0.1549 s and 0.2724 s more time to process than BDB-algorithm, respectively. Obviously, for the calculation of 162 $300 \times 300$ pixels-size, the difference in calculation time is less than a half second, which is insignificant.

For the effect of phase step, the sequences of images with fringe contrast of 1 at phase step 10, 20, 30, 40, 50, 60, 70, 80, and 90 degree are simulated. CDB-, BDB-, and 5-step algorithms are applied to remove fringe from the simulated images.

By comparing, the mean $\langle \Delta I \rangle$ of the fringe-removed images, $\langle \Delta I \rangle$ of CDB- and BDB-algorithms decreases with phase step. $\langle \Delta I \rangle$ from CDB-algorithm is around one-tenth and the one of BDB-algorithm is around a half of the original image. It means that fringe-removed images from CDB-algorithm are brighter than those from BDB-algorithm. However, $\langle \Delta I \rangle$ of 5-step algorithm is almost zero. In addition, $\sigma^2(\Delta I)$ of BDB- and 5-step algorithms is also nearly zero, while that of CDB-algorithm is insignificantly low especially at phase step less than 40 degree.

For the effect of fringe contrast, the sequences of images at phase step 84.83 degree with fringe contrast of 0.2, 0.4, 0.6, 0.8, and 1 are simulated. CDB-, BDB-, and 5-step algorithms are finally applied for those of simulated images. In this section, $\langle \Delta I \rangle$ of CDB- and BDB-algorithms also linearly decreases with fringe contrast. The reduction of $\langle \Delta I \rangle$ for CDB-algorithm is less than one-tenth of the original image while that for BDB-algorithm is around 0.3. Therefore, fringe-removed images from CDB-algorithm are also brighter than the ones from BDB-algorithm. However, $\langle \Delta I \rangle$ of 5-step algorithm is almost zero. In addition, $\sigma^2(\Delta I)$ of BDB- and 5-step algorithms is very low, compared to the one from CDB-algorithm, which increases with fringe contrast. This makes the fringe-removed images from CDB-algorithm have more parasitic fringe than the ones from BDB- and 5-step algorithms.

In the experiment, four samples are prepared with overlapping "C"- and "U"-alphabets written on a reflected surface under a cover slide and on the top surface of the cover slide, respectively. The alphabets are written with a blue maker and a silver maker so that each sample differs. Finally, CDB-, BDB-, and 5-step algorithms are applied to distinguish the overlapping "C"- and "U"-alphabets on each sample.

For the effect of fringe-removing algorithms, it is found that every algorithm can be used to separate the overlapping "C"- and "U"-alphabets. The intensity of images from CDB-algorithm is higher than that of images from BDB-algorithm but almost the same as that of images from 5-step algorithm. Since images from BDB-algorithm have lower intensity, the alphabet images from this method are hard to be seen. However, parasitic fringe remains most in the images from CDB-algorithm. On the other hands, there is almost no parasitic fringe for

5-step algorithm. These results agree with the simulation results. However, some parasitic fringe remains in images from BDB-algorithm as well. This contradicts the simulation result. The remaining parasitic fringe would be induced by experimental error, such as, flatness of sample's surface, etc.

For the effect of optical properties of markers and surfaces, the bottom layer gives us far higher fringe contrast images than those from the top layer. This is because the high reflectivity of the surface gives us high fringe contrast images. Also, silver marker has higher reflective property than blue one; thus, it produces high fringe contrast images. When the contradiction of fringe contrast between the surface and the background is high, the alphabet image can be seen obviously. In the opposite way, the fringe contrast of an alphabet and a surface is almost similar, the alphabet is hard to be seen.

In conclusion, the calculation times for CDB-, BDB-, and 5-step algorithms are insignificantly different. The CDB-algorithm produces fringe-removed images with higher intensity than the ones from BDB-algorithm and almost as same as the ones from 5-step algorithm. However, fringe-removed images from CDB-algortihm have the most parasitic fringe. BDB-algorithm leaves some parasitic fringe, contrasting with the simulation result. That is because of experimental factors, such as, flatness of sample's surface, etc. However, 5-step algorithm leave almost no fringe. Lastly, the alphabets can be distinguished obviously when they are written with the marker, having different reflective property from the surface.

# Appendix A

# The effect of phase step and fringe contrast to CDB-algorithm

## A.1  The effect of phase step to CDB-algorithm

TABLE A.1: The relation data between phase step and the mean $\langle \Delta I \rangle$ and the variance $\sigma^2(\Delta I)$ of CDB-algorithm

| Phase step (degree) | $\langle \Delta I \rangle$ (au) | $\sigma^2(\Delta I)$ (au) |
|---|---|---|
| 10 | -2.6302 $\times 10^{-5}$ | 1.2832 $\times 10^{-10}$ |
| 20 | -4.1523 $\times 10^{-4}$ | 3.1769 $\times 10^{-8}$ |
| 30 | -0.0021 | 7.7014 $\times 10^{-7}$ |
| 40 | -0.0063 | 7.1177 $\times 10^{-6}$ |
| 50 | -0.0148 | 3.8402 $\times 10^{-5}$ |
| 60 | -0.0291 | 1.4623 $\times 10^{-4}$ |
| 70 | -0.0509 | 4.3479 $\times 10^{-4}$ |
| 80 | -0.0812 | 0.0011 |
| 90 | -0.1206 | 0.0023 |

TABLE A.2: The relation data between phase step and the mean $\langle \Delta I \rangle$ and the variance $\sigma^2(\Delta I)$ of BDB-algorithm

| Phase step (degree) | $\langle \Delta I \rangle$ (au) | $\sigma^2(\Delta I)$ (au) |
|---|---|---|
| 10 | -0.0063 $\times 10^{-5}$ | 4.6110 $\times 10^{-24}$ |
| 20 | -0.0252 $\times 10^{-4}$ | 8.3466 $\times 10^{-26}$ |
| 30 | -0.0559 | 2.0956 $\times 10^{-25}$ |
| 40 | -0.0979 | 7.9713 $\times 10^{-25}$ |
| 50 | -0.1498 | 1.2465 $\times 10^{-24}$ |
| 60 | -0.2103 | 1.6581 $\times 10^{-24}$ |
| 70 | -0.2778 | 5.9313 $\times 10^{-24}$ |
| 80 | -0.3506 | 5.9876 $\times 10^{-24}$ |
| 90 | -0.4268 | 3.9307 $\times 10^{-24}$ |

TABLE A.3: The relation data between phase step and the mean $\langle \Delta I \rangle$ and the variance $\sigma^2(\Delta I)$ of 5-step algorithm

| Phase step (degree) | $\langle \Delta I \rangle$ (au) | $\sigma^2(\Delta I)$ (au) |
|---|---|---|
| 10 | -2.3712 $\times 10^{-4}$ | 4.1591 $\times 10^{-7}$ |
| 20 | -5.4041 $\times 10^{-5}$ | 7.4899 $\times 10^{-8}$ |
| 30 | 5.3995 $\times 10^{-6}$ | 2.1881 $\times 10^{-8}$ |
| 40 | 4.7945 $\times 10^{-5}$ | 3.0190 $\times 10^{-8}$ |
| 50 | 1.0542 $\times 10^{-4}$ | 1.3834 $\times 10^{-8}$ |
| 60 | 5.3582 $\times 10^{-5}$ | 7.6828 $\times 10^{-9}$ |
| 70 | 7.8029 $\times 10^{-5}$ | 1.1660 $\times 10^{-8}$ |
| 80 | 2.5957 $\times 10^{-5}$ | 1.5709 $\times 10^{-9}$ |
| 90 | -3.9121 $\times 10^{-17}$ | 1.7832 $\times 10^{-30}$ |

## A.2 The effect of fringe contrast to CDB-algorithm

TABLE A.4: The relation data between fringe contrast and the mean $\langle \Delta I \rangle$ and the variance $\sigma^2(\Delta I)$ of CDB-algorithm

| Fringe contrast | $\langle \Delta I \rangle$ (au) | $\sigma^2(\Delta I)$ (au) |
|---|---|---|
| 0.2 | -0.0198 | $6.2768 \times 10^{-5}$ |
| 0.4 | -0.0396 | $2.5107 \times 10^{-4}$ |
| 0.6 | -0.0595 | $5.6491 \times 10^{-4}$ |
| 0.8 | -0.0793 | 0.0010 |
| 1.0 | -0.0991 | 0.0016 |

TABLE A.5: The relation data between fringe contrast and the mean $\langle \Delta I \rangle$ and the variance $\sigma^2(\Delta I)$ of BDB-algorithm

| Fringe contrast | $\langle \Delta I \rangle$ (au) | $\sigma^2(\Delta I)$ (au) |
|---|---|---|
| 0.2 | -0.0774 | $5.3605 \times 10^{-26}$ |
| 0.4 | -0.1548 | $2.1442 \times 10^{-25}$ |
| 0.6 | -0.2323 | $1.0776 \times 10^{-24}$ |
| 0.8 | -0.3097 | $8.5768 \times 10^{-25}$ |
| 1.0 | -0.3871 | $1.5730 \times 10^{-24}$ |

TABLE A.6: The relation data between fringe contrast and the mean $\langle \Delta I \rangle$ and the variance $\sigma^2(\Delta I)$ of 5-step algorithm

| Fringe contrast | $\langle \Delta I \rangle$ (au) | $\sigma^2(\Delta I)$ (au) |
|---|---|---|
| 0.2 | $-1.0517 \times 10^{-13}$ | $6.1507 \times 10^{-28}$ |
| 0.4 | $-2.1034 \times 10^{-13}$ | $2.4603 \times 10^{-27}$ |
| 0.6 | $-3.1543 \times 10^{-13}$ | $5.5327 \times 10^{-27}$ |
| 0.8 | $-4.2063 \times 10^{-13}$ | $9.8385 \times 10^{-27}$ |
| 1.0 | $-5.2575 \times 10^{-13}$ | $1.5369 \times 10^{-26}$ |

# Appendix B

# MATLAB Program

## B.1   Fringe Simulation

```
function I_total = fringe_sim_plain(Lx, Ly, sep, lambda, FWHM, phase_step,
z1, z2, I0, gamma, n, length_c, length_r)
    %I = fringe_sim_plain(Lx, Ly, sep, lambda, FWHM, phase_step, z1, z2, I0,
    gamma, n, length_c, length_r)
    %Lx = horizontal space between fringes
    %Ly = vertical space between fringes
    %sep = position of vertical seperation
    %lambda = central wavelength
    %FWHM = FWHM spectrum width
    %phase_step = phase shift between steps (degrees)
    %z1 = 1st sample phase (in deg in increasing order)
    %z2 = 2nd sample phase
    %I0 = central intensity
    %gamma = fringe contrast
    %n = refractive index
    %length_c = numbers of column
    %length_r = numbers of row

    r = 1:length_r;
    r = r.';
```

```matlab
step = phase_step*lambda/4/180;
z1 = z1*lambda/4/180;
z2 = z2*lambda/4/180;


lc=sqrt(2.0*log(2.0)/pi)*lambda*lambda/n/FWHM;
z_times=ceil((2.0*lc+(z2-z1))/step);
lc_times=nearest(lc/step);


I_total = zeros(length_r, length_c, z_times+1);
for i=0:z_times
    z = (i-lc_times)*step+z1;
    for k = 1:sep
        I = I0+I0*gamma.*exp(-((z-z1)/lc).^2).*cos(4.0*pi*((z-z1)/lambda)
        +2.0*pi*(k/Lx+r./Ly));
        im(1:length_r,k) = I;
    end


    for k = sep:length_c
        I = I0+I0*gamma.*exp(-((z-z2)/lc).^2)
        .*cos(4.0*pi*((z-z2)/lambda)+2.0*pi*(k/Lx+r./Ly));
        im(1:length_r,k) = I;
    end


    I_total(:,:,i+1) = im;
end
end
```

## B.2  Central Derivative-Based Algorithm

```
function I_envelope = fringe_del_cen_diff(I_raw, phase_step)
    [size_row,size_column,size_layer] = size(I_raw);
    step = phase_step*pi/180;


    I_envelope = zeros(size_row, size_column, size_layer-6);
    %use central difference of order O(step^4)
    for layer = 4:size_layer-3
        I_1dot = (-I_raw(:,:,layer+2)+8.*I_raw(:,:,layer+1)
        -8.*I_raw(:,:,layer-1)+I_raw(:,:,layer-2))./12./step;
        I_2dot = (-I_raw(:,:,layer+2)+16.*I_raw(:,:,layer+1)
        -30.*I_raw(:,:,layer)+16.*I_raw(:,:,layer-1)
        -I_raw(:,:,layer-2))./12./step./step;
        I_3dot = (-I_raw(:,:,layer+3)+8.*I_raw(:,:,layer+2)
        -13.*I_raw(:,:,layer+1)+13.*I_raw(:,:,layer-1)-8.*I_raw(:,:,layer-2)
        +I_raw(:,:,layer-3))./8./step./step./step;
        I_envelope(:,:,layer-3) = sqrt(abs(I_2dot.*I_2dot-I_3dot.*I_1dot));
    end


    %{
    %use central difference of order O(step^2)
    for layer = 3:size_layer-2
        I_1dot = (I_raw(:,:,layer+1)-I_raw(:,:,layer-1))./(2*step);
        I_2dot = (I_raw(:,:,layer+1)
        -2.*I_raw(:,:,layer)+I_raw(:,:,layer-1))./(step*step);
        I_3dot = (I_raw(:,:,layer+2)
        -2.*I_raw(:,:,layer+1)+2.*I_raw(:,:,layer-1)-I_raw(:,:,layer-2))
        ./(2*step*step*step);
        I_envelop(:,:,layer) = sqrt(abs(I_2dot.*I_2dot-I_3dot.*I_1dot));
    end
    %}
```

```matlab
%ratio = max(I_raw,[],'all')/max(I_envelop,[],'all');
%I_envelop = I_envelop.*ratio;
end
```

## B.3   Backward Derivative-Based Algorithm

```matlab
function I_envelope = fringe_del_back_diff(I_raw, phase_step)
    %phase_step is in degree.
    [size_row,size_column,size_layer] = size(I_raw);
    step = phase_step*pi/180;


    I_envelope = zeros(size_row, size_column, size_layer-3);
    for layer = 1:size_layer-3
        I1(:,:) = I_raw(:,:,layer);
        I2(:,:) = I_raw(:,:,layer+1);
        I3(:,:) = I_raw(:,:,layer+2);
        I4(:,:) = I_raw(:,:,layer+3);
        I_envelope(:,:,layer) = sqrt(abs(((I3-2*I2+I1)./step./step).^2-
        ((I4-3*(I3-I2)-I1)./step./step./step).*((I2-I1)./step)));
        %I_envelope(:,:,layer) = sqrt(abs(((I3-2*I2+I1)./step./step).^2+
        ((I2-I1)./step).^2));
    end


    %ratio = max(I_raw,[],'all')/max(I_envelop,[],'all');
    %I_envelop = I_envelop./ratio;
end
```

## B.4   5-step phase-shift Algorithm

```
function I_envelop = fringe_del_5step_exp(I_raw)
    [size_row,size_column,size_layer] = size(I_raw);
    I_envelop = zeros(size_row, size_column, size_layer-4);


    for layer = 3:size_layer-2
        I1(:,:) = I_raw(:,:,layer-2);
        I2(:,:) = I_raw(:,:,layer-1);
        I3(:,:) = I_raw(:,:,layer);
        I4(:,:) = I_raw(:,:,layer+1);
        I5(:,:) = I_raw(:,:,layer+2);


        tmpsin(:,:) = sqrt(1-(0.5.*(I1-I5)./(I2-I4)).^2);
        tmpsin = real(tmpsin);
        sin = mean(tmpsin,'all','omitnan');


        I_envelop(:,:,layer-2) = sqrt(abs(((I2-I4)./(2.*sin)).^2+
        ((2.*I3-I5-I1)./(4.*sin.*sin)).^2));
    end


    %ratio = max(I_raw,[],'all')/max(I_envelop,[],'all');
    %I_envelop = I_envelop.*ratio;
end
```

# Bibliography

[1] D. Huang et al. "Optical coherence tomography". In: *Science* 254 (1991), pp. 1178–1181.

[2] Y. Jia et al. "Quantitative optical coherence tomography angiography of vascular abnormalities in the living human eye". In: *Proceedings of the National Academy of Sciences of the United States of America* 112 (2015), pp. 2395–2402.

[3] MR. Hee et al. "Optical coherence tomography of the human retina". In: *Archives of ophthalmology* 113 (1995), pp. 325–332.

[4] O. Tan et al. "Detection of macular ganglion cell loss in glaucoma by Fourier-domain optical coherence tomography". In: *Ophthalmology* 116 (2009), pp. 2305–2314.

[5] S. Chang, X. Cai, and C.Flueraru. "An efficient algorithm used for full-field optical coherence tomography". In: *Optics and Lasers in Engineering* 45 (2007), pp. 1170–1176.

[6] S. Bourquin et al. "Video-rate optical low-coherence reflectometry based on a linear smart detector array". In: *Optics Letters* 25 (2000), pp. 2102–2104.

[7] A. Dubois, AC. Boccara, and M. Lebec. "Real-time reflectivity and topography imagery of depth-resolved microscopic surfaces". In: *Optics Letters* 24 (1999), pp. 309–311.

[8] JM. Schmitt, SL. Lee, and KM. Yung. "An optical coherence microscope with enhanced resolving power in thick tissue". In: *Optics communications* 142 (1997), pp. 203–207.

[9] E. Hecht. *Optics*. 4th ed. Pearson Education Limited, 2014.

[10] K. J. Gasvik. *Optical Metrology*. 3rd ed. John Wiley & Sons, Inc., 2002.

[11] M. Li, C. Quan, and C. Tay. "Continuous wavelet transform for micro-component profile measurement using vertical scanning interferometry". In: *Optics & Laser Technology* 40 (2008), pp. 920–929.

[12] S. C. Chapra. *Numerical Methods for Engineers*. 6th ed. McGraw-Hill, 2009.